IEEE 802.21 MEDIA INDEPENDENT HANOVER
DCN: 21-10-0144-02-HWNM

Title: Practical experiences with media-independent link control

Date Submitted: July 13, 2010

Presented to Heterogeneous Wireless Networks Mgmt. SG at IEEE 802.21 session #39 in San Diego

Authors or Source(s):
Krzysztof Grochla, Walter Buga (Proximetry)

Abstract: A description of practical experiences in creating a common interface for multi-technology radio interface management software

IEEE 802.21 presentation release statements

This document has been prepared to assist the IEEE 802.21 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.

The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE’s name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE’s sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.21.

Content

• Introduction
• EU-Mesh works in correspondence to requirements identified within 802.21
• How we try to realize the requirements identified for this SG
• Interfaces for cross layer interface management used in EU-Mesh
  • XIAN
  • DDL
• Lessons learned
• Some comments to the API primitives identified

Background

• EU-Mesh project
  • a 30 month collaborative project which started January 2008, and is funded by the European Commission under Call 1 of ICT (Information and Communication Technologies) in FP7 (7th Framework Programme), targeting the objective “The Network of the Future” of Challenge 1: Pervasive and trusted network and service infrastructures.
  • EU-MESH’s goal is to develop, evaluate, and trial a system of software modules for building dependable multi-radio multi-channel mesh networks with QoS support that provide ubiquitous and ultra-high speed broadband access.
  • Multiple radio technologies can be used to form a mesh network
  • Links using WIFI and WiMAX
  • Layer 3 routing highly preferred
EU-MESH Data Plane Architecture

Network Management in EU-Mesh

- Proximetry
  - NMS system for WiFi / WiMAX network (including mesh)
  - 802.16d BS
Req 1: Topology Formation/Radio Config

Mesh node with WLAN interfaces
Mesh node with WiMAX interfaces

NMS server

Topology Formation/Radio Config cont.

• Common interface for network management of 802.11 and 802.16 interfaces
• DHCP used for autoconfiguration, CAPWAP used for management of the nodes
• NMS server providing autoconfiguration for both WiMAX and WIFI devices
• Topology optimization / channel assignment algorithms for multi-radio WiFi mesh networks, centralized optimization algorithms executed on server
• Support for topology optimization in heterogeneous networks still seen as a future work
Req. #2: LSP Setup/Resource Reservation

- Mesh resource management functions provided by the NMS server and GUI
- Common interface for QoS parameters for WiFi and WiMAX
  - Defining the parameters by bandwidth and latency limits
  - Mapped to WiMAX classes of service
  - Mapped to IP HFSC queues on WiFi
- No automatic path reservation for multihop networks
- No interaction with routing

Req. #3: Link State Info Propagation

- Link state information propagated to the NMS server using CAPWAP messages
- The NMS server monitors the network topology, provides its visualization for the user
- Single point of gathering the knowledge provides easier implementation of topology management algorithm
How we do so far

- The agent on each of the devices gathers the information and provides it to the NMS server
- The code is specific for each type of interface and specific for some vendors
  - The code needs to be ported between different hardware providers, even for the same technology

<table>
<thead>
<tr>
<th>Vendor specific</th>
<th>Technology Specific</th>
<th>OS Specific</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WIFI</td>
<td>Linux</td>
</tr>
<tr>
<td></td>
<td>WiMAX</td>
<td>VxWorks</td>
</tr>
</tbody>
</table>

How would we utilize the media independed interface

- Single code for multiple vendors and interfaces
Interfaces used for link management

- DDL
  - Proximetry Propietarry
  - Supports WiFi and WiMAX
- XIAN
  - Developed by Thales
  - WiFi only

XIAN

- Library for accessing WiFi metrics from 802.11 MAC layer
- Communication is implemented on a Request/Response model
- Microprotocol to provide metrics from neighbor nodes
Function provided by XIAN

- Configuration states
  - configuration parameters of the 802.11 network device
- Aggregated metrics
  - counters providing statutes of the 802.11 interface
- Per neighbor/link metrics
  - Statistics of specific connections

DDL

- Proprietary library for monitoring and managing the interfaces and devices
- Support for WIFI and WiMAX
- Function provided:
  - Configuration management
  - Statistics collection and link state monitoring
  - QoS management
- Reading the parameters by multiple interfaces
  - /proc filesystem
  - ioctl
  - interprocess communication
Parameters generalization

• To easily manage multiple parameters on different devices we introduced a concept of capability files
  • parameters identified by ID
  • capability files defines what device/interface supports what parameters and provides the user-readable description
  • the agent has hardcoded mapping between parameter ID and implementation of its application on specific platform

• This may be implemented by providing a space for vendor-specific parameters IDs
  • Application by MIH_Radio_Set_Parameters
• MIH_Radio_Get_Parameters / MIH_Radio_Set_Parameters
  • The list of parameters should include multiple parameters, starting from IP address ending on some technology specific parameters like e.g. beacon interval for WiFi
  • Easy way of adding a vendor-specific parameter would be a plus

• The flow may be identified not only by layer 3 5-tuple, but also by layer 2 fields, e.g. by VLAN ID