LKML.ORG
Messages in this thread

- *First message in thread*
- *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - **Greg Kroah-Hartman**
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Greg Kroah-Hartman*
    - *Naresh Kamboju*
    - *Guenter Roeck*
    - *shuah*
    - *Jon Hunter*
        - *Greg Kroah-Hartman*

Patch in this message

- *Get diff 1*

From Greg Kroah-Hartman <>
Subject [PATCH 4.14 21/59] crypto: authenc - fix parsing key with misaligned rta_len
Date Mon, 21 Jan 2019 14:43:46 +0100

4.14-stable review patch. If anyone has any objections, please let me know.

From: Eric Biggers <ebiggers@google.com>

commit 8f9c469348487844328e162db57112f7d347c49f upstream.

Keys for "authenc" AEADs are formatted as an rtattr containing a 4-byte 'enckeylen', followed by an authentication key and an encryption key. crypto_authenc_extractkeys() parses the key to find the inner keys.

However, it fails to consider the case where the rtattr's payload is longer than 4 bytes but not 4-byte aligned, and where the key ends before the next 4-byte aligned boundary. In this case, 'keylen -= RTA_ALIGN(rta->rta_len);' underflows to a value near UINT_MAX. This causes a buffer overread and crash during crypto_ahash_setkey().

Fix it by restricting the rtattr payload to the expected size.

Reproducer using AF_ALG:

```
        #include <linux/if_alg.h>
        #include <linux/rtnetlink.h>
        #include <sys/socket.h>

        int main()
        {
                int fd;
                struct sockaddr_alg addr = {
                        .salg_type = "aead",
                        .salg_name = "authenc(hmac(sha256),cbc(aes))",
                };
                struct {
                        struct rtattr attr;
                        __be32 enckeylen;
                        char keys[1];
                } __attribute__((packed)) key = {
                        .attr.rta_len = sizeof(key),
                        .attr.rta_type = 1 /* CRYPTO_AUTHENC_KEYA_PARAM */,
                };

                fd = socket(AF_ALG, SOCK_SEQPACKET, 0);
                bind(fd, (void *)&addr, sizeof(addr));
                setsockopt(fd, SOL_ALG, ALG_SET_KEY, &key, sizeof(key));
        }
```

It caused:

```
        BUG: unable to handle kernel paging request at ffff88007ffdc000
        PGD 2e01067 P4D 2e01067 PUD 2e04067 PMD 2e05067 PTE 0
        Oops: 0000 [#1] SMP
        CPU: 0 PID: 883 Comm: authenc Not tainted 4.20.0-rc1-00108-g00c9fe37a7f27 #13
        Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS 1.12.0-20181126_142135-anatol 04/01/2014
        RIP: 0010:sha256_ni_transform+0xb3/0x330 arch/x86/crypto/sha256_ni_asm.S:155
        [...]
        Call Trace:
         sha256_ni_finup+0x10/0x20 arch/x86/crypto/sha256_ssse3_glue.c:321
         crypto_shash_finup+0x1a/0x30 crypto/shash.c:178
         shash_digest_unaligned+0x45/0x60 crypto/shash.c:186
         crypto_shash_digest+0x24/0x40 crypto/shash.c:202
         hmac_setkey+0x135/0x1e0 crypto/hmac.c:66
         crypto_shash_setkey+0x2b/0xb0 crypto/shash.c:66
         shash_async_setkey+0x10/0x20 crypto/shash.c:223
         crypto_ahash_setkey+0x2d/0xa0 crypto/ahash.c:202
         crypto_authenc_setkey+0x68/0x100 crypto/authenc.c:96
         crypto_aead_setkey+0x2a/0xc0 crypto/aead.c:62
         aead_setkey+0xc/0x10 crypto/algif_aead.c:526
         alg_setkey crypto/af_alg.c:223 [inline]
         alg_setsockopt+0xfe/0x130 crypto/af_alg.c:256
         __sys_setsockopt+0x6d/0xd0 net/socket.c:1902
         __do_sys_setsockopt net/socket.c:1913 [inline]
         __se_sys_setsockopt net/socket.c:1910 [inline]
         __x64_sys_setsockopt+0x1f/0x30 net/socket.c:1910
         do_syscall_64+0x4a/0x180 arch/x86/entry/common.c:290
         entry_SYSCALL_64_after_hwframe+0x49/0xbe
```

Fixes: e236d4a89a2f ("[CRYPTO] authenc: Move enckeylen into key itself")
Cc: <stable@vger.kernel.org> # v2.6.25+
Signed-off-by: Eric Biggers <ebiggers@google.com>
Signed-off-by: Herbert Xu <herbert@gondor.apana.org.au>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

```
---
 crypto/authenc.c |   14 +++++++++++---
 1 file changed, 11 insertions(+), 3 deletions(-)

--- a/crypto/authenc.c
+++ b/crypto/authenc.c
@@ -58,14 +58,22 @@ int crypto_authenc_extractkeys(struct cr
                return -EINVAL;
        if (rta->rta_type != CRYPTO_AUTHENC_KEYA_PARAM)
                return -EINVAL;
-       if (RTA_PAYLOAD(rta) < sizeof(*param))
+
+       /*
+        * RTA_OK() didn't align the rtattr's payload when validating that it
+        * fits in the buffer.  Yet, the keys should start on the next 4-byte
+        * aligned boundary.  To avoid confusion, require that the rtattr
+        * payload be exactly the param struct, which has a 4-byte aligned size.
+        */
+       if (RTA_PAYLOAD(rta) != sizeof(*param))
                return -EINVAL;
+       BUILD_BUG_ON(sizeof(*param) % RTA_ALIGNTO);

        param = RTA_DATA(rta);
        keys->enckeylen = be32_to_cpu(param->enckeylen);

-       key += RTA_ALIGN(rta->rta_len);
-       keylen -= RTA_ALIGN(rta->rta_len);
+       key += rta->rta_len;
+       keylen -= rta->rta_len;

        if (keylen < keys->enckeylen)
                return -EINVAL;
```