Hierarchical Storage for Building Management System

Zhehao Wang[1], Jiayi Meng[2], Jeff Burke[3]
[123]UCLA REMAP

Introduction

Building management system (BMS) is a sensor data acquisition system which automatically manages a building's heating, ventilation and air conditioning, and other systems.
An NDN based BMS leverages the architecture's advantages in hierarchical data naming and name-based routing and forwarding, in-network caching, and inherent security support, and may overcome the challenges IP faced, namely complexity of network addressing and configuration, reliance on middleware, and a lack of security.
This summer's work focuses on the data aggregation and signing/verification in NDN BMS, and updates the previous work by Wentao [1].

Objectives

- Provide campus, building, and department-level monitoring and query possibility
  - Data hierarchical storage and aggregation
    Design a hierarchical storage approach and a stream-based approach to calculating aggregates, distributing processing and taking advantage of local storage.
  - Data signing and verification

Design Overview

Figure 1 illustrates the namespace of the BMS application;

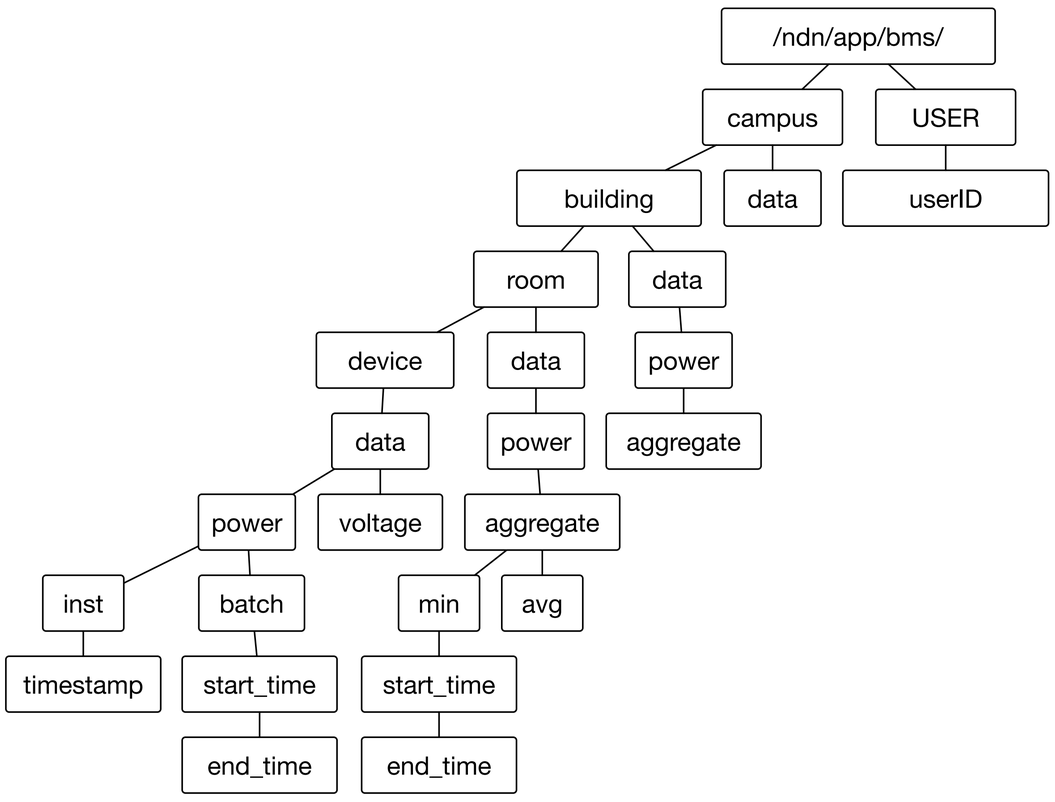

Figure 1: BMS namespace

- Physical location branch represents the hierarchical structure of BMS data: Campus - Building - Room - Device
- User branch records the list of BMS user identities, which can be used for access control

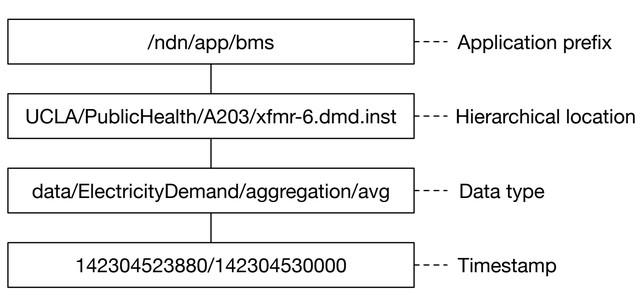

Figure 2: Example BMS data name

Figure 2 illustrates an example name of a piece of BMS data.

Data Aggregation

- Leaf nodes publish aggregated data at fixed time window.
- Non-leaf aggregate the data after all children respond, and publish data with the same time window.
- Long lived interests with received start time excluded moves data across layers.

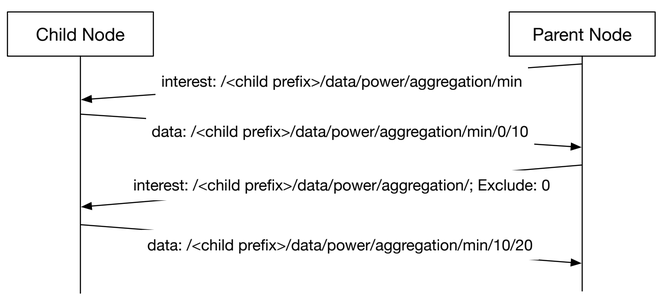

Figure 3: BMS move aggregation sequence

Trust Schema

Signing and Verification

- BMS data is verified hierarchically
- The certificates of BMS children node should be signed by their parent nodes in the tree
- Campus certificate is the root of trust for BMS data and user.

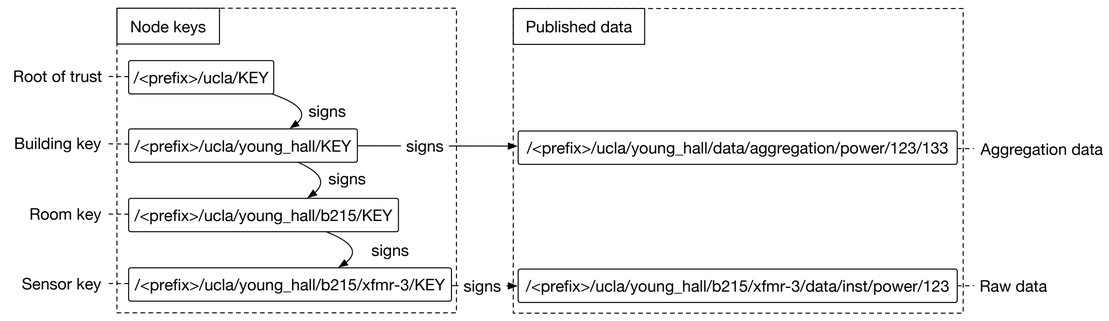

Figure 4: Example of signing in BMS

Bootstrap

- Child node obtains a signed certificate from its parent.
- Bootstrapping process from ndn-pi, illustrated in Figure 5, can be used for this process.

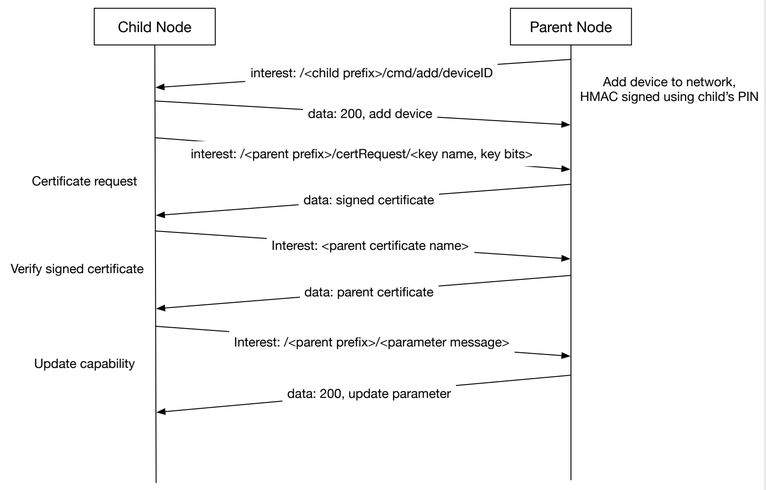

Figure 5: BMS add child sequence

Demo Implementation

We run BMS aggregation nodes in mini-ndn as an experiment.

- Mini-ndn is a mininet based emulation tool that enables easy deployment and configuration of NDN nodes. [2]
- Each mini-ndn node has its own forwarder, and runs as a BMS node (for example, Room b217 in Young Hall).
- BMS nodes in mini-ndn connects to UCLA's gateway sensor data publisher, and to the testbed

Figure 6 illustrates the structure of our deployment.

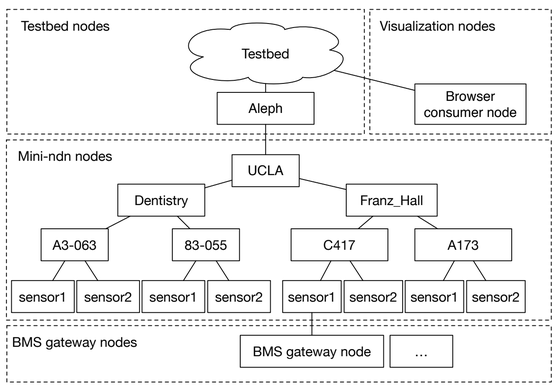

Figure 6: BMS deployment structure

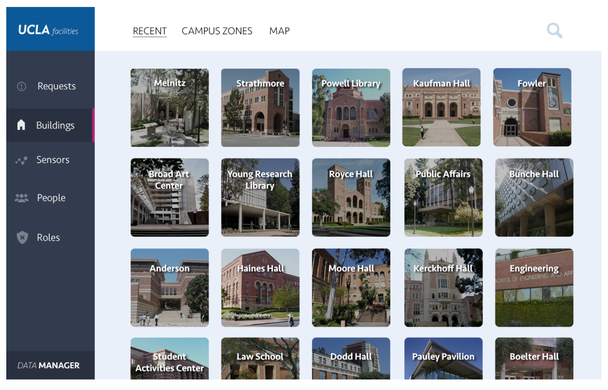

Figure 7: BMS browser UI demo

An in-browser consumer and visualization interface, as demonstrated in Figure 7 is being developed.

Future Work

- Access control in BMS
- In-browser consumer interface
- BMS command namespace

References

[1] Wentao Shang, Qiuhan Ding, A. Marianantoni, J. Burke, and Lixia Zhang. Securing building management systems using named data networking. *Network, IEEE*, 28(3):50–56, May 2014.
[2] Mini ndn on Github. `https://github.com/named-data/mini-ndn`.
[3] Yingdi Yu, Alexander Afanasyev, David Clark, kc claffy, Van Jacobson, and Lixia Zhang. Schematizing and automating trust in named data networking. Technical Report NDN-0030, Revision 3, NDN, June 2015.

zhehao@remap.ucla.edu