

SECURITY: mutt_decode_uuencoded() can read past the of the input line


Hello, In `mutt_decode_uuencoded()`, the line length is read from the untrusted uuencoded part without validation. This could result in including private memory in message parts, for example fragments of other messages, passphrases or keys in replies.

Reproduce with the following mbox, note that these are literal 0x9f bytes. This should show some uninitialized garbage in the message.

```
From tavis@ Thu Mar 31 16:53:55 2022
From: tavis@
Subject: mutt_decode_uuencoded test
Content-Disposition: inline
Content-Transfer-Encoding: x-uuencode
Content-Type: text/plain

begin 644 test
<9f>
M2&5L;&\L"@I)9B!Y;W4@87)E(')E861I;F<@=&AI<R!M97-S86=E(&EN(&UU
M='0L('1H92!N97AT(&QI;F4*<VA0=6QD(&-O;G1A:6X@9V%R8F%G92X*"@H*
<9f>
54&QE87-E(')E<QY+'I4879I<RX*
`
end.
```

Edited 7 months ago by [Tavis Ormandy](#)

 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

Link issues together to show that they're related or that one is blocking others. [Learn more](#).

Activity

 [Tavis Ormandy](#) changed title from **SECURITY: mutt_decode_uuencoded() can read the past the of the input line** to **SECURITY: mutt_decode_uuencoded() can read past the of the input line** [7 months ago](#)



[Kevin J. McCarthy](#) [@kevin8t8](#) · 7 months ago

Maintainer

Thank you for opening the confidential issue. I'm a bit short on time tonight, but I will work on this tomorrow.

Taking a quick peek at the code, I do see the issue you have reported. It looks like a few things could be tightened up to make it a bit more defensive. Again, I will work on this tomorrow and get a release out this week with a fix.



[Tavis Ormandy](#) [@tavis@](#) · 7 months ago

Author

No problem, thanks Kevin!



[Tavis Ormandy](#) [@tavis@](#) · 7 months ago

Author

It looks like this bug has been there since the original uuencode support in 1998. I don't think `Content-Transfer-Encoding: x-uuencode` ever caught on, I wonder if anyone has ever used it legitimately? What do you think about trying to remove it?

It would be a bit less attack surface when parsing messages, and one less thing to maintain. I totally understand that there probably are old comp.unix.sources posts with uuencoded archives on them, and someone might legitimately want to read those with mutt -- That doesn't matter, because they won't have Content-Transfer-Encoding set, so mutt won't decode them anyway. If you remove native mutt uudecoding, that doesn't stop anyone piping to uudecode, which they would have to do anyway...

Just a suggestion from someone who likes removing attack surface! 😊

Kevin J. McCarthy @kevin8t8 · 7 months ago

Maintainer

I think uuencode was completely supplanted by MIME and Base64 encoding, so in principal I agree it would be better to just remove it from Mutt. Before anything like that happened, it would need a discussion on mutt-dev. However, this is complicated by the fact that I'm now only maintaining stable/security releases (2.2.x); right now there is no active development maintainer.

For a stable-security fix, of course, we have to fix the problem. I've looked a bit more at the code this morning. I was sorely tempted to add in a bunch more error-checking, but after thinking about it, I'll only address the security problem. The more I touch the code the more chance of inadvertently causing an accidental new bug or a regression - the current decoder is quite lax. I'm testing a patch this morning and will post it here later today.

Kevin J. McCarthy @kevin8t8 · 7 months ago

Maintainer

I'm testing a minimal patch right now:

```
modified handler.c
@@ -404,9 +404,9 @@ static void mutt_decode_uuencoded (STATE *s, LOFF_T len, int istext, :
     pt = tmps;
     linelen = decode_byte (*pt);
     pt++;
-   for (c = 0; c < linelen;)
+   for (c = 0; c < linelen && *pt;)
     {
-       for (l = 2; l <= 6; l += 2)
+       for (l = 2; l <= 6 && *pt && *(pt + 1); l += 2)
         {
             out = decode_byte (*pt) << 1;
             pt++;
         }
     }
 }
```

Again, there is a lot of room for more robust checking in this function (e.g. validating linelen is <= 45; checking if input lines are larger than SHORT_STRING; checking decode_byte() returns a value from 0-63). However, I believe all that just prevents "gibberish" output, not an actual security issue.

Tavis Ormandy @taviso · 7 months ago

Author

Thanks Kevin, looks good to me, I agree the other issues are not security problems.

Kevin J. McCarthy @kevin8t8 · 7 months ago

Maintainer

Just an update that I'm waiting for a CVE ID. I used the gitlab tool for the first time for this, so I've been struggling a bit with it. I think the ball is in their court. If I don't hear from them soon I'll probably just release this weekend with a note about an upcoming CVE.

Kevin J. McCarthy mentioned in issue [#405 \(closed\)](#) 7 months ago

Kevin J. McCarthy @kevin8t8 · 7 months ago

Maintainer

This has been assigned CVE-2022-1328. I'll create a release in the next few hours. After that I will make this and 405 publicly accessible, and ask the CVE details to be published.

Thank you Tavis!

Tavis Ormandy @taviso · 7 months ago

Author

Sounds good to me, Thank you!

Kevin J. McCarthy @kevin8t8 · 7 months ago

Maintainer

This has been pushed up in commit [e5ed080c](#). I've cut the release tarball and will be announcing shortly.



[Kevin J. McCarthy](#) made the issue visible to everyone [7 months ago](#)



[Kevin J. McCarthy](#) closed [7 months ago](#)



[Renato Aguiar](#) mentioned in commit [renatoaguiar/openbsd-ports@24d300b2b9ce07d6e212820528619b6f770fb475](#) [7 months ago](#)



[Derek Schrock](#) mentioned in commit [vishwin/freebsd-ports@2e4ae7db](#) [7 months ago](#)



[Derek Schrock](#) mentioned in commit [vishwin/freebsd-ports@377603c4](#) [7 months ago](#)



[Derek Schrock](#) mentioned in commit [FreeBSD/freebsd-ports@fec517b6](#) [7 months ago](#)



[Maximilian Bosch](#) mentioned in commit [lama-corp/infra/mirrors/nixpkgs@db1a3b7d](#) [7 months ago](#)



[Lf](#) mentioned in commit [podiki/guix-mirror@31f42393e8fe3b84fc0b58eb16866302cdff27c9](#) [7 months ago](#)

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