NFC ticket application

Oksana Baranova & Arthur Carels

Professor Tuomas Aura

December 2020

Memory lay-out

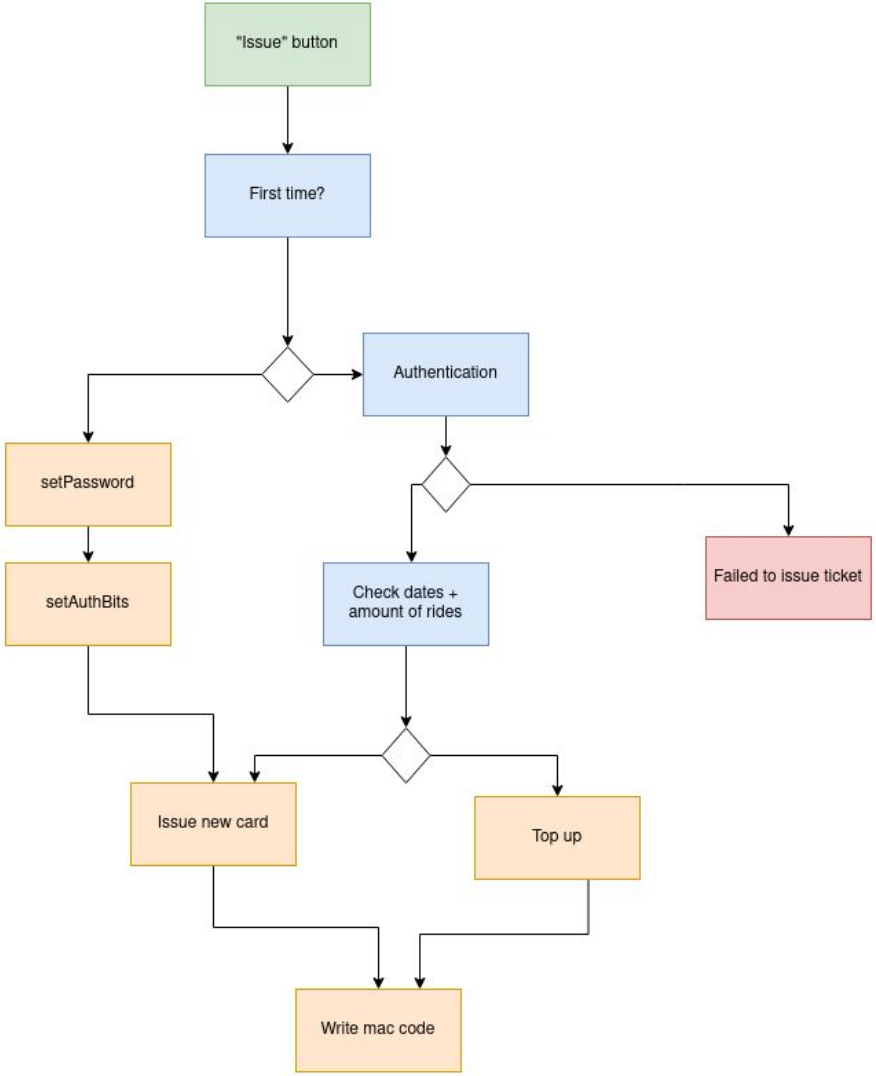
Total memory:

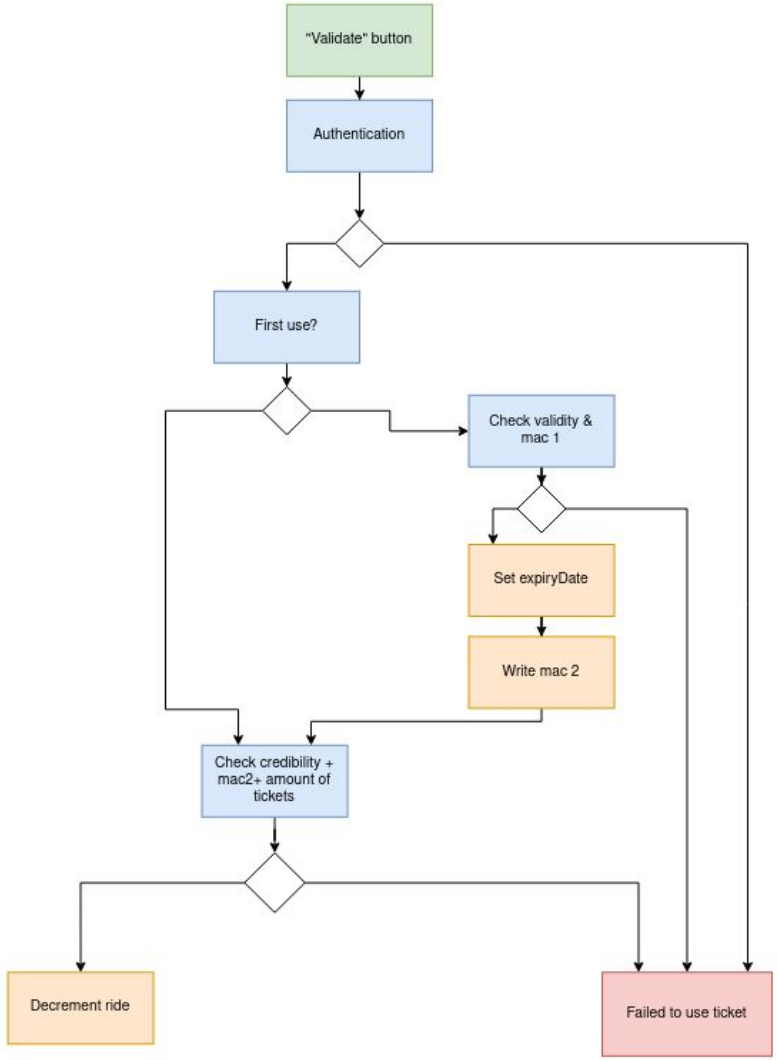
48 pages = 192 bytes

Free memory:

26 pages = 104 bytes

Page number	Content
1 - 3	UID + check bits
4	Application tag
5	Version Number
6	Initial counter
7	Ride counter
8 - 9	Validity date
10-11	Expiry date
12	MAC_1
13	MAC_2
14 - 39	Empty
40	Lock bits
41	16bit one-way counter
42	auth0
43	auth1
44 - 47	authentication key


"Issue" button
First time?
Authentication
setPassword
setAuthBits
Check dates + amount of rides
Failed to issue ticket
Issue new card
Top up
Write mac code


"Validate" button
Authentication
First use?
Check validity & mac 1
Set expiryDate
Write mac 2
Check credibility + mac2+ amount of tickets
Decrement ride
Failed to use ticket

Validity Date <-> Expiry Date

Validity Date:

- Updated on issue / topping up of unused card
- For bookkeeping reasons

Expiry Date:

- Set on first use of ticket
- Shorter than validity date!

Security measures

- Password:
  - Unique: hmac(master secret | UID)
- MITM prevention:
  - Using MAC
- Tearing protection
- Rollback prevention (monotonic counter)
- Replay attacks
- Cloning
- Passback prevention
- Checking safe limits
- No error messages on reader

→ hard for attacker to know what goes wrong

MAC codes

Truncate MAC to 4 bytes (= 1 page) → tearing protection

We use 2 MAC codes:

- Mac1:
  - Issuing ticket | Topping up card before first use
  - Create MAC of pages 6-9 (no expiry date)
- Mac2:
  - Written when first using ticket
  - Add new/updated expiry date into MAC
  - Create MAC of pages 6-11
  - Note: Also written when topping up a used card, but tearing protection is no concern then!

Security measures: Tearing prevention

Writing multiple pages → risk of tearing

- Only an issue when using card
- ↔ issuing is done by professional

Use of first ticket → write multiple pages

- Only consider it valid if counter has been updated (= last step!)

Demo + Questions