

crash due to null pointer access when imap server sends early preauth

We observed a crash in balsa. This can be reproduced by having a server that simply sends a PREAUTH command, e.g. by doing this on the command line with netcat and letting balsa connect to localhost via imap:

```
echo -n "" PREAUTH\r\n" | nc -l -p 143
```

I'm pasting an error from Address Sanitizer below, it indicates the error is in libbalsa/imap/imap-handle.c:827.

That code looks like this:

```
handle->can_fetch_body =
(strncmp(handle->last_msg, "Microsoft Exchange", 18) != 0);
```

It seems this crashes when handle->last_msg is not set (i.e. NULL), which causes strcmp accessing a null pointer. preventing this by first checking that handle->last_msg is not NULL prevents the crash, however there's probably an underlying deeper issue within the state management of the imap implementation.

ASAN error:

```
==3171==ERROR: AddressSanitizer: SEGV on unknown address 0x000000000000 (pc 0x7f312ac84c30 bp 0x7ffce7d9e2d0 sp 0x7ffce7d9da30 T0)
==3171==The signal is caused by a READ memory access.
==3171==Hint: address points to the zero page.
#0 0x7f312ac84c2f (/usr/lib/gcc/x86_64-pc-linux-gnu/9.2.0/libasan.so.5+0xd5c2f)
#1 0x55f918e2da93 in imap_mbox_connect /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/libbalsa/imap/imap-handle.c:
#2 0x55f918e2e052 in imap_mbox_handle_connect /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/libbalsa/imap/imap-ha
#3 0x55f918d990b7 in libbalsa_imap_server_get_handle /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/libbalsa/imap
#4 0x55f918d9081c in libbalsa_scanner_imap_dir /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/libbalsa/folder-scan
#5 0x55f918d0cf08 in imap_dir_cb /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/src/mailbox-node.c:582
#6 0x7f3129347d7c in g_closure_invoke (/usr/lib64/libgobject-2.0.so.0+0x13d7c)
#7 0x7f312935aa36 (/usr/lib64/libgobject-2.0.so.0+0x26a36)
#8 0x7f31293637cc in g_signal_emit_valist (/usr/lib64/libgobject-2.0.so.0+0x2f7cc)
#9 0x7f3129363a36 in g_signal_emit (/usr/lib64/libgobject-2.0.so.0+0x2fe36)
#10 0x55f918d1c4c0 in scan_mailbox_idx_cb /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/src/main.c:254
#11 0x7f3129265129f in g_main_context_dispatch (/usr/lib64/libglib-2.0.so.0+0x4e29f)
#12 0x7f31292651667 (/usr/lib64/libglib-2.0.so.0+0x4e667)
#13 0x7f31292651992 in g_main_loop_run (/usr/lib64/libglib-2.0.so.0+0x4e992)
#14 0x7f3129a2d95c in gtk_main (/usr/lib64/libgtk-3.so.0+0x23895c)
#15 0x55f918d1f46c in real_main /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/src/main.c:566
#16 0x55f918d1f46c in command_line_cb /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/src/main.c:749
#17 0x7f3129643bac (/usr/lib64/libffi.so.7+0xb0ac)
#18 0x7f3129643138 (/usr/lib64/libffi.so.7+0x6d138)
#19 0x7f3129458599 in g_closure_marshaller_generic (/usr/lib64/libgobject-2.0.so.0+0x14599)
#20 0x7f3129347d7c in g_closure_invoke (/usr/lib64/libgobject-2.0.so.0+0x13d7c)
#21 0x7f312935aa36 (/usr/lib64/libgobject-2.0.so.0+0x26a36)
#22 0x7f3129362e49 in g_signal_emit_valist (/usr/lib64/libgobject-2.0.so.0+0x2ee49)
#23 0x7f3129363e36 in g_signal_emit (/usr/lib64/libgobject-2.0.so.0+0x2fe36)
#24 0x7f312945a0b2 (/usr/lib64/libgio-2.0.so.0+0xbcb0b2)
#25 0x7f312945ccab (/usr/lib64/libgio-2.0.so.0+0xbcbcab)
#26 0x7f312945ce09 in g_application_run (/usr/lib64/libgio-2.0.so.0+0xbce09)
#27 0x55f918cb4c08 in main /var/tmp/portage/mail-client/balsa-2.5.6-r1/work/balsa-2.5.6/src/main.c:773
#28 0x7f312901ce8a in __libc_start_main ../csu/libc-start.c:308
#29 0x55f918cb5e79 in _start (/usr/bin/balsa+0xb9e79)
```


📁 Drag your designs here or [click to upload](#).


Tasks 📌 0 |

No tasks are currently assigned. Use tasks to break down this issue into smaller parts.

Linked items 📌 0 |

Activity

 Andre Klapper added 1 crash label 2 years ago

 Peter Bloomfield @peterb 2 years ago


Developer

Hi Hanno,

Sorry about the crash, but thanks for the detailed report!

You're right, it could be avoided by testing `handle->last_msg`. But I'm not sure whether this is a corner case that Balsa should just handle in place, or whether it's a protocol violation for that PREAUTH response to be the first response from the server. If the latter, the response would most likely be different.


[@pawels](#), any thoughts?

 Albrecht Dreß @albrecht 2 years ago

Developer

whether it's a protocol violation for that PREAUTH response to be the first response from the server

No, no corner case, but "[...] one of three possible greetings at connection startup" (see [RFC 3501, Sect. 7.1.4](#), I'll look into that...

 Albrecht Dreß @albrecht 2 years ago


Developer

Could you please try the attached patch (works for both the master and gmime3 branches)? It fixes the issue for me when trying to access a local dovecot instance in PREAUTH mode.

Apart from checking against a non-NULL `last_msg`, the `PREAUTH` reply must be parsed like `OK`.

Opinions?

[📄 fix-issue-23.diff](#)

 Peter Bloomfield @peterb 2 years ago

Developer


Thanks for the quick patch—looks good to me!


I don't have a handy server that would raise the issue, but the fix looks good. Parsing is needed because the rest of the `PREAUTH` greeting may carry the same information as the `OK` greeting, is that right?


Will push master branch and rebase gmime3.

Peter

Please [register](#) or [sign in](#) to reply

 Albrecht Dreß closed via commit [4e245d75](#) 2 years ago

 Albrecht Dreß mentioned in commit [4e245d75](#) 2 years ago

 Damian Poddebniak mentioned in issue [#37 \(closed\)](#) 2 years ago

Please [register](#) or [sign in](#) to reply