Demo: A Mesh-based Command and Control Sensing System for Public Safety Scenarios

Nils Aschenbruck [1 2]
na@cs.uni-bonn.de

Jan Bauer [1]
bauer@cs.uni-bonn.de

Raphael Ernst [1]
ernst@cs.uni-bonn.de

Christoph Fuchs [1 2]
cf@cs.uni-bonn.de

Jonathan Kirchhoff [1]
kirchho@cs.uni-bonn.de

[1]University of Bonn - Institute of Computer Science 4,
Friedrich-Ebert-Allee 144, 53113 Bonn, Germany

[2]Fraunhofer FKIE,
Neuenahrer Str. 20, 53343 Wachtberg, Germany

Abstract

Public safety organizations need robust communication networks to transmit different kind of sensor information. These networks must be reliable even when all infrastructure has been destroyed. Wireless multi-hop networks (such as Mobile Ad-Hoc Networks (MANETs), Wireless Sensor Networks (WSNs), and Wireless Mesh Networks (WMNs)) are supposed to meet the requirements of (1) spontaneous deployment, (2) being independent of any kind of existing infrastructure, and (3) robustness in the sense of self-organization and self-healing by their very definition. These networks have been a topic in research for more than a decade now. Recently, real-world tests and deployments provide valuable insights concerning challenges and future research directions. There are different mesh and WSN testbeds (e.g., [4, 9, 10]) enabling the research community to run tests in static real-world networks. However, concerning public safety requirements, there are significant differences: (1) No spontaneous deployment, (2) no or at least no mobility typical for public safety, (3) no typical applications and traffic for public safety scenarios. Due to these characteristics, developing algorithms and protocols for public safety scenarios and deploying public safety networks is a huge challenge.

To overcome this challenge, we developed a prototype based on commercial off-the-shelf (COTS) hardware. The prototype comprises typical public safety application and is spontaneously deployable. Furthermore, this prototype enables us to perform evaluations with real public safety end-users, e.g. by deploying the prototype in maneuvers. In our demo, we will demonstrate our COTS-based prototype.

SenSys'11, November 1–4, 2011, Seattle, WA, USA.
ACM 978-1-4503-0718-5/11/11

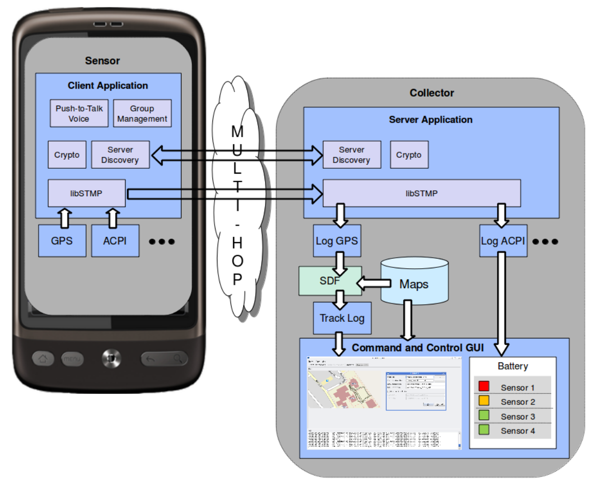

Figure 1. BonnSens System Architecture.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Design, Reliability, Management

Keywords

Sensing System, Command and Control, Public Safety

1 System Architecture

In public safety scenarios there are two core requirements for a command and control system: (1) classical push-to-talk voice communication and (2) map-based tracking. On the different hierarchical layers members of a talk-group communicate via push-to-talk voice channels. Furthermore, the people in charge want to know where their units are.

The architecture consists of a distributed sensor and collector application for the transmission of sensor data over a wireless multi-hop network as shown in figure 1. The sensor supports the collection of sensor information (e.g., GPS, accelerometers, and magnetometer) via modular extensible plugins. The transmitted sensor data is stored in a database by the collector on the server side. Depending on the type of

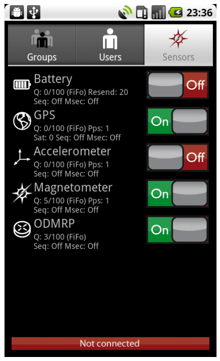

Figure 2. Screenshot of the BonnSens client application.

sensor, the data is visualized using different types of GUIs. Some kind of data, such as positing data, may be additionally processed by a sensor data fusion algorithm before being visualized on a map. The voice communication is realized using a peer-to-peer based voice application including a dynamic group management on the lightweight nodes. Due to space limitations we omit further details on the voice communication part.

The architecture consists of two components: (1) portable, lightweight sensing nodes and (2) fully-equipped collector nodes. For the sensor nodes we use standard COTS smartphones. Many smartphones on the market today have integrated sensors like GPS, accelerometer, and compass. These devices are an ideal basis for the lightweight node. We have implemented a client sensing application for Android OS. Figure 2 shows a screenshot of our sensor data client application *BonnSens*. For the lightweight nodes, we currently use the HTC Desire smartphone. For the fully-equipped collector nodes, standard COTS laptops can be used. All nodes (lightweight and fully-equipped) are routers at the same time. However, to have a more robust network and to safe energy at the lightweight nodes, we add a mesh backbone. For the mesh, we tested two kinds of COTS mesh routers: (1) ASUS WL-500g Premium V1 and (2) ALIX 3D2 In both routers we use WiFi-cards with Atheros chipsets (TP-LINK TL-WN660G). For an easy on site deployment an infrastructure-independent power support is needed as well. Thus, we use motorbike-batteries with 12V-20Ah. Using these batteries, we can run the mesh backbone for more than 12 hours without any infrastructure.

For routing we implemented a multicast approach, as both core requirements imply multicast. The voice application is group-based. Thus, it can be efficiently realized by multicast-groups. In some public safety scenarios, there may be several fully-equipped nodes with a demand for a visualization. As multicast routing protocol we chose the reactive On-Demand Multicast Routing Protocol (ODMRP) [8]. ODMRP is mesh-based scheme that is based on scoped flooding. A selected subset of nodes forwards the packets. We chose ODMRP as it showed promising results in public safety specific simulative performance evaluations [2]. Furthermore, it is quite effective even under attacks like sinkholes, as the mesh structure provides robustness against the attraction of routes. We implemented ODMRP using the Click modular routing framework [7]. The Click user-space mode enables us to run ODMRP on the mesh router as well as on Android phones.

To provide functionalities for sensors such as registering at a receiver, timestamping of sensor data, synchronization of all nodes, as well as providing sensor management functions, we specified and implemented the Sensor Data Transmission and Management Protocol (STMP) [1]. In order to realize a consistent implementation of STMP for lightweight as well as for fully-equipped nodes, we modularized STMP by implementing it as a commonly usable library in C named *libSTMP*. Client applications may thus be programmed either using the Android API for the deployment on smartphones or using native C code for the deployment on a laptop both accessing the same STMP library.

To provide accurate tracking appropriate sensor data fusion algorithms have been integrated. As shown in [5] a standard Kalman filter [3] may suffer significantly from Out-of-Sequence (OoS) measurements. An Accumulated State Density (ASD) methodology [6] which allows to calculate the impact on all states within a given time window proved to be a valuable alternative.

2 Conclusion and Future Work

We have developed *BonnSens* a COTS-based prototype of a mesh-based command and control sensing system for public safety scenarios. Overall, our goal is to see which approaches are applicable for public safety networks and where further specific challenges are. Next, we will perform several on-site deployment tests in disaster area maneuvers.

3 References

[1] N. Aschenbruck and C. Fuchs. STMP - Sensor Data Transmission and Management Protocol. In *Proc. of the 36th IEEE Conference on Local Computer Networks*, 2011.

[2] N. Aschenbruck, E. Gerhards-Padilla, and P. Martini. Modeling Mobility in Disaster Area Scenarios. *Elsevier Performance Evaluation - Special Issue on Performance Evaluation of Wireless Ad Hoc, Sensor and Ubiquitous Networks*, 66(12):773–790, 2009.

[3] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. Wiley & Sons, 2001.

[4] J. Bicket, D. Aguayo, S. Biswas, and R. Morris. Architecture and Evaluation of an Unplanned 802.11b Mesh Network. In *Proc. of the 11th ACM Intern. Conference on Mobile Computing and Networking*, 2005.

[5] F. Govaers, C. Fuchs, and N. Aschenbruck. Evaluation of Network Effects on the Kalman Filter and Accumulated State Density Filter. In *Proc. of the 2nd Int. Workshop on Cognitive Information Processing*, 2010.

[6] W. Koch. On Accumulated State Densities with Applications to Out-of-Sequence Measurement Processing. In *Proc. of the 12th ISIF International Conference on Information Fusion*, 2009.

[7] E. Kohler, R. Morris, B. Chen, J. Jannotti, and F. M. Kaashoek. The Click Modular Router. *ACM Transactions on Computer Systems*, 18(3):263–297, Aug. 2000.

[8] S.-J. Lee, M. Gerla, and C.-C. Chiang. On-demand multicast routing protocol. In *Proc. of the IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1298–1302, 1999.

[9] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh. Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols. In *Proc. of the IEEE Wireless Communications and Networking Conference*, 2005.

[10] G. Werner-Allen, P. Swieskowski, and M. Welsh. Motelab: a wireless sensor network testbed. In *Proc. of the 4th International Symposium on Information Processing in Sensor Networks*, pages 483–488, 2005.

This work was supported in part by CONET, the Cooperating Objects Network of Excellence, funded by the European Commission under FP7 with contract number FP7-2007-2-224053.