A Smart Home Solution over CCN

Vartika Srivastava, Dabin Kim, Young-Bae Ko
Department of Computer Engineering,
Ajou University, Suwon, South Korea.
{vartika, dabin912}@uns.ajou.ac.kr, youngko@ajou.ac.kr

I. INTRODUCTION

Content-centric network (CCN) [1] is an emerging future network paradigm that concentrates on the content itself rather than the content owner or location information. Many research efforts have been made to apply CCN over various fields such as sensor networks, vehicular networks, mobile ad-hoc networks, disaster recovery services, machine-to-machine communications and Internet of Things (IoT). Especially, IoT is an upcoming network scenario which can tap into the benefits of CCN [2]. A smart home solution is a popular IoT application which needs inter-operability among smart devices in the home in a secure fashion. Although, this sounds like it should be a part of the everyday world; even after twenty years of its inception, it is still a developing technology. This is because of the setup, maintenance and security challenges that a smart home solution faces. CCN supports ease of naming and scoping, security and as a platform for providing the devices connectivity to the outside world without the need of any translation gateway.

Based on these benefits, Ravindran, et al proposed Homenet [3], It comprises of a node discovery and a service discovery component. The node discovery scheme consists of sending out periodic interest packets for node discovery to the device. The status queries are also sent to the device. However, due to the device-centric nature of this approach, both of these will be burdensome on stand-alone devices. The service discovery module requires the services to agree on standard values for actions. This might be difficult to follow because of the variety of devices and service updates. To address these issues, we propose server-centric node discovery and service discovery methods, i.e., the devices are set up and managed by a central storage while device control takes place via direct communications. We believe that, as our scheme is based on the concept of CCN, it can provide easier setup and maintenance of the smart home by utilizing the name-based networking, better synchronization of device data, more robust security and reduced energy consumption.

II. SMART HOME OVER CCN

A. Proposed Solution

The target of our proposed solution is to create a system of inter-awareness between groups of devices over a CCN network. This scheme consists of two communication planes: 1) server plane and 2) data plane. The server plane is the connection between servers that have access to the central storage, the smart home devices and controllers. It is responsible for device registration and controller-device mapping, and providing a robust and resilient access to the central storage. The data plane presents the direct connection between devices and controllers, and is responsible for changing and updating the device status. The underlying architecture removes load off the devices because after registration, the device does not send any interest packets for node discovery on its own. Fig.1 depicts the overall framework for our smart home over CCN. As the figure shows, the inter-relationship between multiple smart devices is maintained by a central storage system accessible via a network of servers. Note that a central storage maintains detailed device information as described in Fig.2, which shows the schematic diagram at the central storage.

B. Core Functionalities

Device Registration: When a smart home device is powered on for the first time, it sends out a one-time registration interest packet to the server closest to it with device serial number, CCN-id, timestamp, timestamp encrypted with its private key, service description information and service version number. When a server receives this information, it checks whether the device is authentic or a fake by decrypting the timestamp with the device key for that device. It then creates an entry in the device_details and service_details tables with device information and supported intents of the device (See Fig.2). At this time, the field <Device_status> is set ‘To be registered’ and the field <Registered_user_list> is empty. At this stage, the smart home device is ready and waiting for controller devices.
Controller-Device Mapping: When a user (= controller device) wants to register as a controller to a smart home device, it sends out an interest packet to the closest server with device CCN-id, timestamp, timestamp encrypted with key-id for that device and its own CCN-id. The key-id of devices can be distributed at the time of purchasing the set up or through a key distribution service [4]. On receiving this interest packet, the server authenticates that the controller indeed has access privileges and if so, it adds the controller device to the <Registered_user_list>. At this stage, the field <Device_status> is marked “registered” and it will remain so until the device is reset. The server then sends out response data to the user with the registered user list and service description information. The server includes a timestamp and timestamp encrypted with key to ensure that only authorized receivers can use this information. After user receives this response, it communicates directly with all the other registered users and the smart home device that it is also a registered user henceforth. After this step, all the controller devices have an updated list of the registered users.

Requesting Control: When a controller wants to change the status of a smart home device, it will send out an interest packet directly to the device with an action, timestamp, timestamp encrypted with key-id for that device, device’s CCN-id and its own CCN-id. The action field will be extracted from the service description information of the device which is stored with the controller. The device will respond to the controller with its status.

Status Updates: When a controller device gets the status for a smart home device, it sends out an interest packet to the other controllers registered as observers with the device. This interest packet consists of the device CCN-id, its current settings and the current settings’ version number. The interest packet in this case acts as a control message instead of a named data request [5]. After receiving this interest packet, the observer synchronizes the status with the latest settings if the version is newer.

C. An Example of Operational Scenario

When a smart fridge is switched on for the first time, it registers with the central storage with its CCN-id and supported intents (e.g. increase_cooling, decrease_cooling). A smart phone becomes a controller for this fridge by sending an interest packet with the fridge’s key-id to the central storage. It can then increase cooling in the fridge by sending the intent, increase_cooling, to the fridge. CCN provides security here in the sense that if the phone is not authenticated, it will not know that the fridge supports increase_cooling and even if it gets to know, it cannot call this method without the key-id.

III. CONCLUSION AND FUTURE DIRECTION

In this paper, we have proposed a novel smart home solution over CCN, where we tried to address the common problems that the existing solutions face. The advantages of the proposed CCN-based scheme are as follows.

Easy setup: Smart home is not yet truly an off-the-shelf product because of installation issues. With our proposed scheme, the user can securely and easily set up the smart home devices by scanning the barcode or entering the CCN-id of each device when she/he installs them.

Easy synchronization: As soon as the status of a device changes, it sends out the new status to the current controller which in turn sets the new status in the remaining registered controllers and also in the central database.

Robustness: Our scheme provides robustness and resilience through a distributed control plane and a synchronization mechanism between the controller devices.

Energy Saving: When a controller changes the device’s status, the device sends it status to that controller. This controller sends out the status to the other controllers and so it will not be polled for its status again and again. This saves the power resources of the device.

In future, we plan to do performance evaluation of our solution on the CCN testbed. We also plan to extend the solution to improve user experience so that status update information can be passed between devices and a device can sometimes control other devices without the controller intervention.

IV. ACKNOWLEDGEMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2012R1A1B3003573)

REFERENCES