Have a secure RIOT

Eric Sesterhenn <eric.sesterhenn@x41-dsec.de>

2017

X41 D-SEC GmbH

Whoami

- Eric Sesterhenn
- Principal Security Consultant
- Pentesting/Code Auditing at X41 D-Sec
- Father of a daughter

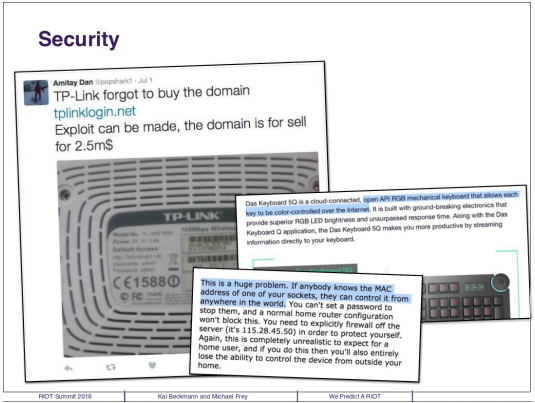
Disclaimer

- A lot of people know RIOT better than I do
- Intended to be positive criticism
- This is a spare time project for me
- Spent 2hr/week from June to mid August

"RIOT implements all relevant open standards supporting an Internet of Things that is connected, secure, durable, and privacy-friendly."

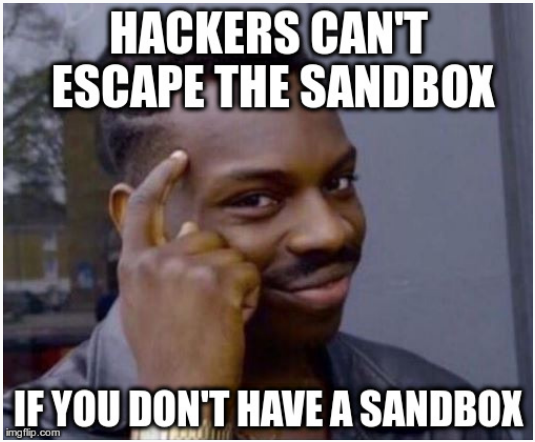
Motivation

- Good friend asked me ;-)
- Drunk in Warsaw... one thing leads to another



Secure Operating System

- Sandboxing - nah
- ACL - nah
- Fancy User Separation - nah
- DEP, ASLR... - nah



Secure Operating System

- Secure Language - nah
- C has tons of issues

What is already happening?

- Hard to find out, no Wiki page...
- Coding guide
- Static Analysis (cppcheck, clang)

Fuzzing

- Fuzzing just for SPIFFS
- In the dumbest version, throw random garbage at parsers
- Tools: American Fuzzy Lop, Libfuzzer, Radamsa, Hongfuzz...

Taking a closer look

Name	Issues reported
RIOT OS Core	3
Wakaama	3
Tiny-ASN1	5
SPIFFS	4
TinyDTLS	7
CCN-Lite	>15
OONF	3

Oneshot Malloc

- Classic Integer Overflow
- Fixed in glibc and others in 2002
- Used by MSP430

```
1  void __attribute__((weak)) *calloc(size_t
    ↪  size, size_t cnt)
2  {
3    void *mem = malloc(size * cnt);
4    if (mem) {
5      memset(mem, 0, size * cnt);
6    }
7    return mem;
8  }
```

Wakaama

- Type Confusion
- lwm2m_data_free() depends on the type
- Data reinterpreted as pointer and array size

```
1 targetP->type = LWM2M_TYPE_MULTIPLE_RESOURCE;
2 targetP = prv_extendData(targetP);
3 if (targetP == NULL) goto error;
4 ...
5 error:
6 lwm2m_data_free(size, *dataP);
```

Tiny-ASN1

- Should have been picked up by static analysis
- Bugs in error handling are common
- Crashes are different for IoT

```
1 if(encryption_algorithm_identifier == NULL
  ↪   ||
  ↪   encryption_algorithm_identifier->type
  ↪   != ASN1_TYPE_OBJECT_IDENTIFIER) {
2   fprintf(stderr, "ERROR: ...%d\n",
      ↪   encryption_algorithm_identifier->type
      ↪   );
3   return 1;
4 }
```

Randomness / TinyDTLS

```
1 dtls_ticks(&now);
2 #if (defined(WITH_CONTIKI) || defined(RIOT_VERSION))
3 /* FIXME: need something better to init PRNG here */
4 dtls_prng_init(now);
5 #else /* WITH_CONTIKI */
```

- Randomness infrastructure missing (just PRNGs exist)
- Guess this is known...

CCN-Lite

- Use with care...
- Rewrite is happening

```
int i, maxi, vallen;
int typ;
...
if (typ < NDN_TLV_MAX_TYPE &&
  ↪  ndntlv_recurse[typ]) {
   *len -= vallen;
```

```
struct key_s *k = (struct key_s *)
  ↪  calloc(1, sizeof(struct key_s*));
```

```
cp = ccnl_malloc(strlen(pending)+1);
strcpy(cp, pending);
```

OONF

- Properly checked, that no OOB Read oocurs?
- Whats missing?

```
1  if (*ptr + tail_len > eob){
2    /* not enough buffer for head */
3    return RFC5444_END_OF_BUFFER;
4  }
5
6  /* copy address tail into buffer */
7  memcpy(addr_entry->addr +
     ↪  tlv_context->addr_len - tail_len, *ptr,
     ↪  tail_len);
```

Lessons learned

- Port parts to secure language (lua?)
- Proper Audits
- Proper Fuzzing Infrastructure
- Use -m32 to find integer overflows easier
- Use asan, usan, msan and friends

Lessons learned

- GPG key for the security contact would be nice
- List with security issues/advisories (I didnt see a changelog entry, CVE-ID, ... about the issues reported)
- List the stuff that is happening
- Talk with upstream - just two of five upstream kept the CC

"Actually, I wasn't even aware [redacted] was a part of riot os until now"

Thanks

- Q & A
- eric.sesterhenn@x41-dsec.de
- Sorry no Twitter... stalk me on linkedin if you must