# 堆中的检查

## _int_malloc

## 初始检查

| 检查目标 | 检查条件 | 信息 |
| --- | --- | --- |
| 申请的大小 | REQUEST_OUT_OF_RANGE(req)： ((unsigned long) (req) >= (unsigned long) (INTERNAL_SIZE_T)(-2 * MINSIZE)) | __set_errno(ENOMEM) |

### fastbin

| 检查目标 | 检查条件 | 报错信息 |
| --- | --- | --- |
| chunk 大小 | fastbin_index(chunksize(victim)) != idx | malloc(): memory corruption (fast) |

### Unsorted bin

| 检查目标 | 检查条件 | 报错信息 |
| --- | --- | --- |
| unsorted bin chunk 大小 | chunksize_nomask (victim) <= 2 * SIZE_SZ \|\| chunksize_nomask (victim) av->system_mem | malloc(): memory corruption |

### top chunk

| 检查目标 | 检查条件 | 信息 |
| --- | --- | --- |
| top chunk size | (unsigned long) (size) >= (unsigned long) (nb + MINSIZE) | 方可进入 |

## __libc_free

- **mmap 块**

| 检查目标 | 检查条件 | 信息 |
| --- | --- | --- |
| chunk size 标记位 | chunk_is_mmapped (p) | 方可进入 |

- **非mmap 块**

## __int_free

- **初始检查**

| 检查目标 | 检查条件 | 报错信息 |
| --- | --- | --- |
| 释放chunk位置 | (uintptr_t) p > (uintptr_t) -size \|\| misaligned_chunk(p) | free(): invalid pointer |
| 释放chunk的大小 | size < MINSIZE \|\| !aligned_OK(size) | free(): invalid size |

- **fastbin**

| 检查目标 | 检查条件 | 报错信息 |
| --- | --- | --- |
| 释放chunk的下一个chunk大小 | chunksize_nomask(chunk_at_offset(p, size)) <= 2 * SIZE_SZ， chunksize(chunk_at_offset(p, size)) >= av->system_mem | free(): invalid next size (fast) |
| 释放 chunk对应链表的第一个chunk | fb = &fastbin(av, idx)， old= *fb， old == p | double free or corruption (fasttop) |
| fastbin索引 | old != NULL && old_idx != idx | invalid fastbin entry (free) |

- **non-mmapped 块检查**

| 检查目标 | 检查条件 | 报错信息 |
| --- | --- | --- |
| 释放chunk位置 | p == av->top | double free or corruption (top) |
| next chunk位置 | contiguous (av) && (char *) nextchunk >= ((char *) av->top + chunksize(av->top)) | double free or corruption (out) |
| next chunk大小 | chunksize_nomask (nextchunk) <= 2 * SIZE_SZ \|\| nextsize >= av->system_mem | free(): invalid next size (normal) |

## unlink

| 检查目标 | 检查条件 | 报错信息 |
| --- | --- | --- |
| size **vs** prev_size | chunksize(P) != prev_size (next_chunk(P)) | corrupted size vs. prev_size |
| Fd, bk 双向链表检查 | FD->bk != P \|\| BK->fd != P | corrupted double-linked list |
| nextsize 双向链表 | P->fd_nextsize->bk_nextsize != P \|\| P->bk_nextsize->fd_nextsize != P | corrupted double-linked list (not small) |