Seminar session 4: Secure System Architecture – Debate: Formal modelling is the best way of ensuring a system is secure by design

Outline:

- Announcements → No announcements
- Reflection
- The importance of:
- Distributed System Future Trends
- Seminar 6 Questions & Debate

Reflection:

- Recommendation: Focus on one or two events
- Analyse of choice
- Multiple perspectives →
- E-Portfolio? → Can be implemented via a link (check over Turnitin should be made)
- Header and Anonymously → No anonymously and header do not count in the word count

The Importance of…:

- The Future of Distributed Systems
- → System Design
- → Fog, Osmotic & Edge

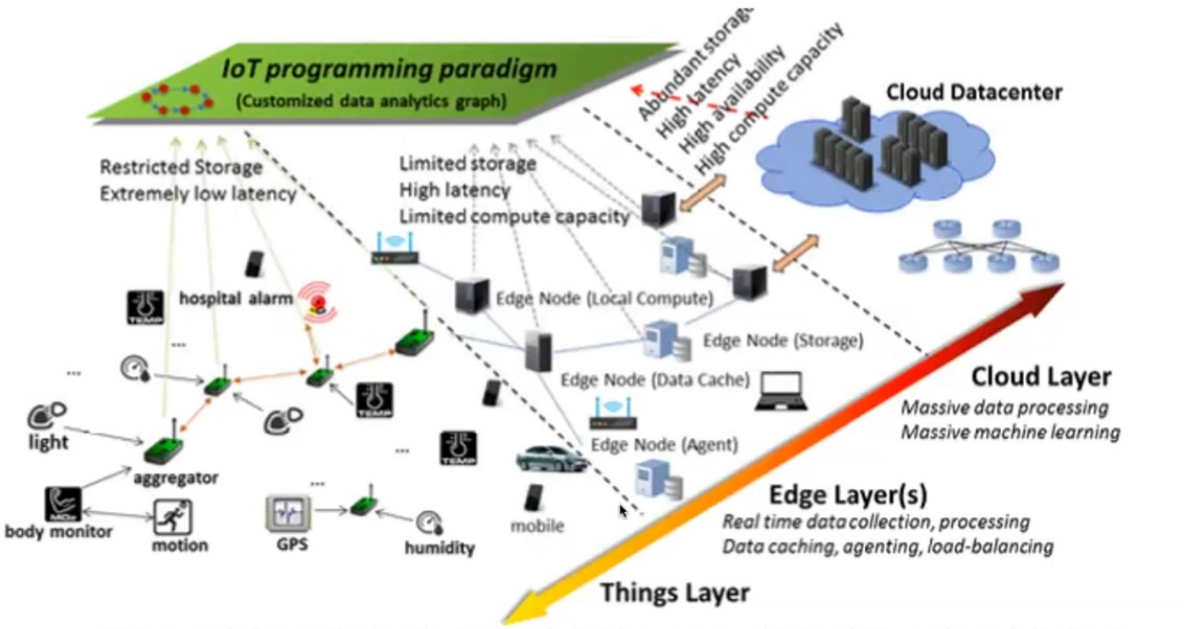

Figure 2: A typical IoT application infrastructure of a healthcare use case showing of Things, Edge, and Cloud layers

From https://rajivranjan.net/research-directory/internet-of-things-and-data-science/

- Distributed Hardware
- → IoT
- → Edge Servers
- → Cloud

➔ One of the big problems is the right system design. In a fog it could be differences around the different physical architectures. A solution could be TOSCA: a specification language with which hardware where the code runs can be defined.

- Challenges
- → Distributed Services
- → Design

TOSCA→

Seminar 3: Reading & Questions

Why Model Based Approach:

- Increasing size and complexity of software (Saifan & Dingel, 2008)
- Clearer requirements using UML (Saifan & Dingel, 2008)
- Formally defined languages to have an unambiguously defined semantics that is more difficult to misinterpret (Sainfan & Dingel, 2008)
- Enable analysis which may automatically analyse based on the model defined.
- Impact analysis on how topology changes affect security (Matthews et al., 2020)
- -> Combine Bayesian statics with attack graphs to automatically prioritise network vulnerabilities from a probabilistic view point, resulting in Bayesian Attack Graphs (BAGs)

- Lesser total cost

➔ I you find security risks/ issues in advance, it is less costly to fix it before development and less recourse as not many people involved (such as software/ hardware engineers, quality assurance engineers ect.)

➔ Case 1:
In CPS, vulnerabilities are particulary hard to fix after deployment because maintenance is not possible or the vulnerability is based on bad design decisions. Hence, it is important to consider secury in a design phase nad to refine these secure designs into deployed systems. Findning and preventing vulnerabilities is up to 30 times less expensive than fixing them in a deployed system (Cigatial Federal Inc., 2011).

Drawbacks:

- Skill required: which model, how to apply, how to control, tailoring, merging models
- Time required
- State space explosion: too many states, rules variables.
- Oracle automation: difficult to automate testing for complex system.

Final Thoughts:

➔ Formal modelling is the best way of ensuring a systems is secure by design? → No complete agreement and no complete disagreement, there are pros and cons, but it definitely helps building a more secure desing.

Reminders and Q&A:

- Unit 6: Submit assessment pt 2: Code
- Schedule Demos
- Submit individual Reflections