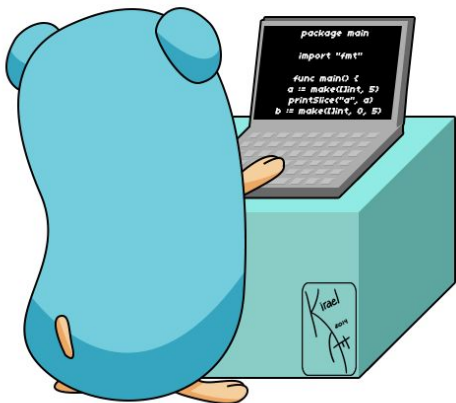


如何向Go官方提交代码

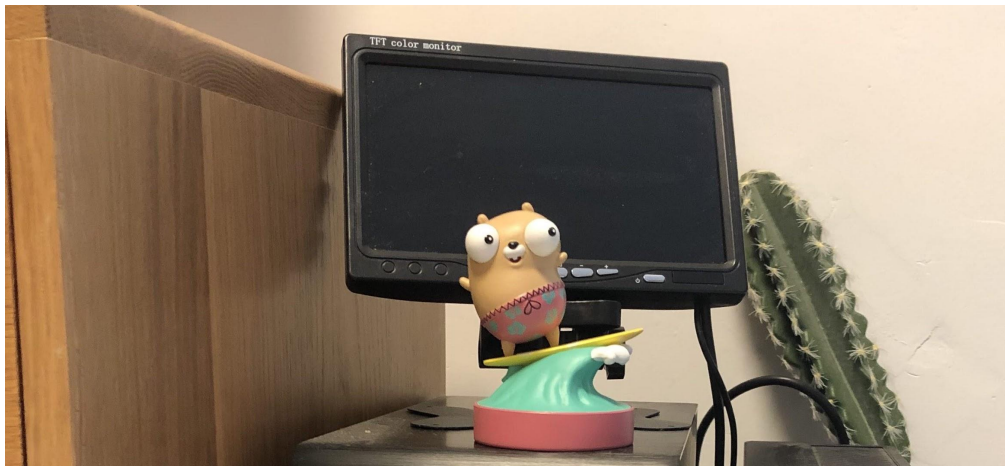
蒙卓

hi@mzh.io



自我介绍

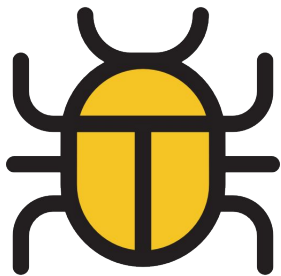
- 后端工程师，主要写Go
- 业余搞Go官方库开发
- 主要喜欢折腾ARM/MIPS 性能优化
- 第一个中国Go 官方MIPS64 builder 维护者



目录

1. 为啥参与Go开发
2. 如何提交CL
3. 交流技巧
4. 参考资料

为啥参与Go开发？



解决BUG

4000+ issue

2510 TODO

typo



提升性能

2400%增速

~50%减少内存



增长知识

runtime

垃圾回收

编译器

HTTP3

密码学....

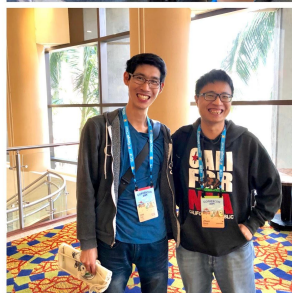
为啥参与Go开发？

没准.....还可以去美国玩一圈

感谢Ben Shi 大佬引荐



—Pitu



07/25

San Diego

如何提交CL

Change Log/List (CL)

The screenshot shows a web browser window displaying a Google Source review page. The URL is `go-review.googlesource.com/c/go/+209317`. The page title is "cmd/link: fix loadelf failed on MIPS family". The status is "Merged as ffa5c85".

Metadata:

- Updated: Dec 10
- Owner: Meng Zhuo
- Uploader: Cherry Zhang
- Committer: Cherry Zhang
- Assignee: Set assignee...
- Reviewers: Gobot Gobot, Cherry Zhang, Than McIntosh
- CC: ADD CC
- Repo: go
- Branch: master
- Parent: a3a630b
- Topic: No topic
- Hashtags: ADD HASHTAG

Code-Review: +2 (Cherry Zhang)

Other labels:

- Run-TryBot: +1 (Cherry Zhang)
- TryBot-Result: -1 (Gobot Gobot)

Review Content:

cmd/link: fix loadelf failed on MIPS family

The relocation of MIPS64 family ELF is different with other architecture according to the document from Linux-MIPS

<https://www.linux-mips.org/pub/linux/mips/doc/ABI/elf64-2.4.pdf>

In "2.9 Relocation" it shows relocation section contains five parts:

1. r_sym Elf64_Word Symbol index
2. r_ssym Elf64_Byte Special symbol
3. r_type3 Elf64_Byte Relocation type
4. r_type2 Elf64_Byte Relocation type
5. r_type Elf64_Byte Relocation type

This CL makes loadelf aware the difference.

Update #35779

Change-Id: `Ib221665641972b1c2bfea5a496e3118e5dc0bc45`

Reviewed-on: <https://go-review.googlesource.com/c/go/+209317>

Run-TryBot: Cherry Zhang <cherryyz@google.com>

Reviewed-by: Cherry Zhang <cherryyz@google.com>

Files:

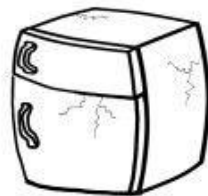
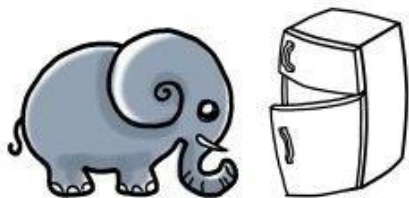
File	Comments	Size	Delta	
Commit message				▼
M src/cmd/link/elf_test.go		■	+0 -7	Reviewed ▼
M src/cmd/link/internal/loadelf/ldelf.go		■	+48 -17	Reviewed ▼
			+48 -24	

如何提交CL

1. 申请Google账号, 填写CLA(Contributor License Agreement)
2. 申请Gerrit(评审平台)账号
3. 安装go-codereview 工具
4. clone Go官方源
5. 切换开发分支
6. 做出修改和commit
7. 执行git mail 即可

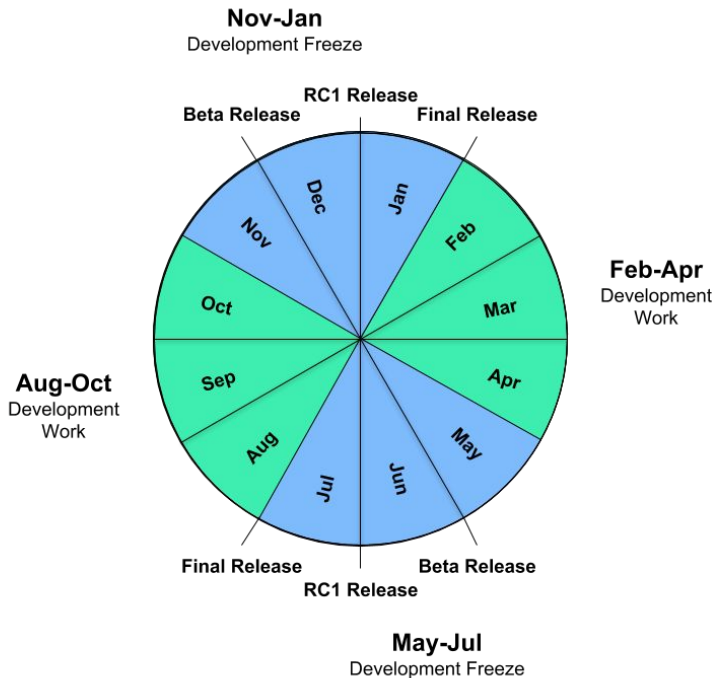
<https://golang.org/doc/contribute.html>

如何提交CL



官方wiki
把大象放进冰箱

如何提交CL



发布周期

如何提交CL

CL类型

1. typo -- 简单
2. 新特性 -- 中等
3. bug -- 困难
4. release-blocker -- 地狱

如何提交CL

CL类型要求

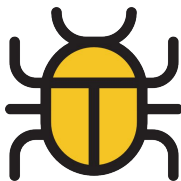
1. typo
2. 新特性
 - a. 新API、新语法, 不能破坏兼容性
 - b. 性能提升, 太少的基本会被打回
 - c. 新硬件指令需要真实环境
3. bug
 - a. 所有平台都没有问题
 - b. 不能引起其他 bug

CL举例



接下来我举个栗子

CL举例



cmd/link: load of "ld -r" host object fails in -newobj mode #35779

Closed thanm opened this issue on 23 Nov · 13 comments

Assignees 

Labels **NeedsFix**

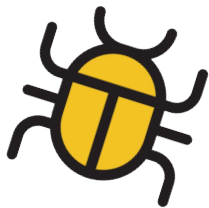
Milestone  Go1.14

 thanm commented on 23 Nov Member ...

What version of Go are you using (`go version`)?

```
$ go version
go version devel +9c6f6409ad Wed Nov 20 15:16:17 2019 +0000 linux/amd64
```

CL举例



cmd/link: load of "ld -r" host obje x +

github.com/golang/go/issues/35779

Closed cmd/link: load of "ld -r" host object fails in -newobj mode #35779
thanm opened this issue on 23 Nov · 13 comments

thanm commented on 23 Nov Member Author ...

Representative failures:

```
https://build.golang.org/log/98d47379c6065fd00edcd68aa384451c5deae722
elf_test.go:202: # elf_test
loadelf: $WORK/b001/_pkg_.a(ldr.syso): malformed elf file: invalid elf sy
```

```
https://build.golang.org/log/ad7532a8bd9457571fe394269d7a017b82abe21e
elf_test.go:202: # elf_test
/tmp/gobuilder-mips64/tmp/go-build246906970/b001/_pkg_.a(ldr.syso): unkno
```

+😊

@goherbat commented on 23 Nov

CL举例



Go Build Dashboard

build.golang.org/?page=4&branch=master

4a378d7	jayc...	26 Nov 20:21	doc: add section on ...	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok
27e50a1	jayc...	26 Nov 19:05	cmd: update golang....	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
67f0f83	cher...	26 Nov 16:55	runtime: disable asy...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
20bf6a4	kyle	26 Nov 12:03	lib/time: update tz d...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
87805c9	sho1...	25 Nov 23:37	os: fix broken comm...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
5de3b97	fsha...	25 Nov 21:15	cmd/link: fix TestStr...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
7adafc9	Jason	25 Nov 19:02	cmd/go: prefix calls ...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
6d5f8b1	bcm...	25 Nov 17:05	misc/cgo/testshare...	fail	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
e16f64c	bcm...	25 Nov 16:43	misc: log 'ok' from '...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
476395c	bcm...	25 Nov 16:26	misc: remove use of...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
01f15b6	dian...	25 Nov 15:08	cmd/internal/obj/ar...	ok	ok	ok	ok	ok	ok	fail	fail	fail	ok	ok
9940c77	dan...	25 Nov 03:41	runtime: add go:nos...	ok	ok	ok	ok	ok	ok	fail	fail	fail	ok	ok
b81dd1c	thanm	24 Nov 15:06	cmd/link: disable ne...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
6f7b96f	tkla...	23 Nov 11:00	syscall: follow conve...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
c2e2296	mail	23 Nov 09:35	syscall/js: handle int...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
688aa74	thanm	23 Nov 00:06	cmd/link: add new li...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
f29e53d	thanm	23 Nov 00:06	cmd/link: fix bug wit...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
6ba1bf3	dmit...	22 Nov 23:23	net/http: rename tes...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
980f0c0	brad...	22 Nov 22:24	net/http: make Tran...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
9f3c2b6	bcm...	22 Nov 21:09	cmd/go/internal/wo...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
8324aca	jayc...	22 Nov 19:17	cmd/go: fix and re-...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok

[newer](#) [older](#) [latest](#)

<https://go-review.golang.org/q/688aa748579f07552a50d2534eccb16afda4174b>

CL举例



Go Build Dashboard

build.golang.org/?page=4&branch=master

4a378d7	jayc...	26 Nov 20:21	doc: add section on ...	ok	ok	ok	ok	ok	ok	fail	ok	ok	ok	ok
27e50a1	jayc...	26 Nov 19:05	cmd: update golang...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 16:55	runtime: disable asy...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 12:03	lib/time: update tz d...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 23:37	os: fix broken comm...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 21:15	cmd/link: fix TestStr...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 19:02	cmd/go: prefix calls ...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 17:05	misc/cgo/testshare...	fail	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 16:43	misc: log 'ok' from '...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 16:26	misc: remove use of...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 15:08	cmd/internal/obj/ar...	ok	ok	ok	ok	ok	ok	fail	fail	fail	ok	ok
		Nov 03:41	runtime: add go:nos...	ok	ok	ok	ok	ok	ok	fail	fail	fail	ok	ok
		Nov 15:06	cmd/link: disable ne...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
		Nov 11:00	syscall: follow conve...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
c2e2296	mail	23 Nov 09:35	syscall/js: handle int...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
688aa74	thanm	23 Nov 00:06	cmd/link: add new li...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
f29e53d	thanm	23 Nov 00:06	cmd/link: fix bug wit...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
6ba1bf3	dmit...	22 Nov 23:23	net/http: rename tes...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
980f0c0	brad...	22 Nov 22:24	net/http: make Tran...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
9f3c2b6	bcm...	22 Nov 21:09	cmd/go/internal/wo...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok
8324aca	jayc...	22 Nov 19:17	cmd/go: fix and re-...	ok	ok	ok	ok	ok	ok	fail	ok	fail	ok	ok

[newer](#) [older](#) [latest](#)

<https://go-review.googlesource.com/q/688aa748579f07552a50d2534eccb16afda4174b>



CL举例

ELF¹⁰¹ a Linux executable walk-through

ANGE ALBERTINI
CORKAM.COM



DISSECTED FILE

```
~$ uname -m  
armv7L  
~$ ./simple.ARM  
Hello World!
```

SIMPLE.ARM

HEADER^{1/2}
TECHNICAL SPECIFICATION
IDENTIFICATION AND EXECUTION

SECTIONS
CONTENTS OF THE EXECUTABLE

HEADER^{2/2}
TECHNICAL DETAILS FOR LINKING
REQUIRED FOR EXECUTION

ELF HEADER
IDENTIFIABLE FILE TYPE
CORKAM.COM

PROGRAM HEADER TABLE
EXECUTABLES ONLY

CODE
EXECUTABLE INSTRUCTION

DATA
INFORMATION USED BY THE CODE

SECTIONS' NAMES

SECTION HEADER TABLE
LINKING CORRECTING PROGRAM OBJECTS TO INFORMATION

HEXADECIMAL DUMP

ASCII DUMP

FIELDS

VALUES

EXPLANATION

1

e_ident

0x7F, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

2

p_type

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

3

0x7F, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

4

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

5

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

6

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

7

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

8

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

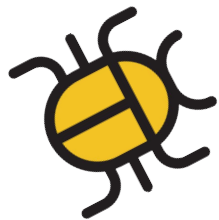
0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

0x00, "ELF"

CL举例



```
tmux a
→ ~
→ ~ readelf -r big32.o

Relocation section '.rel.text' at offset 0x238 contains 1 entry:
  Offset      Info      Type           Sym.Value  Sym. Name
00000034  00000304  R_MIPS_26      00000000   .text

Relocation section '.rel.pdr' at offset 0x240 contains 2 entries:
  Offset      Info      Type           Sym.Value  Sym. Name
00000000  00000302  R_MIPS_32      00000000   .text
00000020  00000c02  R_MIPS_32      00000024   Cfunc1
→ ~
```

```
[0] 0:~ 1:~/godev/src- 2:~/godev/src 3:~* "ls-3a3k" 22:33 31-Dec-19
```

CL举例

elf64-2.4.pdf

linux-mips.org/pub/linux/mips/doc/ABI/elf64-2.4.pdf

there.

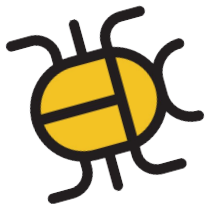
The content of each section is an array of relocation records, as described by Table 29 below:

Table 29 Relocation Operation (*Elf64_Rel*, *Elf64_Rela*)

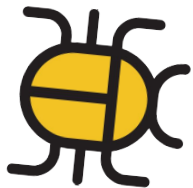
Field Name	Type	Comments
<i>r_offset</i>	Elf64_Addr	Where to apply relocation: relocatable: byte offset in section executable: virtual address
<i>r_sym</i>	Elf64_Word	Symbol index
<i>r_ssym</i>	Elf64_Byte	Special symbol — see Table 30
<i>r_type3</i>	Elf64_Byte	Relocation type — see Table 32
<i>r_type2</i>	Elf64_Byte	Relocation type — see Table 32
<i>r_type</i>	Elf64_Byte	Relocation type — see Table 32
<i>r_addend</i>	Elf64_Sxword	Explicit addend for relocation operation (<i>Elf64_Rela</i> only)

2.9.1 Rules for Interpreting Relocation Records

The relocation operation consists of applying the operation(s) implied by the type subfield(s) to operands which may include the address or offset of the storage unit being relocated (*r_offset*), the current content of the



CL举例



.../ldelf.go · Gerrit Code Review

go-review.googlesource.com/c/go/+209317/5/src/cmd/link/internal/loadelf/ldelf.go

209317: cmd/link: fix loadelf failed on MIPS family — ☒

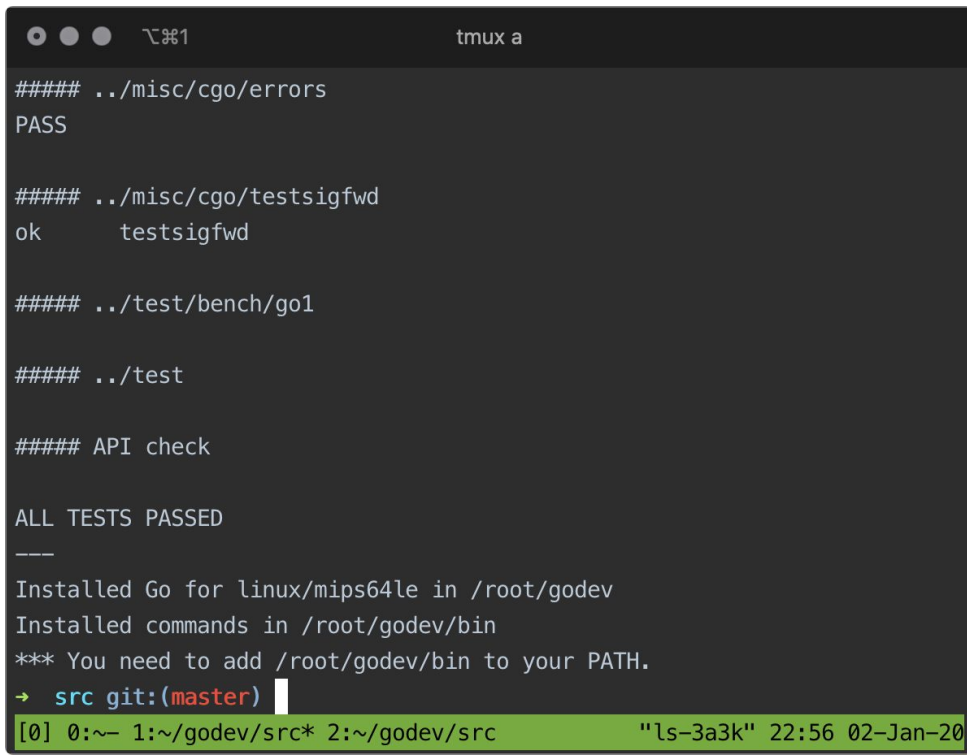
src/cmd/link/internal/loadelf/ldelf.go

Base [gitiles](#) → Patchset 5 [gitiles](#) [DOWNLOAD](#) [SHOW BLAME](#) Diff view:

```
906 906 » » » var add uint64
907 907 » » » var symIdx int
908 908 » » » var relocType uint64
909
910 910 » » » rp := &r[j]
911 911 » » » var info uint64
912 912 » » » if is64 != 0 {
913 913 » » »     // 64-bit rel/rela
914 914 » » »     rp.Off = int32(e.Uint64(p))
915
916 916 » » »     p = p[8:]
917 917 » » »     info = e.Uint64(p)
918 918 » » »     switch arch.Family {
919 919 » » »     case sys.MIPS64:
920 920 » » »         // https://www.linux-mips.org/pub/linux/mips/doc/ABI/elf64-2
921 921 » » »         .4.pdf
922 922 » » »         // The doc shows it's different with general Linux ELF
923 923 » » »         symIdx = int(e.Uint32(p))
924 924 » » »         relocType = uint64(p[7])
925 925 » » »         default:
926 926 » » »             info := e.Uint64(p)
927 927 » » »             relocType = info & 0xffffffff
928 928 » » »             symIdx = int(info >> 32)
929 929 » » »         }
930 930 » » »     p = p[8:]
931 931 » » »     if rela != 0 {
932 932 » » »         add = e.Uint64(p)
```

CL举例

1. 跑全量用例 (cd src; ./all.bash -v)



A terminal window titled "tmux a" showing the output of a test script. The script runs several tests, all of which pass. The output is as follows:

```
##### ../misc/cgo/errors
PASS

##### ../misc/cgo/testsigfwd
ok      testsigfwd

##### ../test/bench/go1

##### ../test

##### API check

ALL TESTS PASSED
---
```

Below the test results, there is a message indicating that Go has been installed for linux/mips64le in /root/godev, and that the commands are installed in /root/godev/bin. It also states that the user needs to add /root/godev/bin to their PATH.

```
Installed Go for linux/mips64le in /root/godev
Installed commands in /root/godev/bin
*** You need to add /root/godev/bin to your PATH.
```

The terminal prompt is `→ src git:(master)`. The status bar at the bottom shows `[0] 0:~ 1:~/godev/src* 2:~/godev/src` and the time `"ls-3a3k" 22:56 02-Jan-20`.

CL举例

1. 跑全量用例 (cd src; ./all.bash -v)
2. commit信息
 - a. 信息要完整, 改了什么, 为什么要改
 - b. 引用数据源
 - c. 如果有对应bug, 那么要带上bug号



CL举例

1. 跑全量用例 (cd src; ./all.bash -v)
2. commit信息

cmd/link: fix loadelf failed on MIPS family

第一行
改了什么模块
本次CL的主题

The relocation of MIPS64 family ELF is different with other architecture according to the document from Linux-MIPS

<https://www.linux-mips.org/pub/linux/mips/doc/ABI/elf64-2.4.pdf>

This CL makes loadelf aware the difference.

Update [#35779](#)

CL 举例

1. 跑全量用例 (cd src; ./all.bash -v)
2. commit 信息

cmd/link: fix loadelf failed on MIPS family

改变的原因
资料引用地址

The relocation of MIPS64 family ELF is different with other architecture according to the document from Linux-MIPS

<https://www.linux-mips.org/pub/linux/mips/doc/ABI/elf64-2.4.pdf>

This CL makes loadelf aware the difference.

Update [#35779](#)

CL举例

1. 跑全量用例 (cd src; ./all.bash -v)
2. commit信息

cmd/link: fix loadelf failed on MIPS family

The relocation of MIPS64 family ELF is different with other architecture according to the document from Lin

<https://www.linux-mips.org/wiki/ABI/elf64-2.4.pdf>

跟哪些bug有关
修复的就用Fixed
没修复就用Update

This CL makes loadelf aware the difference.

Update [#35779](#)

CL举例

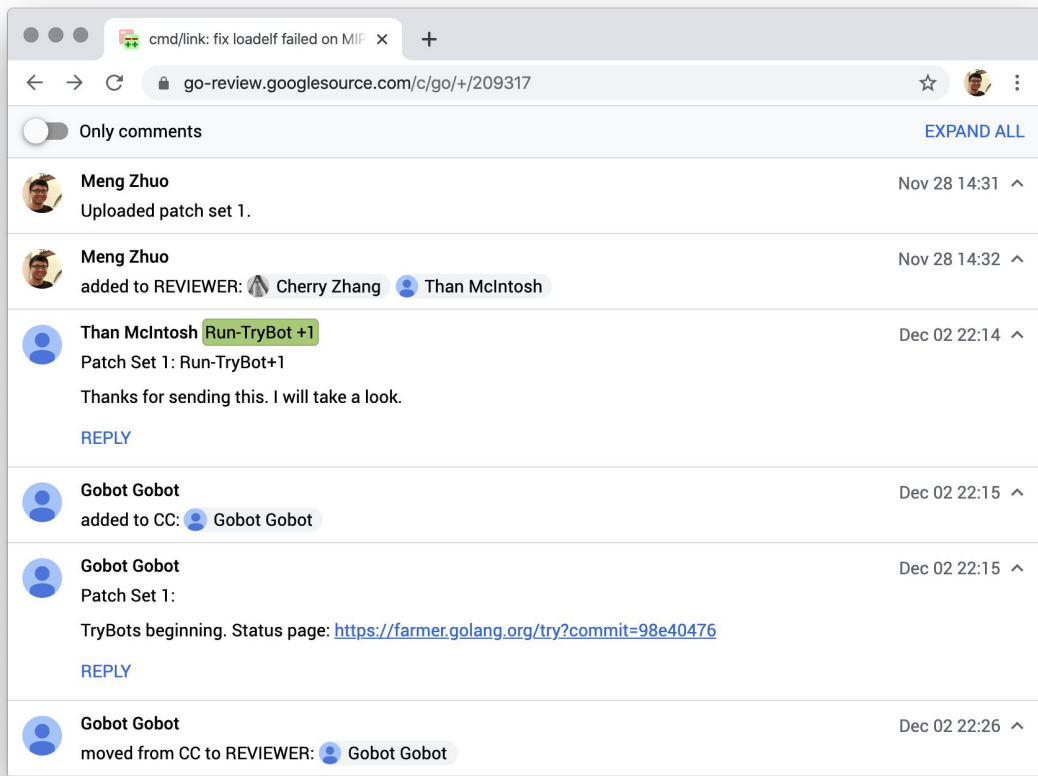
1. 跑全量用例 (cd src; ./all.bash -v)
2. commit信息
 - a. 信息要完整, 改了什么, 为什么要改
 - b. 引用数据源
 - c. 如果有对应bug, 那么要带上bug号
3. git mail 提交
 - a. 记住Change ID
 - b. 下次提交时commit消息要加上 这个Change ID

提交完了

1. 终于可以放松了？

提交完了

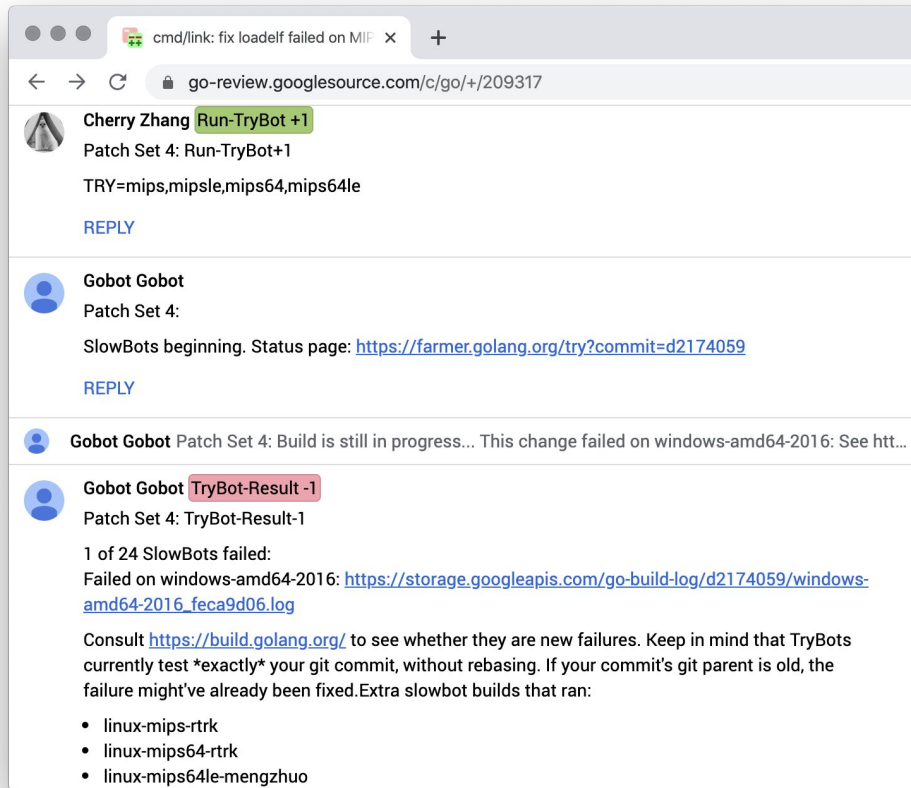
1. Trybot测试



提交完了

1. TryBot测试

- 模拟全平台全测试
- 不通过不让提交
- 通过了会得到: TryBot-Result +1



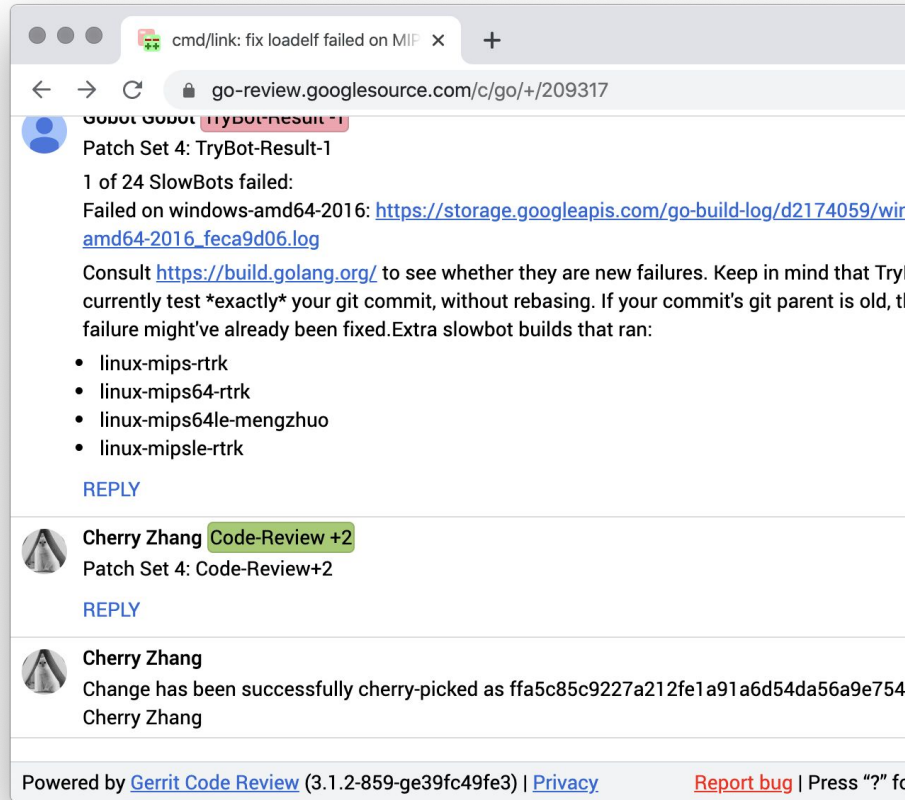
提交完了

1. TryBot测试

- 模拟全平台全测试
- 不通过不让提交
- 通过了会得到: TryBot-Result +1

2. 核心成员

- 讨论, 修改
- 记得commit信息带上ChangeID
- 成员满意后: CodeReview +2



如何提交CL

小结一下

1. 按开发周期提交
2. 测试都要通过
3. commit信息很重要
4. 让TryBot高兴起来

如何交流

1. 开发团队



核心团队



领域大牛

如何交流

1. 开发团队
2. 英语, 英语, 英语
 - a. 不要提中文issue
3. 事实为依据
 - a. 资料要有出处
 - b. benchstat 要带上
4. 政治正确

如何交流

1. 开发团队
2. 英语, 英语, 英语
 - a. 不要提中文issue
3. 事实为依据
 - a. 资料要有出处
 - b. benchstat 要带上
4. 政治正确
5. 俗语
 - a. LGTM: Looks Good To Me (我觉着还行)
 - b. AFAIK: As Far As I Know (据我所知)
 - c. IMHO: In My Humble Opition (鄙人认为)

One More Thing

版权问题

1. 抄也要抄BSD兼容的代码
 - a. GPL 不行
 - b. 闭源更不行

谢谢各位

参考资料

<https://github.com/golang/go/issues>

<https://stackoverflow.com/questions/25716920/what-does-cl-mean-in-commit-message-what-does-it-stand-for>

<https://golang.org/doc/contribute.html>