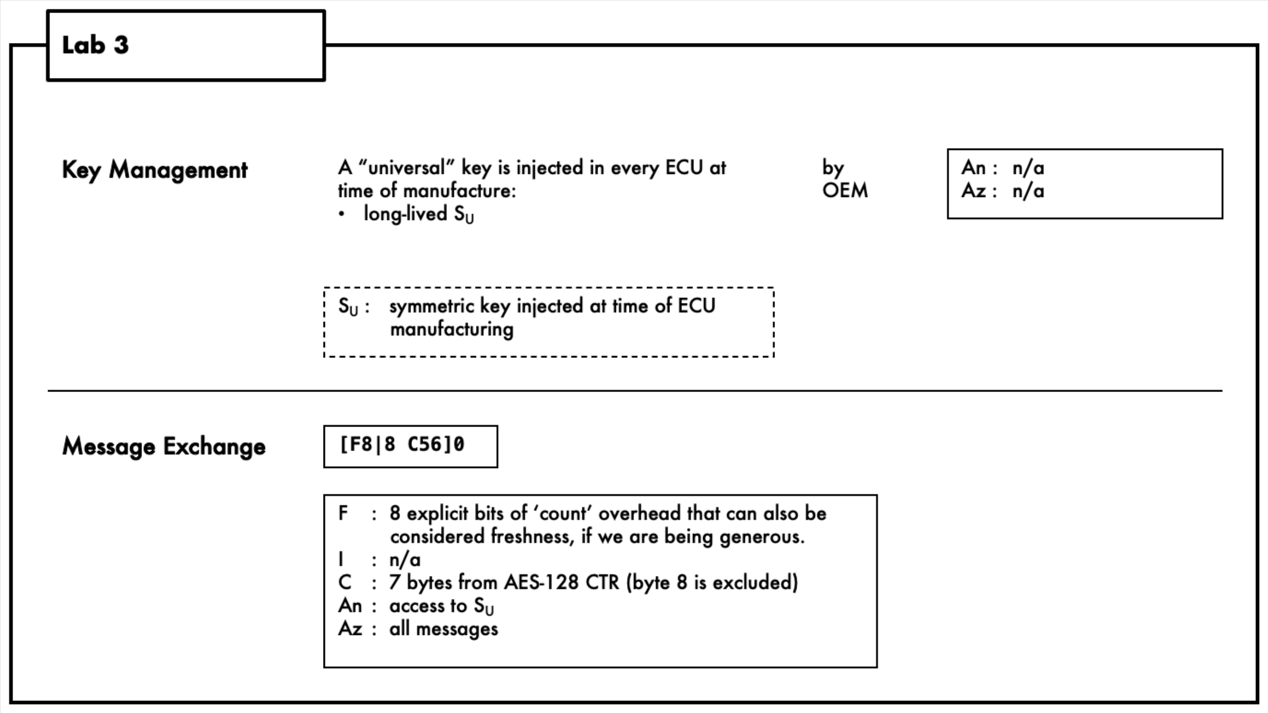

Lab 3

Key Management

A "universal" key is injected in every ECU at time of manufacture:

- long-lived S_U

by OEM

An : n/a
Az : n/a

S_U : symmetric key injected at time of ECU manufacturing

Message Exchange

[F8|8 C56]0

F : 8 explicit bits of 'count' overhead that can also be considered freshness, if we are being generous.
I : n/a
C : 7 bytes from AES-128 CTR (byte 8 is excluded)
An : access to S_U
Az : all messages

Scenario : The management team didn't like the idea of doubling the amount of data being sent on the bus, so they directed product engineering to find something better (less expensive in terms of bus capacity required). They (the product engineers) discover that AES can also be used as a stream cipher (AES-CTR mode). This means that they can get by with using only 8 bytes... with a little caveat. They need to "steal" one byte, say it is the last byte, to share some overhead that keeps the transmitter and receiver in sync.

You are told the team's use of AES-CTR looks like this:

```
IV = nonce || count
nonce is 15 bytes (all 0's)
count is 1 byte (monotonically increasing)
ct = AES-CTR(IV, pt)   // where pt is first 7 bytes
data in CAN frame = ct || count
```

You are also told the team didn't have time to develop code to enforce any security policies in 'adapterA'.

Exercise 1 : Review the network security framework, above. Comment on the feasibility of stealing a byte from every message sent on a J1939 bus.

Exercise 2 - Plan : [ATTACK] Prepare to attack the driver assistance operations by monitoring the bus. Make an estimate on how much traffic you need to see/log before you can launch your attack. Show your work. (note: if you feel motivated you can modify the 1Hz generator before doing your logging. If you modify the generator please comment on your reasons).

Exercise 3 - PoC : [ATTACK] Create a PoC using a MitM attack to allow driver assistance operations at any speed *without needing to exfiltrate* S_U.

Exercise 4 : [DEFEND] Propose a *much* stronger version of the security, still living with the constraints of having just 1 byte of overhead and using AES-CTR mode. Show this proposal using the message exchange notation. Implement your concept by updating 'adapterA' and 'adapterB'.

Did you introduce policy enforcement in 'adapterA'?