SecureCloud

Secure Big Data Processing in Untrusted Clouds

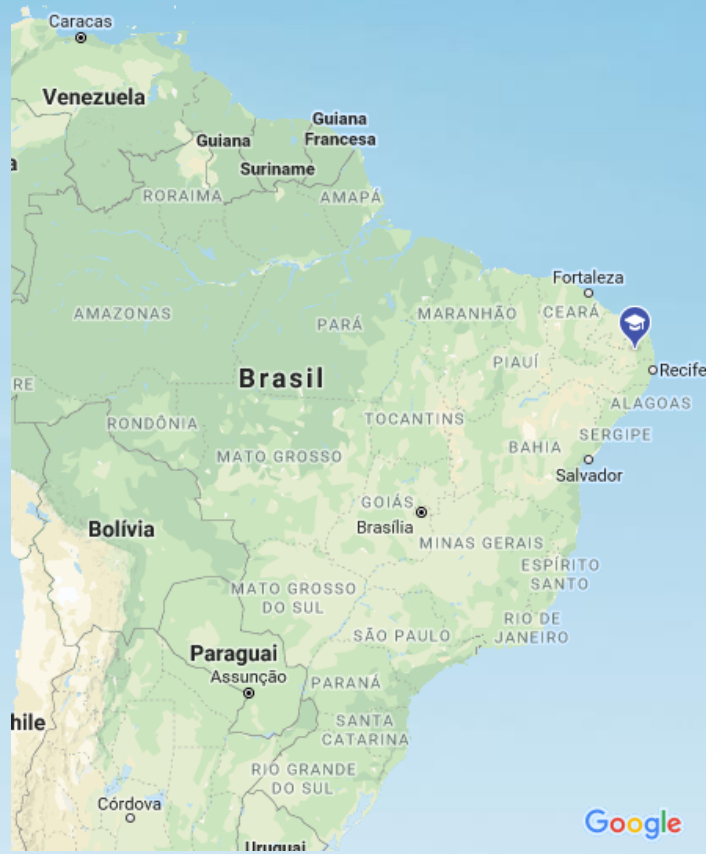
Secure and Privacy-aware Data Dissemination

Andrey Brito (andrey@computacao.ufcg.edu.br)

73rd IFIP Meeting, Goa, India, 16th Jan. 2018

About me

- Universidade Federal de **Campina Grande** (since 2011), Distributed Systems Lab
- R&D projects with industry around cloud computing, energy efficiency, and privacy/security
- H2020 projects (Cloud & Security/Privacy)
  - BigSea (2016-2018)
  - SecureCloud (2016-2018)
  - Atmosphere (2017-2019)



Motivation

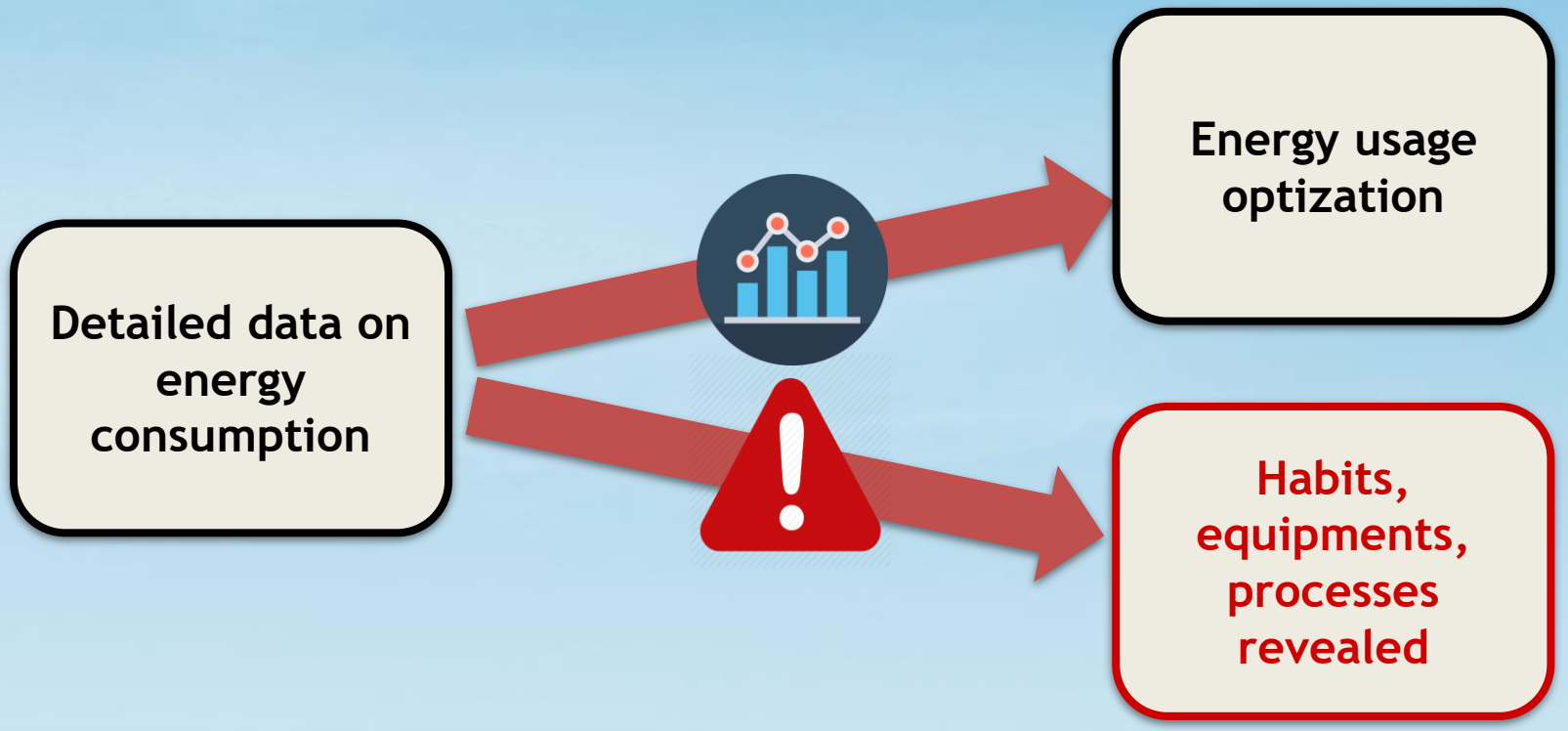
- Data leaks happen everyday (Equifax, Uber, Yahoo, Verizon, US Military, …), affecting millions of users at a time
- Concerns about security and data privacy are still major obstacles for cloud adoption
- The power of data lies in sharing
- Nice to hear: “'We simply don't want people to have to trust us,’ he said. ‘That's not what privacy is about.’"
  - https://www.theregister.co.uk/2017/09/27/signal_turns_to_intels_sgx_to_lock_down_contacts_from_spying_eyes/

My use case for today

- In generic terms…
  - Sensors producing sensitive data
  - Third parties offering services or interested in consuming the data
  - Users with different privacy concerns
  - Cloud is hosting data and/or applications

- This could be applied to health, GPS, or other smart-society data, but today the application is…

My use case for today: Smart Metering



Non-Intrusive Load Monitoring (NILM)

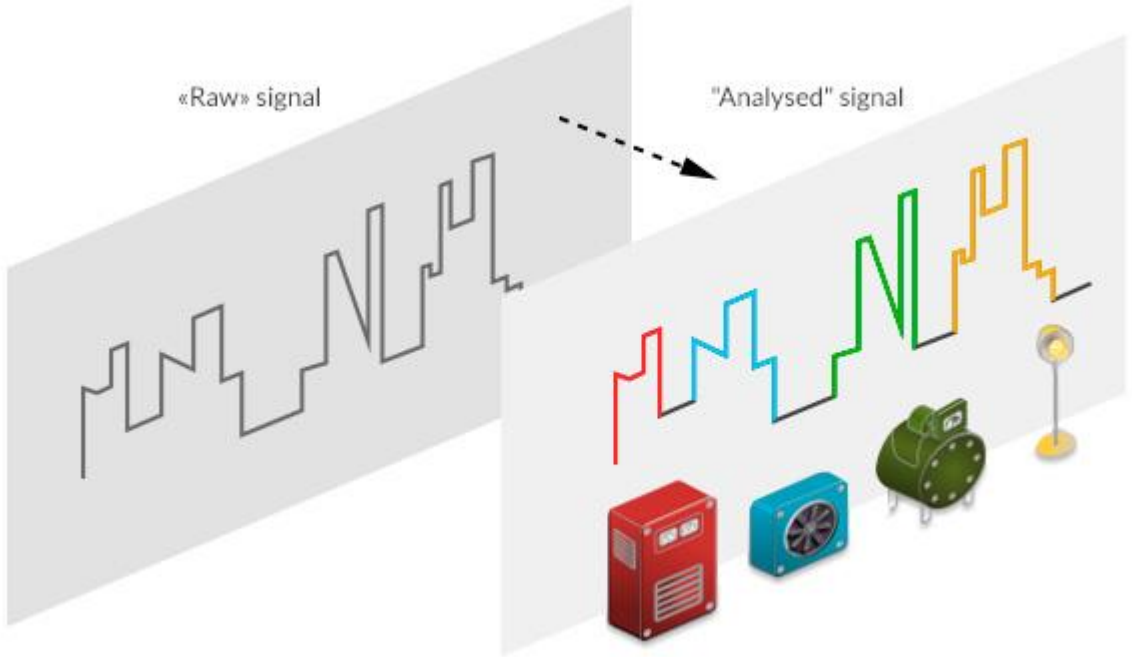


Image from: http://qualisteo.com/ws/en/the-wattseeker-solution/

Approach

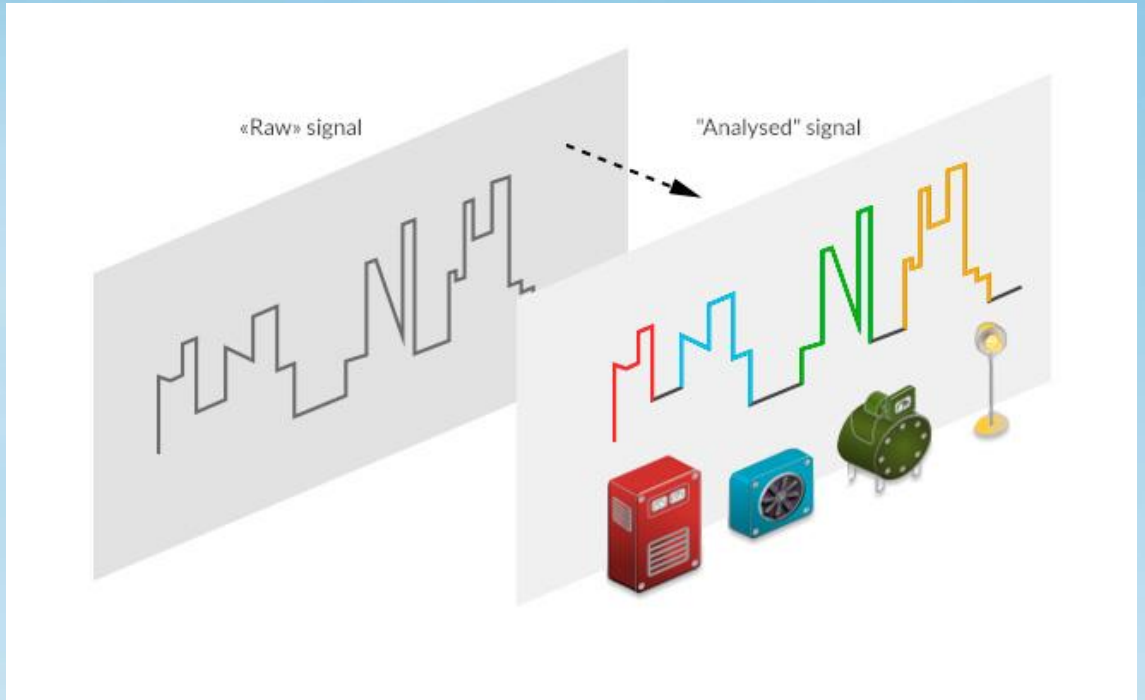
- Use Trusted Execution Environments (e.g., Intel SGX) combined with aggregation/obfuscation/anonymization of data to support privacy-aware dissemination

- Goals
  - Enable applications with different levels of security to get some data
  - Enable users to control levels of information shared with applications (“better” data to more trusted application)
  - Enable scalability by having agents representing the users

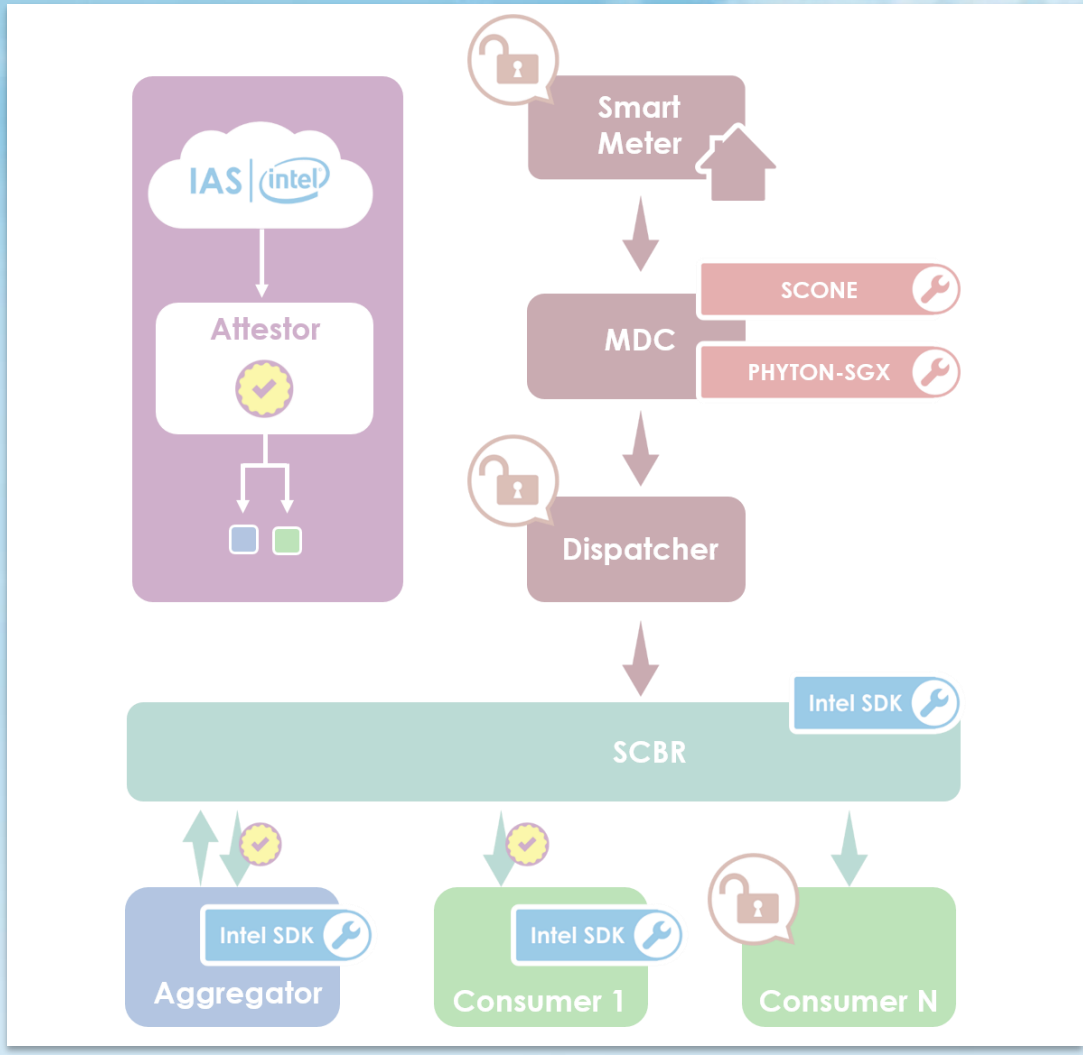
PART 1: DIFFERENT LEVELS OF DATA FOR DIFFERENT LEVELS OF TRUST

Example use case

- User visits a web site, finds a service that does NILM
- Decides to trust that service
- Enters the client ID for the power distribution company
- Goes to his meter, which now wants a confirmation that this consumer can be trusted

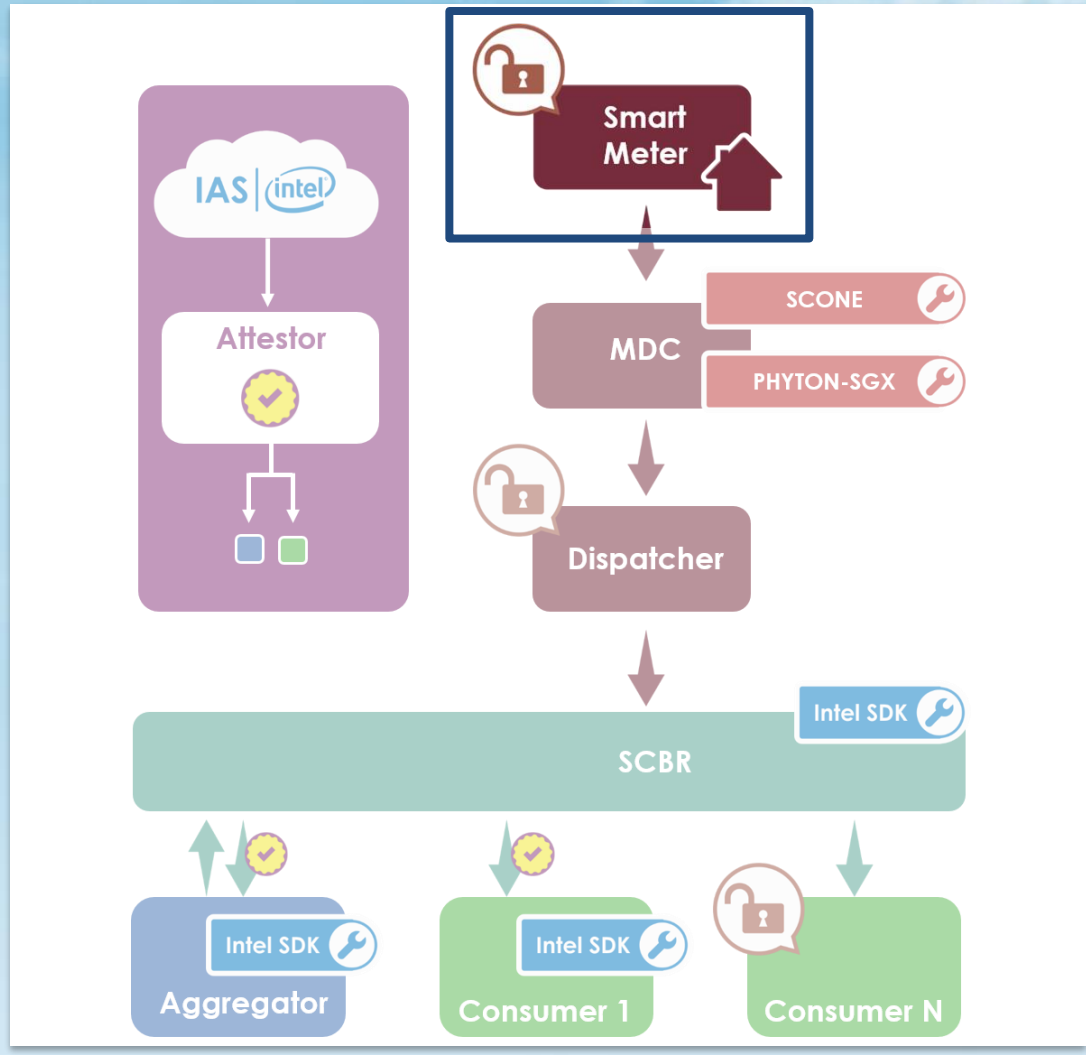


Example use case – Architecture



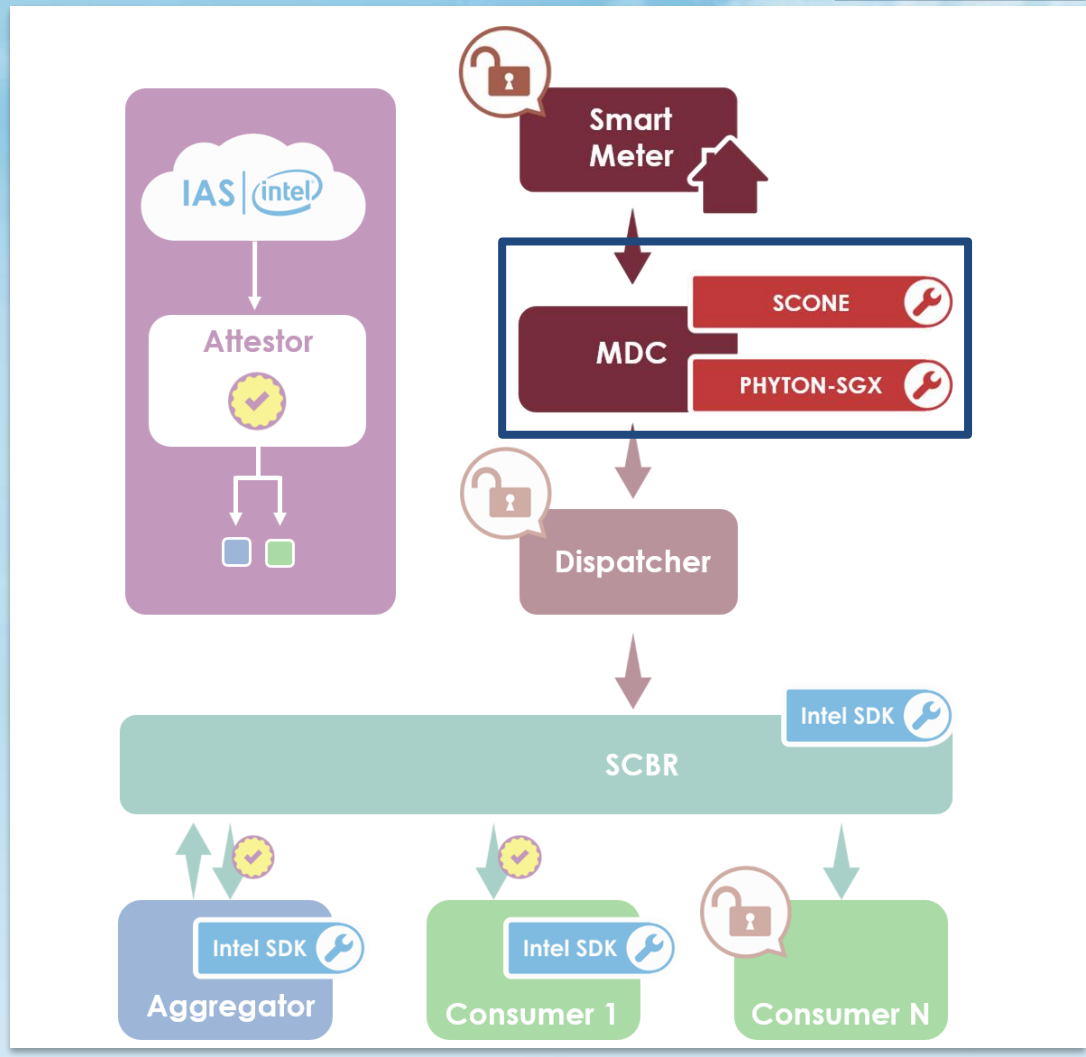
Example use case – Sensors

- Collect the measurements
- Are connected to the Internet
- Are under complete control of the user (or “are the user”), do not misbehave
- Send data to the a measurement proxy



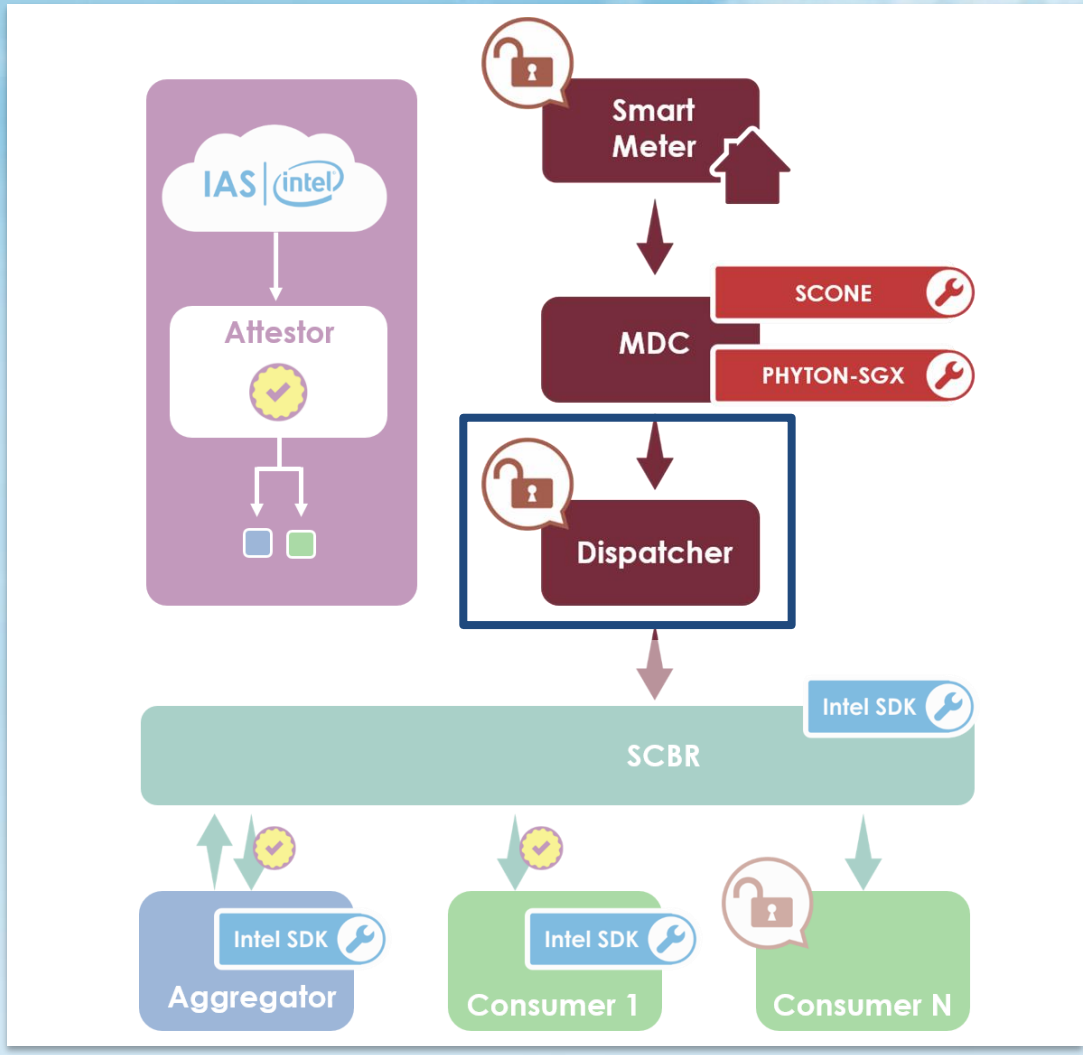
Example use case – Proxy (MDC)

- Trusted because of attestation (run inside TEE)
  - Maybe built with tools like SCONE (from TU Dresden), maybe Intel SDK
  - Had its code validated by the user, the community, or trusted parties
- Receives data from several meters, encrypt, send to the next component



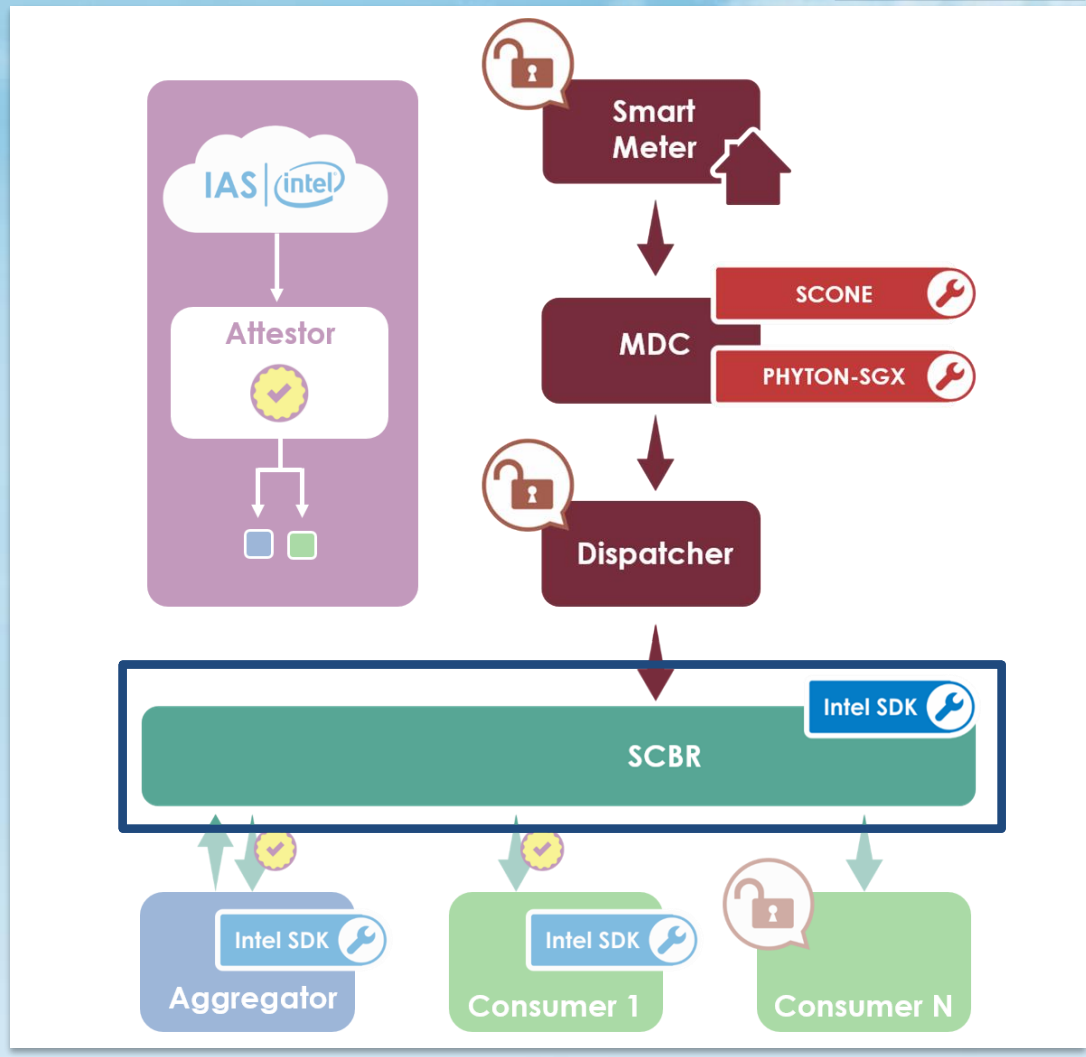
Example use case – Dispatcher

- Does not need to be trusted, handles only encrypted data
- Used for protocol conversion



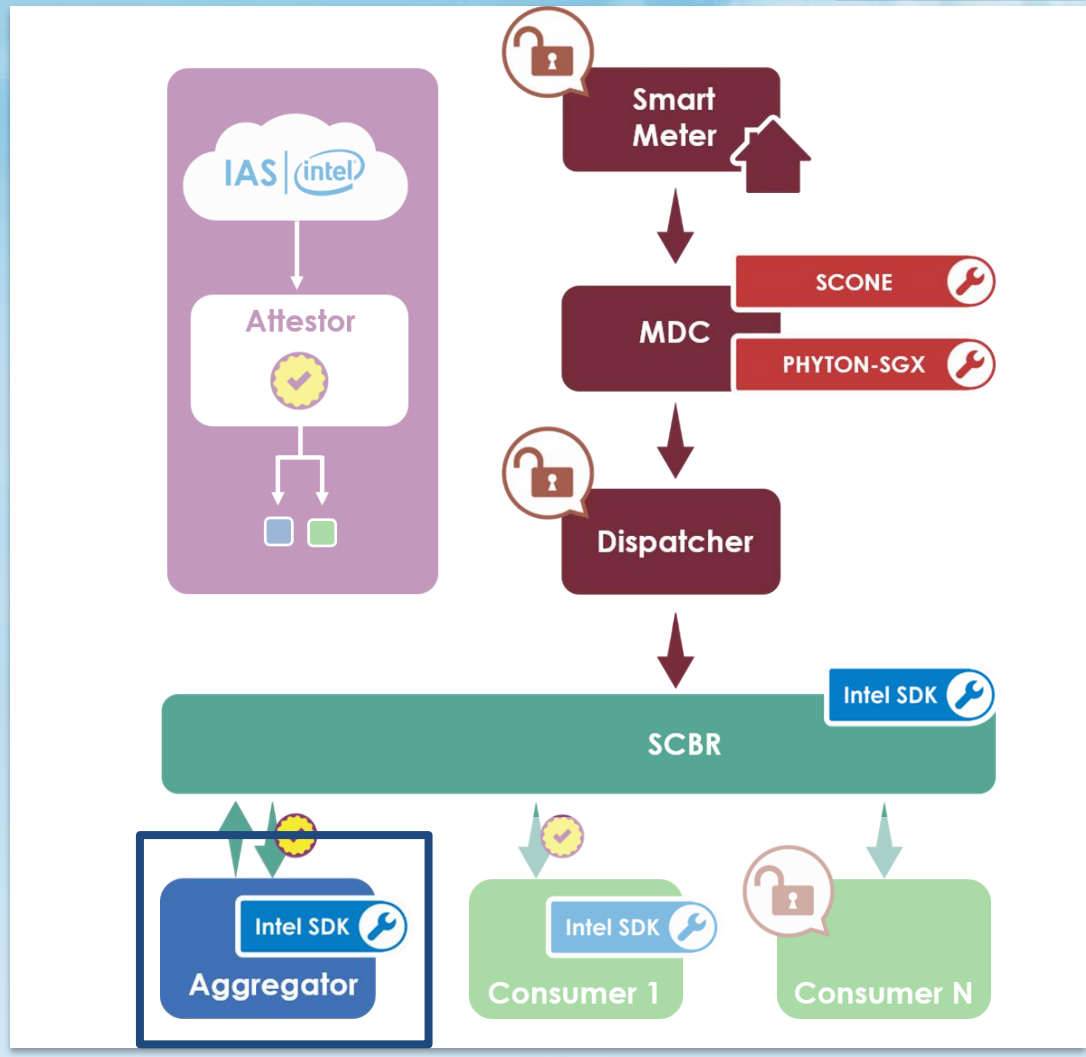
Example use case – SCBR

- Secure Content-Based Routing
  - One result of the project (UNINE)
  - Routing is done inside the TEE
  - Subscribers have to request registration to the publishers
- Sensitivity level is an attribute in the event published, it can be used to route messages



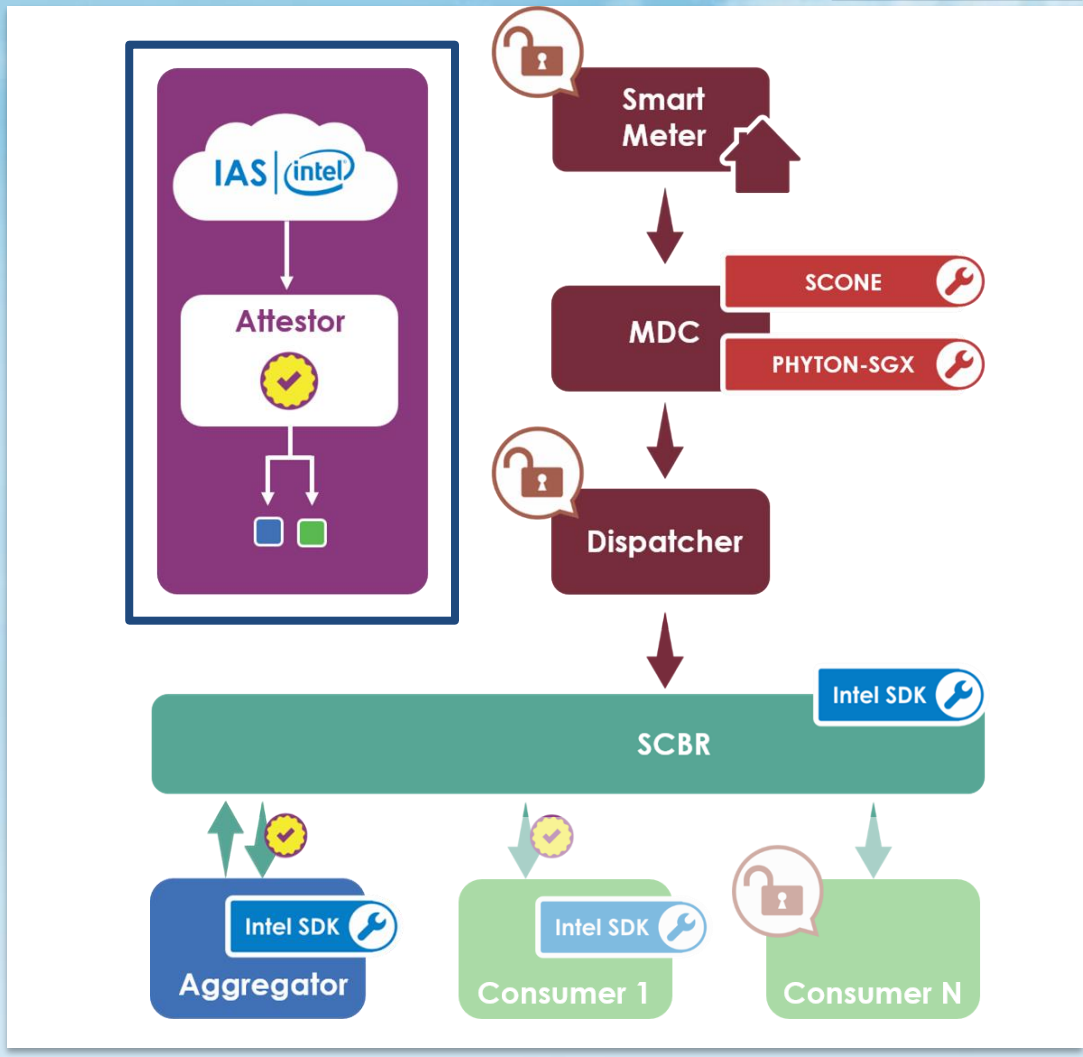
Example use case – Aggregator

- One example of trusted consumer
  - Has validated source code (by the user, by the community)
  - Was attested by the user when requested registration
- In this example, it receives sensitive events and generates events with less sensitive (e.g., aggregated) data



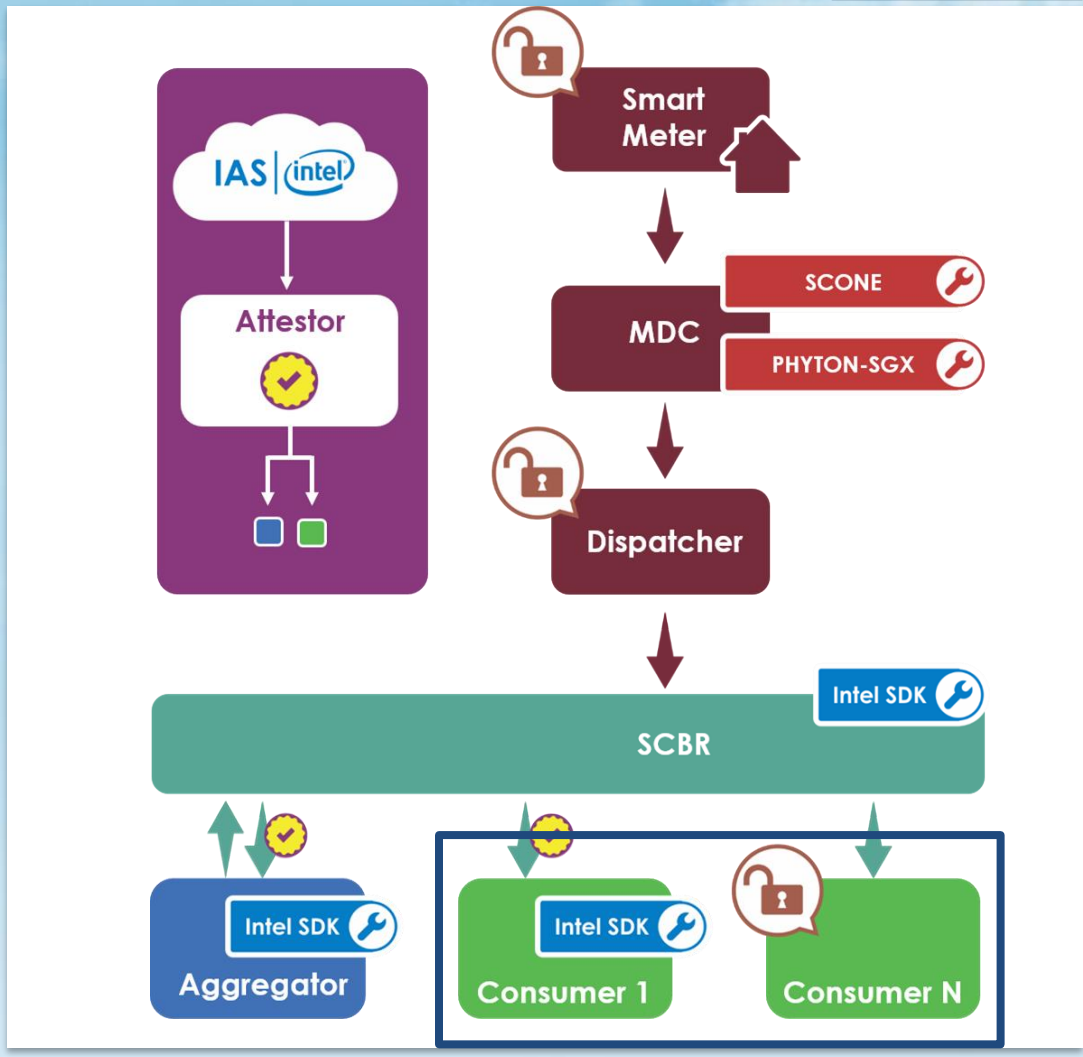
Example use case – Attestor

- Was used by the Aggregator to convince the Proxy/MDC that it runs genuine, trusted code
- May use technology manufacturer's attestations services (here, Intel Attestation Service)



Example use case – Consumers/Subscribers

- Trusted consumer/subscriber
  - Registered to the low level data
  - Was explicitly trusted by the user
- Untrusted consumer/subscriber
  - Registered to public data
  - Consumes data that, for example, has been previously aggregated or anonymized



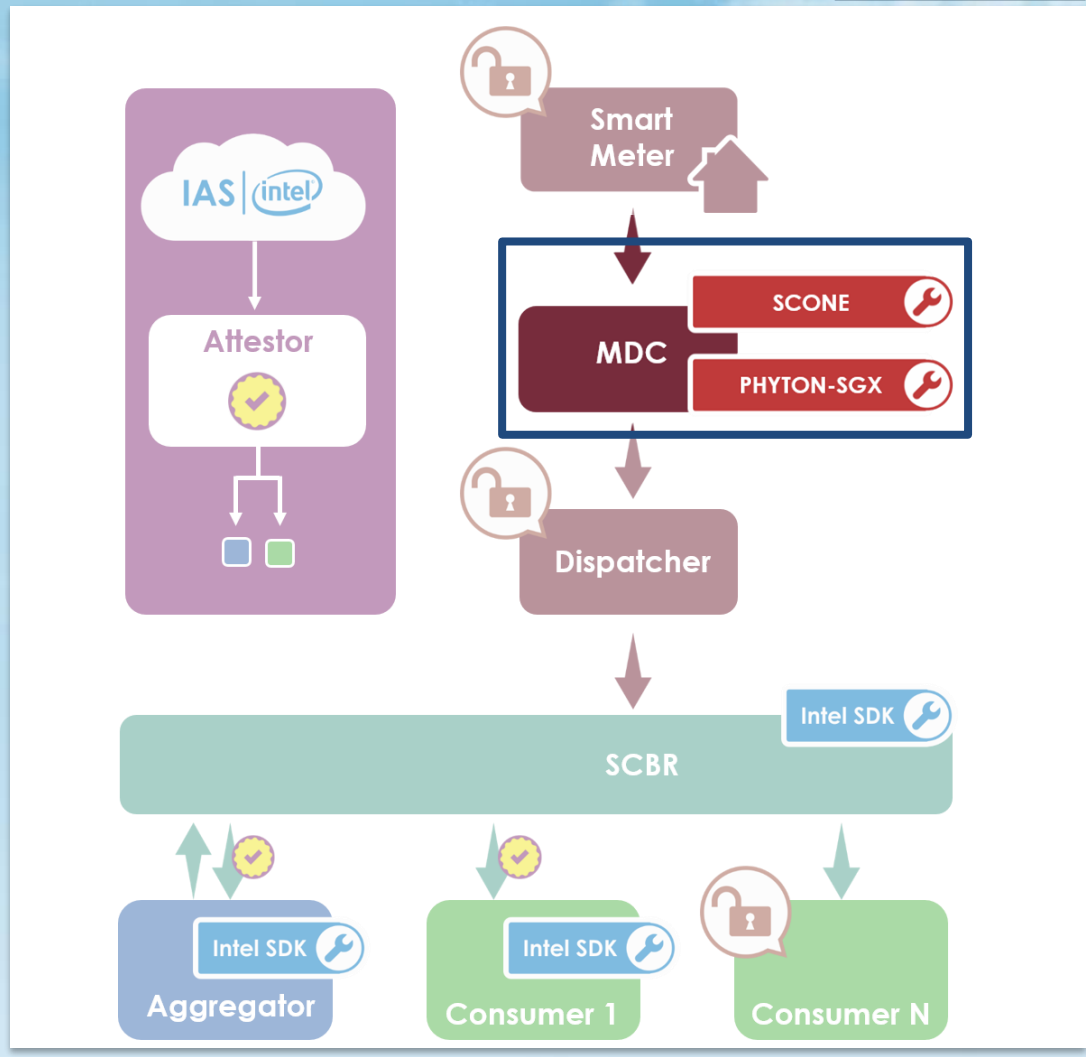
PART 2: SCALING THE SYSTEM BY REPRESENTING THE USER

Example use case

- What if now the users is associated to many data sources?
  - Produces home data, health data, movement data
  - Is identified by have its cell phone/face/car plate recognized & tracked
- Worse: Everywhere the user goes, some system wants to check how to better interact with him/her
  - Should the user receive real time sales adds in a shopping mall?
  - Should the user receive “motivating messages” when approaching a restaurant?
  - Should the salesperson in a store recognize him/her by the name and also see recent purchase history?
- Some users will want some of those features, some users will not

The proxy can decide…

- In our example, the proxy could decide if a new subscriber may have access to information, based…
  - On explicit authorization by the user
  - On the "trustworthiness" metrics of the requesting application
  - On the social network of the user
  - On what the user typically does…



Conclusion

- We start to see technologies to make TEEs practical (e.g., SGX), once ubiquitous, these technologies simplify many problems related to confidential data processing and, thus, will enable novel applications
- The approach discussed helps to ensure that entities interested in data can have access only to details compatible with user's trust on them
- But there are many, obvious challenges
  - How to define the sensitivity level of the information? (Easier for the smart grids, but very difficult to many others domains)
  - How to validate the code? (One more unhelpful "click ok to continue"?)
  - How to avoid overloading the user? (Proxy deciding by him?)

SecureCloud project is funded by the 3rd EU-Brazil coordinated call within the Horizon 2020 program.

European Commission
Horizon 2020

Brazil
Federal Government
MCTIC – RNP – CTIC

Swiss Confederation
State Secretariat for Education,
Research and Innovation