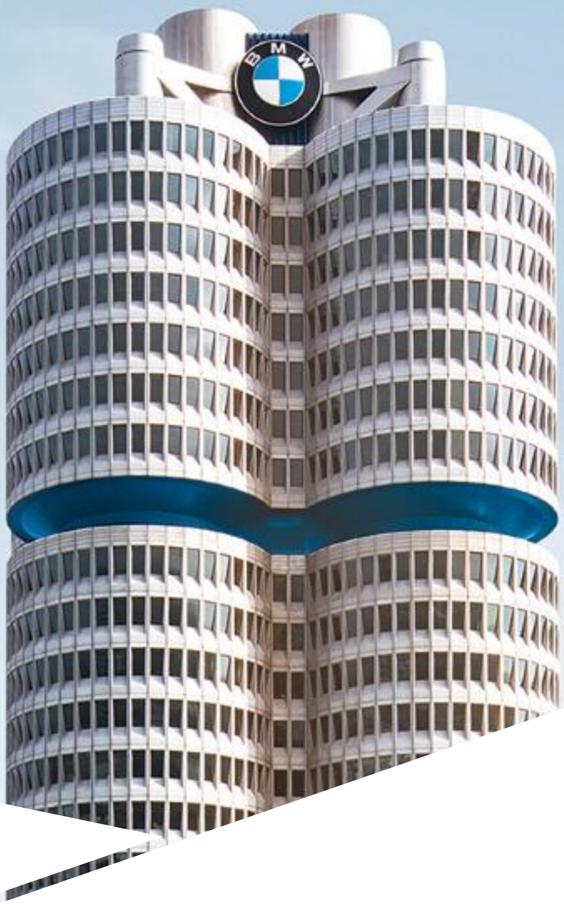
WHY IS NETWORK SECURITY IN VEHICLES SO HARD?

AUTOMOTIVE NETWORKS 2018.
IVN TECHNOLOGIES AND E/E ARCHITECTURE.



Dr. Lars Völker.
2018-11-06.

BMW GROUP

THE NEXT 100 YEARS

MOTIVATION.

YOU ONLY NEED TO INSTALL A STANDARDIZED NETWORK SECURITY SOLUTION!

JOB WELL DONE???

WHAT MAKES THE PROBLEM HARD?

WHAT MAKES THE PROBLEM HARD?

PRODUCTION.

STARTUP.

DRIVE.

And those are only selected aspects.


PRODUCTION: BUILDING A CAR.

BUILDING CARS (1)
FAST.

- We build cars at about 1 car/minute per assembly line.
  - That is about 2.5 million secure networks per year.
  - What other industries are producing this many?
  - Automation for some aspects is available (e.g. acme protocol and let’s encrypt). We need this for every aspect.

- What does this mean?
  - Everything needs to geared towards full automation:
    - Installing credentials/certificates? Fully automated!
    - Configuration? Fully automated!
    - Starting Network Security? Fully automated!

→ Build the secure networks fully automated.

BUILDINGS CAR (2) ROBUST.

- Now you can build a Secure Network per Minute and per assembly line.

- But what can still go wrong?
  - Your security process might not be robust enough and you interrupt production.
    - E.g. cars cannot be moved off the assembly line, if the car immobilizer needs to be set up first.
    - Stopping an assembly line means very high cost because of production units lost.
  - Connectivity to your central systems (e.g. PKI) is required but fails.
    - Again, stopping the assembly line is not an option.

- What do you have to do?
  - Plan the processes with enough robustness.
  - Make sure that connectivity is not a limiting factor.

→ Have processes and systems robust and distributed!

BUILDINGS CAR (3) UNTRUSTED.

- Lets assume you have developed a robust and fully automated process. You have put all the needed systems in all your plants and even lost connectivity can be overcome for a decent amount of time.

- What can go wrong?
  - You placed local systems (e.g. PKI systems) to achieve automated security setup.
  - But can you trust every plant?
  - Can you trust a plant run by a third party?
  - Can you trust third party final assembly plants?

- If you have not considered that earlier, you might need to adapt your plans again.

→ Design for an untrusted production environment!

STARTUP: STARTING A CAR.

STARTING A CAR (1)
WE DON'T HAVE TIME FOR THIS!

- How often are servers restarted? How long does this take? Would this be acceptable for a car?
  - More than 1 minute startup time is not uncommon. Even mobile phones are not starting up much faster.
  - Customers expect cars to startup instantaneous but power consumption does not allow for continuous standby.

- This means: A car needs to cold boot extremely fast compared to other systems!
  - If you open your car, you want the parking systems (ultra sonic sensors, cameras, display, etc.) to be there in less than 2s!
  - This includes: secure boot your systems, load credentials, start security protocols, exchange keys, start applications, etc.

- Did you consider the startup time of your security protocol?
  - Software, default configuration, hardware crypto (e.g. TrustZone)?

- You might even need to startup in both directions at the same time to speed up the process (think IPsec initiated by both peers).

→ Go for the fastest possible startup times (e.g. < 100ms)!

STARTING A CAR (2)
CHANGING ALL THE TIME.

- While the car is running, there are different situations:
  - The customer is driving from A to B.
  - The customer is sitting in the car, waiting for someone, and listens to music.
  - The customer has locked the car and it is not being used.
  - The electric car is being charged.
  - The customer wants to check the state of charge.

- To save energy, you want to limit the systems running to a minimum.
  - Customer listing to music but not driving? You might want to turn off the engine.
  - Customer checking the state of charge? You need to wake up but a few systems as possible to figure that out.

- Can you cope with that?

→ Plan for the car to have parts constantly going to sleep and waking up!

STARTING A CAR (3)
SCALING UP.

– Starting up Network Security for 2 systems is already a challenge. How about 50? 100? 150?

– Some protocols scale better than other…
  – TLS/DTLS: One startup per UDP and TCP connection. → n^3
  – IPsec: One startup per IP connection. → n^2
  – MACsec: One startup per Ethernet link. → n

– With only 20 Ethernet ECUs, this could mean up to (*):
  – 20 MACsec startups.
  – 800 IPsec startups (both directions for speed up).
  – 1600 DTLS/TLS startups or more. (Assuming UDP + TCP and both directions).

→ Make your solution scale for large networks with high connectivity.

DRIVE: USING THE CAR.

USING A CAR (1)
LIFETIME.

- Typical IT systems and the software on them are used a few years only. The same applies for mobile phones.

- Many cars are being used for 10-20 years.
  - Can you imagine the security of such old devices?
  - If you are using Certificates, did you remember that they might expire?

- A car model is produced over many years and many parts of the electronics might stay the same.
  - You need to design a product that might be produced for more than 10 years from now.

- Basically you need to design a product, for which <u>production starts a few years later</u>,
  a production that might be <u>produced for a very long time</u>,
  and a product that is <u>used by the customer for decades</u>.

→ Design for decades not years!

USING A CAR (2)
VALIDITY AND EXPIRATION.

- Certificates (e.g. x.509 based) define the “lifetime” of a certificate by NotBefore and NotAfter. The same might be true for other security artifacts.

- Did you think about the following things?
  - The car might be used longer than the default 2 years typically used in x.509 certificates.
  - During startup an ECU might not have a valid time and assumes the current date is e.g. 01.01.1970.
  - If something goes wrong, you might have a wrong time and date in an ECU. Can you handle this?
  - Did you test your libraries and code that it can run e.g. after the year 2050?
    - We found a bug in all OpenSSL implementations that a certificate with a “NotBefore” < 2050 and a “NotAfter” > 2050 were considered not valid by OpenSSL, if the current year > 2050.

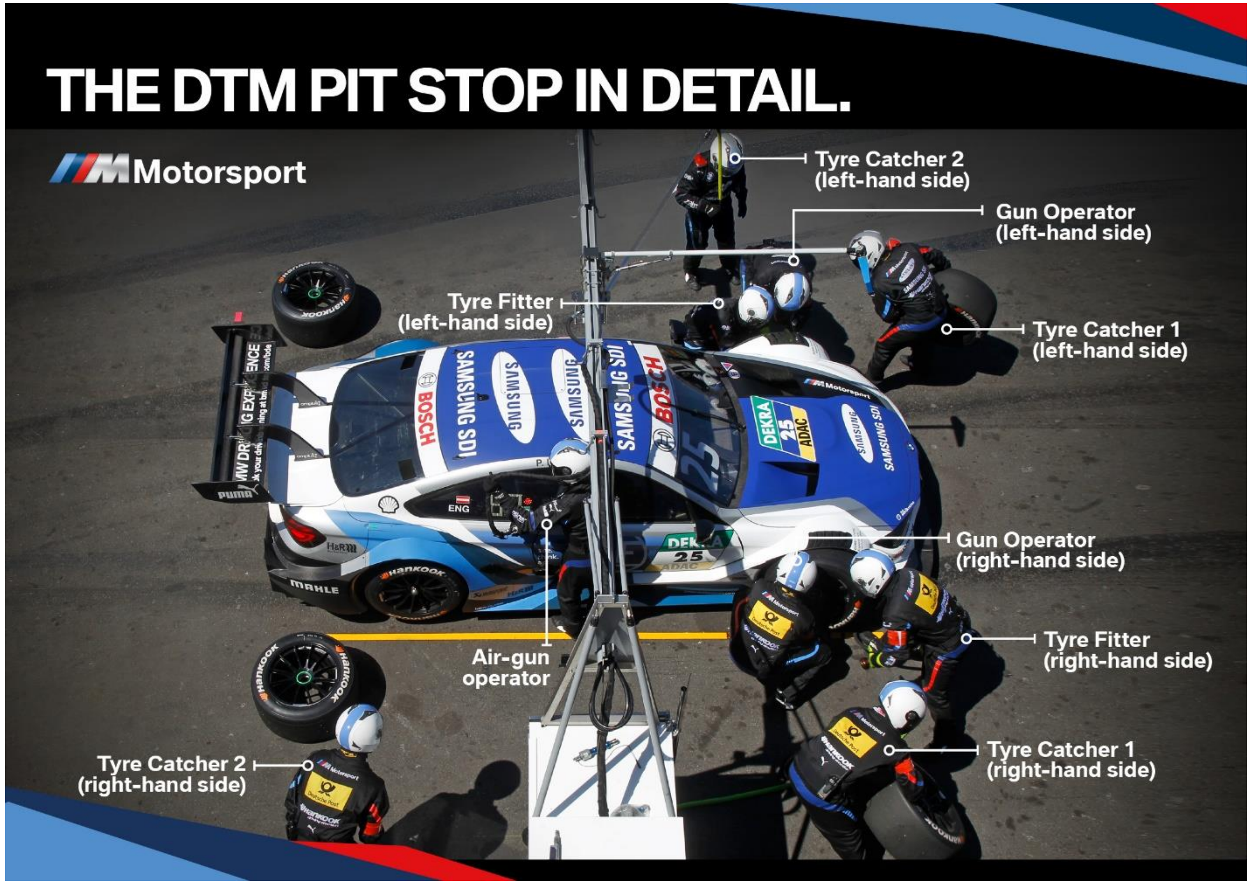
→ Double check your assumptions and test everything!

USING A CAR (3) SERVICE.

- Cars are being serviced all around the world.
  - Security professionals might not be present.

- You need fully automated processes to fix problems.

- Did you think about:
  - Replacing an ECU?
  - Updating Software?
  - Remote updating Software?
  - Activating new Features?



→ Design for service by people that don’t know about security!

CONCLUSION

CONCLUSION

→ Build the secure networks fully automated!

→ Have processes and systems robust and distributed!

→ Design for an untrusted production environment!

→ Go for the fastest possible startup times (e.g. < 100ms)!

→ Plan for the car to have parts constantly going to sleep and waking up!

→ Make your solution scale for large networks with high connectivity!

→ Design for decades not years!

→ Double check your assumptions and test everything!

→ Design for automated service by people that don't know about security!

→ If your solution is ready for IT, your work might have just started…

THANK YOU FOR YOUR ATTENTION.