

sagredo-dev / qmail Public

Code Pull requests Actions Security and quality Insights

# Commit 749f607

sagredo-dev committed 2 weeks ago

[security] Remote Code Execution via Shell Injection in qmail-remote TLS Error Handler

main (#42) · v2026.04.07

1 parent 06b79b3 commit 749f607

2 files changed +106 -24 lines changed

Top Settings

Filter files...

- README.md
- qmail-remote.c

2 files changed +106 -24 lines changed

Search within code

README.md

Navigation icons

@@ -28,8 +28,9 @@ This distribution of `qmail` puts together `netqmail-1.06` with the following pa

28	28	* Frederik Vermeulen's qmail-tls patch v. 20231230
29	29	implements SSL or TLS encrypted and authenticated SMTP.
30	30	The key is now 4096 bit long and the cert will be owned by vpopmail:vchkpw
31		- Patched to dinamically touch control/notlshosts/<fqdn> if control/notlshosts_auto contains any
32		- number greater than 0 in order to skip the TLS connection for remote servers with an obsolete TLS version (tx Alexandre Fonseca).
31		+ Patched to dinamically create _control/notlshosts/<fqdn>_ if control/notlshosts_auto contains any
32		+ number greater than 0 in order to skip the TLS connection for remote servers with an obsolete TLS

```

33 + version (tx Alexandre Fonceca for the original code and to Diep Pham for
    spotting a vulnerability).
33 34 The file update_tmprsdh was modified to chown all .pem files to vpopmail.
34 35 http://inoa.net/qmail-tls/
35 36 * Marcel Telka's force-tls patch v. 2016.05.15
    ↓
    ↑
265 266 - Thanks to Manvendra Bhangui for porting qmail-qfilter to his Indimail and
    to Andreas Gerstlauer for porting
266 267 to my qmail. [Pull request](https://github.com/sagredo-dev/qmail/pull/38)
267 268
268 - * qmail-remote auth method select
269 - - authentication on remote servers by qmail-remote can select the auth method
    even
270 - - when the first method advertized by the remote server is not locally
    available.
271 - - ([tx Pierluigi](https://www.sagredo.eu/qmail-notes-185/smtp-auth-qmail-tls-
    forcetls-patch-for-qmail-84.html#comment5058))
272 -
269 + * Pierluigi's qmail-remote auth method select
270 + authentication on remote servers by qmail-remote can select the auth method
    even
271 + when the first method advertized by the remote server is not locally
    available.
272 + ([More info here](https://www.sagredo.eu/qmail-notes-185/smtp-auth-qmail-tls-
    forcetls-patch-for-qmail-84.html#comment5058))
273 + ([PR here](https://github.com/sagredo-dev/qmail/pull/39))
273 274
274 275 Install
275 276 -----
    ↓

```

```

v qmail-remote.c
... @@ -1,4 +1,12 @@
1 + #ifdef TLS
2 + #include <ctype.h>
3 + #include <string.h>
4 + #include <fcntl.h>
5 + #include <unistd.h>
6 + #include <stdio.h>

```

```
7 + #include <limits.h>
1 8 #include <pwd.h>
9 + #endif
2 10 #include <sys/types.h>
3 11 #include <sys/socket.h>
4 12 #include <netinet/in.h>
@@ -395,28 +403,101 @@ void blast()
395 403 #ifdef TLS
396 404 char *partner_fqdn = 0;
397 405
406 + /*
407 + * Validate a fully qualified domain name (FQDN)
408 + * Rules enforced:
409 + * - Total length: 1-255 characters
410 + * - Labels separated by '.'
411 + * - Each label: 1-63 characters
412 + * - Allowed chars: [A-Za-z0-9-]
413 + * - Labels must not start or end with '-'
414 + * - FQDN must not start or end with '.'
415 + */
416 + int is_valid_fqdn(const char *s) {
417 +     size_t len = strlen(s);
418 +     int label_len = 0;
419 +
420 +     // Check total length
421 +     if (len == 0 || len > 255) return 0;
422 +
423 +     // Must not start or end with '.'
424 +     if (s[0] == '.' || s[len-1] == '.') return 0;
425 +
426 +     for (size_t i = 0; i < len; i++) {
427 +         char c = s[i];
428 +
429 +         if (c == '.') {
430 +             // Reject empty labels or labels longer than 63 chars
431 +             if (label_len == 0 || label_len > 63) return 0;
432 +             label_len = 0;
433 +             continue;
434 +         }
```

```
435 +
436 + // Allow only alphanumeric characters and hyphen
437 + if (!isalnum((unsigned char)c) && c != '-') return 0;
438 +
439 + // Label must not start with '-'
440 + if (label_len == 0 && c == '-') return 0;
441 +
442 + label_len++;
443 + }
444 +
445 + // Validate last label
446 + if (label_len == 0 || label_len > 63) return 0;
447 +
448 + // FQDN must not end with '-'
449 + if (s[len-1] == '-') return 0;
450 +
451 + return 1;
452 + }
453 +
454 + /*
455 +  * Create a file in control/notlshosts/<partner_fqdn>
456 +  */
457 + int create_notlshost_file(const char *pw_dir, const char *partner_fqdn) {
458 +     char path[PATH_MAX];
459 +     char fqdn_buf[256];
460 +     int fd;
461 +
462 +     // Check length before copying
463 +     if (strlen(partner_fqdn) >= sizeof(fqdn_buf)) return -1;
464 +
465 +     // Copy to local buffer
466 +     strcpy(fqdn_buf, partner_fqdn);
467 +
468 +     // Validate FQDN
469 +     if (!is_valid_fqdn(fqdn_buf)) return -1;
470 +
471 +     // Normalize to lowercase
472 +     for (char *p = fqdn_buf; *p; p++)
473 +         *p = tolower((unsigned char)*p);
474 + }
```

```

475 + // Build the full path
476 + if (snprintf(path, sizeof(path),
477 +     "%s/control/notlshosts/%s",
478 +     pw_dir, fqdn_buf) >= sizeof(path))
479 +     return -1;
480 +
481 + // Create the file
482 + fd = open(path, O_WRONLY | O_CREAT, 0644);
483 + if (fd >= 0) close(fd);
484 +
485 + return 0;
486 + }
487 +
398 488 # define TLS_QUIT quit(ssl ? "; connected to " : "; connecting to ", "")
399 489 void tls_quit(const char *s1, const char *s2)
400 490 {
401 491     /*
402     - touch control/notlshosts/<fqdn> if control/notlshosts_auto contains any
403     - number greater than 0 in order to skip the TLS connection for remote
404     - servers with an obsolete TLS version.
405     - Thanks Alexandre Fonceca
492 + Create control/notlshosts/<partner_fqdn> if control/notlshosts_auto
493 + contains any number greater than 0 in order to skip the TLS
494 + connection for remote servers with an obsolete TLS version.
495 + Thanks Alexandre Fonceca for the original code.
496 + Thanks to Diep Pham for spotting the vulnerability.
406 497     */
407     - unsigned long i = 0;
408     - if (control_readint(&i, "control/notlshosts_auto") && i) {
409     -     struct passwd *info = getpwuid(getuid()); // get qmail dir
410     -     FILE *fp;
411     -     char acfcommand[1200];
412     -     sprintf(acfcommand, "/bin/touch %s/control/notlshosts/'%s'", info->pw_dir,
413     -         partner_fqdn);
413     -     fp = popen(acfcommand, "r");
414     -     if (fp == NULL) {
415     -         out("Failed to run touch command ");
416     -         exit(1);
417     -     }
418     -     pclose(fp);

```

```
419 - }
498 + struct passwd *info = getpwuid(getuid()); // get qmail dir
499 + create_notlshost_file(info->pw_dir, partner_fqdn);
500 +
420 501 /* end skip TLS patch */
421 502 out((char *)s1); if (s2) { out(": "); out((char *)s2); } TLS_QUIT;
422 503 }
```

## Comments 0



Please [sign in](#) to comment.