Unlink

> 各个版本中unlink利用方法

Glibc2.23

```
#define unlink(AV, P, BK, FD) {
    if (__builtin_expect (chunksize(P) != prev_size (next_chunk(P)), 0))
 \
      malloc_printerr ("corrupted size vs. prev_size");                \
    FD = P->fd;                                      \
    BK = P->bk;                                      \
    if (__builtin_expect (FD->bk != P || BK->fd != P, 0))            \
      malloc_printerr ("corrupted double-linked list");                \
    else {                                       \
        FD->bk = BK;                                     \
        BK->fd = FD;                                     \
        if (!in_smallbin_range (chunksize_nomask (P))                \
            && __builtin_expect (P->fd_nextsize != NULL, 0)) {           \
        if (__builtin_expect (P->fd_nextsize->bk_nextsize != P, 0)       \
        || __builtin_expect (P->bk_nextsize->fd_nextsize != P, 0))    \
          malloc_printerr ("corrupted double-linked list (not small)");   \
            if (FD->fd_nextsize == NULL) {                       \
                if (P->fd_nextsize == P)                         \
                  FD->fd_nextsize = FD->bk_nextsize = FD;            \
                else {                                   \
                    FD->fd_nextsize = P->fd_nextsize;                \
                    FD->bk_nextsize = P->bk_nextsize;                \
                    P->fd_nextsize->bk_nextsize = FD;                \
                    P->bk_nextsize->fd_nextsize = FD;                \
                  }                                      \
              } else {                                   \
                P->fd_nextsize->bk_nextsize = P->bk_nextsize;            \
                P->bk_nextsize->fd_nextsize = P->fd_nextsize;            \
              }                                          \
          }                                      \
      }                                          \
}
```

Glibc2.27

唯一的差别就是在最开始加了对size的验证

```
1   if (chunksize(P) != prev_size (next_chunk(P)))
2         malloc_printerr ("corrupted size vs. prev_size");
```