

```

◆=  /builddir/build/BUILD/wget-1.16/src/ftp.c
2264     if (f->type != FT_DIRECTORY && !acceptable (f->name))
2265     {
2266         logprintf (LOG_VERBOSE, _("Rejecting %s.\n"),
2267                 quote (f->name));
2268         f = delelement (f, &start);
2269     }
2270     else
2271     {
2272     }
2273 }
2274 /* Remove all files with possible harmful names or invalid entries. */
2275 f = start;

```

6. Falling through to end of if statement

7. Jumping back to the beginning of the loop

9. alias: Assigning: f = start . Now both point to the same storage.

10. Condition f , taking true branch

14. Condition f , taking true branch

19. Condition f , taking false branch

```

◆=  /builddir/build/BUILD/wget-1.16/src/ftp.c
2333 #else /* def __VMS */
2334     int (*cmp) (const char *, const char *)
2335     = opt.ignore_case ? strcasecmp : strcmp;
2336 #endif /* def __VMS [else] */
2337     f = start;
2338     while (f)
2339     {
2340         if (0 != cmp(u->file, f->name))
2341             f = delelement (f, &start);
2342         else
2343             f = f->next;
2344     }
2345 }
2346 }
2347 if (start)
2348 {
2349     /* Just get everything. */
2350     res = ftp_retrieve_list (u, start, con);
2351 }
2352 else
2353 {
2354     if (action == GLOB_GLOBALL)
2355     {

```

23. alias: Assigning: f = start . Now both point to the same storage.

24. Condition f , taking true branch

29. Condition f , taking false branch

25. Condition 0 != (*cmp)(u->file, f->name) , taking true branch

26. freed_arg: delelement frees f . [show details]

27. Falling through to end of if statement

28. Jumping back to the beginning of the loop

30. Condition start , taking true branch

31. pass_freed_arg: Passing freed pointer start as an argument to ftp_retrieve_list .

CID 107799 (#3-2 of 4): Use after free (USE_AFTER_FREE)

/* No luck */

