Commit f547a84

kevin8t8 committed 2 weeks ago

```
Fix imap_auth_gss() security level size check and buf_size type.

Make sure send_token.length is 4 bytes before reading the data.

Fix the buf_size type to be uint32_t instead of long.  ntohl()
operates on, and returns, a 32 bit unsigned integer.  Most
architectures now use a 64-bit long.

I believe this only worked because in Little-Endian, the
least-significant bits come first, so even though we were using 8
bytes of send_token.value (4 of which were out of bounds) for the cast
to long, only the first 4 bytes were used to truncate to the uint32_t
that ntohl() used.  Likewise when we converted htonl() further down.

Additionally, the comments indicate that mutt wasn't using buf_size in
any case, so perhaps that also explains the lack of bug reports.

Thanks to evilrabbit@tutamail.com for the security report.

Reviewed-by: Alejandro Colomar <alx@kernel.org>
```

master · mutt-2-3-2-rel

1 parent fdc04a1 commit f547a84

1 file changed +10 -2

Top

Filter files…

imap

auth_gss.c

Search within code

imap/auth_gss.c

```
@@ -108,7 +108,7 @@ imap_auth_res_t imap_auth_gss (IMAP_DATA* idata, const
char* method)
108 108     int cflags;
109 109     OM_uint32 maj_stat, min_stat;
110 110     BUFFER *buf1 = NULL, *buf2 = NULL;
111     -   unsigned long buf_size;
    111 +   uint32_t buf_size;
112 112     int rc, retval = IMAP_AUTH_FAILURE;
113 113
114 114     if (!mutt_bit_isset (idata->capabilities, AGSSAPI))
@@ -259,6 +259,14 @@ imap_auth_res_t imap_auth_gss (IMAP_DATA* idata, const
char* method)
259 259     }
260 260     dprint (2, (debugfile, "Credential exchange complete\n"));
261 261
    262 +   if (send_token.length < 4)
    263 +   {
    264 +     /* TODO: convert to muttdbg() in master branch merge */
    265 +     dprint(2, (debugfile, "Truncated security level data\n"));
    266 +     gss_release_buffer(&min_stat, &send_token);
    267 +     goto err_abort_cmd;
    268 +   }
    269 +
262 270     /* first octet is security levels supported. We want NONE */
263 271   #ifdef DEBUG
264 272     server_conf_flags = ((char*) send_token.value)[0];
@@ -272,7 +280,7 @@ imap_auth_res_t imap_auth_gss (IMAP_DATA* idata, const
char* method)
272 280
273 281     /* we don't care about buffer size if we don't wrap content. But here it is
*/
274 282     ((char*) send_token.value)[0] = 0;
275     -   buf_size = ntohl (*((long *) send_token.value));
    283 +   buf_size = ntohl(*((uint32_t *) send_token.value));
276 284     gss_release_buffer (&min_stat, &send_token);
277 285     dprint (2, (debugfile, "Unwrapped security level flags: %c%c%c\n",
278 286                 server_conf_flags & GSS_AUTH_P_NONE      ? 'N' : '-',
```

Comments 0

Please sign in to comment.