TUNEGreen: A Distributed Energy Consumption Monitor for Wireless Networks

Karina Gomez, Roberto Riggio,
Daniele Miorandi, and Imrich Chlamtac
CREATE-NET
Via Alla Cascata 56/C, 38123, Trento, Italy
Email: name.surname@create-net.org

Fabrizio Granelli
DISI - University of Trento
Via Sommarive 14, 38123, Trento, Italy
Email: fabrizio.granelli@unitn.it

I. Motivation

Limited resources and the increasing electricity costs are driving the research agendas of governments, industries and academics towards the promotion of more sustainable, 'greener' technologies and systems[1,2]. The Information and Communication Technology (ICT) sector is no exception. In a recent study, it has been estimated that ICT is responsible for about 7% of the total electricity consumption in the EU [1]. Further, the costs related to electricity bill are becoming a major factor limiting the competitiveness, margins and growth of telco operators, in particular of those offering wireless connectivity to end users.

Studies [2], [3] have also shown that the energy consumption of wireless network infrastructure represents about 50% of the total energy consumption of the ICT sector. Yet, a number of key questions remain —to a large extent— unsolved:

1) Where is power consumed in wireless devices?
2) How much power is consumed in transmission, reception and idle modes?
3) Which is the relation between power consumption and traffic load in the network?

Answering such questions would help in identifying opportunities to optimize existing wireless networks protocols in terms of energy consumption. At the same time, profiling power consumption in network devices would be key to define priorities in the related research areas.

In order to address the research challenges outlined above, we designed and implemented *TUNEGreen*, a distributed energy consumption monitoring system capable of gathering and processing power consumption data of a variety of wireless networking devices (e.g. access points, wireless routers, etc.), as well as of correlating such information with data on the status (in terms of CPU, memory load, and network traffic generated/received/relayed). An interactive Web-based interface completes the system, providing means for visualising in real-time the energy consumption patterns.

[1]European Union Funded Projects' Information. Material available at: http://cordis.europa.eu/fp7/ict/sustainable-growth/

[2]Green Touch Industry–driver Initiative. Available at: http://www.greentouch.org

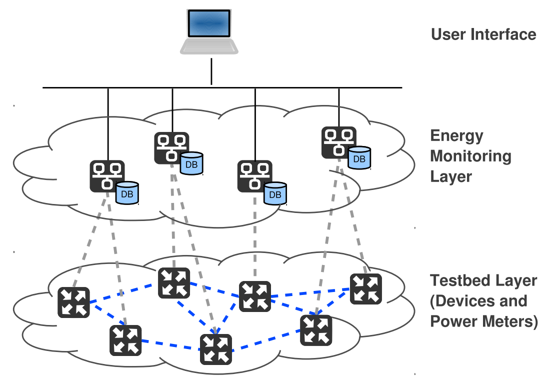

Figure 1: The *TUNEGreen* system architecture.

In this demonstration we will show how the *TUNEGreen* real-time energy consumption monitoring capabilities can be used to understand how and where power is consumed in a simple star-shaped network based on IEEE 802.11. Different traffic patterns will be generated in order to demonstrate the correlation between traffic and energy consumption. The audience will be provided means to perform simple Web browsing tasks (e.g. download a file or watch a video using YouTube) while monitoring the actual power consumption of the wireless router being exploited to access the public Internet.

II. Technical Details

A. Network Architecture

An overview of the *TUNEGreen* architecture is reported in Fig 1. The system consists of three logical layers:

- The testbed layer, containing the network devices whose energy consumption is being monitored and represented in the figure as a multi-hop wireless network;
- The energy monitoring layer, containing the nodes responsible for polling the network devices and for storing the gathered information;
- The system manager layer where, possibly multiple, instances of the *TUNEGreen* web-based interface are running.

The energy monitoring task is implemented in the form of a system daemon written using the Python language. It periodically polls the status of one or more network

978-1-4577-0351-5/11/$26.00 ©2011 IEEE

interfaces/devices. Such information, exported through a RESTful interface, is made available to the system manager trough a web-based interface. The information gathered by the daemon are not limited to the energy consumption statistics, but also include: CPU and memory utilization, transmission and reception statistics for every network interface supported by the network device, and a few more general information such as the system's uptime and the version of the OS.

It is worth noticing that, albeit represented as separate entities in the figure, the energy monitoring nodes can coincide with the actual network device being monitored. In such a case, the energy monitoring daemon is embedded within the monitored device. Alternatively, a dedicated energy monitoring node can be used when such functionality cannot be embedded into the network device itself, e.g. due to licensing issues, lack of computing power, and/or lack of a proper networking stack/APIs. In both scenarios, the actual energy measurement is performed with the help of a dedicated power meter connected to the network device and exporting the energy consumption statistics thorough a suitable interface (a serial interface in the current implementation of the prototype).

B. Energy Measurement

In the current implementation the actual energy consumption monitoring is performed using the *Watts up?* devices[3]. The *Watts up?* devices are "plug load" meters that measure the amount of electricity used by whatever device is plugged into them. The measurements are taken every second with a granularity of 0.1 Watts. The meter is equipped with a mini USB port and an Ethernet connector. The former can be used to access the data stored in the meter from a personal computer, while the later is provided to seamlessly integrate with the *Google PowerMeter* infrastructure.

The choice of the *Watts up?* devices as energy monitors was motivated by their advanced logging and data processing facilities. However, their use in a large scale testbed (or, even worse, in a real deployment) is not realistic due to their relatively high cost (≈ 200 €). For such scenarios, the authors envision the utilization of simpler devices without data processing functionalities and specifically tailored for DC loads. We expect the per-node cost of such solution to be one order of magnitude cheaper than the *Watts up?* devices for a significant number of nodes.

III. Demo Description

In this demonstration we will show a live energy measurement of a simple wireless network with the purpose of illustrating the relationship between network traffic and power consumption.

[3]https://www.wattsupmeters.com/

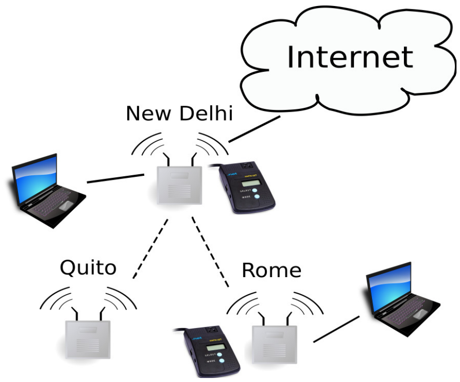

Figure 2: The live energy monitoring scenario considered for this demonstration.

A. Scenario

The monitored network consists of three wireless mesh routers implementing a flat wireless mesh network topology, as depicted in Fig. 2. One wireless router, *New Delhi*, is acting as mesh gateway providing the network with Internet connectivity through a wired Ethernet connection, while the two remaining routers, *Rome* and *Quito*, are exploited by two client PCs. The mesh backhaul has been implemented using the WING toolkit[4] an experimental IEEE 802.11-based wireless mesh network [4], [5]. The *New Delhi* and the *Rome* mesh routers are connected to the power grid trough the *Watts up?* device and are running the energy consumption monitor. The Web-based interface, visualized on the two PCs for the entire duration of the demonstration, will be configured in such a way to show the power consumption of the two routers and the actual traffic flowing through their wireless interfaces.

During this demonstration we will measure the power consumed by the *New Delhi* wireless router when different traffic patterns are transmitted to the *Quito* wireless router. In parallel we will show the energy consumption of a passive node (*Rome*), which is located within the collision domain of the two communicating routers. In order to do so, we will perform a set of experiments varying the rate at which the packets are injected into the network and the size of each packet. The traffic will be generated using Multi-Generator (MGEN)[5] tool. Two different traffic scenarios will be considered:

1) Fixed packet size equal to 1500 bytes and packet generation rate varying between 500 to 4000 packets/s in steps of 500 packets/s (the packet generation rate gets changed every 20s).
2) Constant throughput of 12 Mbps using variable packet length, ranging from 50 up to 2500 bytes (the packet length gets changed every 20s).

The results of a measurement campaign carried in the former scenario and relative to the *New Delhi* wireless router

[4]Online resources available at *http://www.wing-project.org/*

[5]Available at: http://cs.itd.nrl.navy.mil/work/mgen/index.php

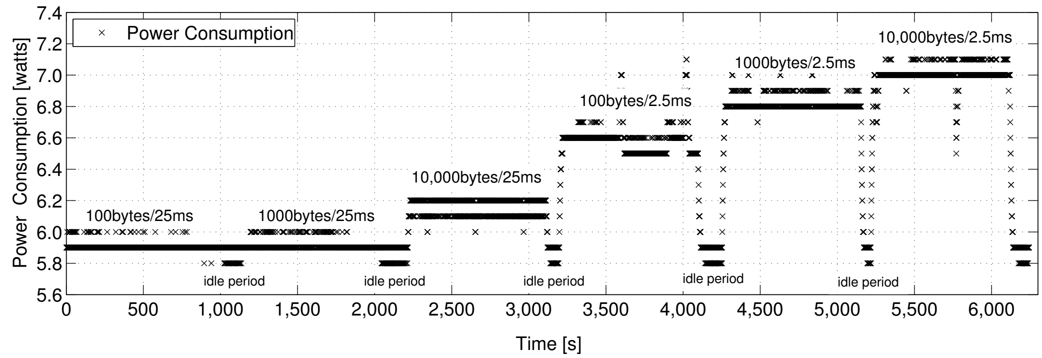

Figure 3: Power consumption level of one wireless device as a function of time for different traffic generation rates.

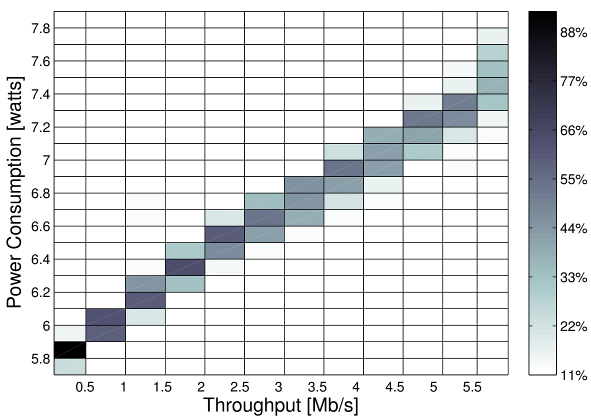

Figure 4: Histogram of the power consumption Vs. the amount of traffic generated at the *New Delhi* wireless router.

are reported in Fig. 3 as a reference. In the figure, we can clearly identify the increase in the power consumption of the wireless router as a consequence of the increase in traffic generated, together with the short idle periods in the transition between different traffic generation rates.

The histogram of the power consumption Vs. the amount of traffic generated at the *New Delhi* wireless router is reported in Fig. 4. As it can be seen, for low values of the throughput more than 77% of the power consumption samples are concentrated around the same value, while for higher values of the throughput the power consumption samples are distributed across a wider range of values.

B. Audience Interaction

The audience will be able to interact with our demonstration by using the PC connected to the *Rome* node in order to perform simple Web-based tasks such as uploading pictures to Flickr, downloading a large file using FTP, or watching a high-resolution video clip using YouTube. The audience will then be able to monitor in real-time the actual power consumed by the *New Delhi* wireless router being exploited to access the public Internet, and therefore understand how the different tasks performed may impact the energy consumption of the underlying network infrastructure.

IV. Conclusions

In this work, we have presented the *TUNEGreen* system for monitoring and visualising in real-time the power consumption of wireless network's devices. The demo will provide the conference attendees with the opportunity of experiencing how different Web-based tasks (e.g., downloading a file, watching a movie etc.) can influence the power consumption of the underpinning network infrastructure. Results of the an extensive measurements campaign performed by the authors will also be shown, providing attendees with insight into how various parameters affect the power consumption of wireless nodes.

Acknowledgment

This work was partially supported by the EC Seventh Framework Programme (FP7/2007-2013) within the framework of the SMART-Net project (grant number 223937).

References

[1] EU, "Impacts of Information and Communication Technologies on Energy Efficiency," Bio Inteligence Service, Tech. Rep., Mar. 2008, available at: http://opera-net.org/Documents/ict4ee-final-report_en.pdf.

[2] H.-O. Scheck, "ICT & Wireless Networks and their Impact on Global Warming," in *European Wireless Conference*, Lucca, Italy, 2010, pp. 911–915.

[3] O. Blume, D. Zeller, and U. Barth, "Approaches to energy efficient wireless access networks," in *Proc. of the 4th International Symposium on Communications*, ser. ISCCSP 2010. Limasso1, Cyprus: Control and Signal Processing, March 2010, pp. 1–5.

[4] R. Riggio, N. Scalabrino, D. Miorandi, F. Granelli, Y. Fang, E. Gregori, and I. Chlamtac, "Hardware and software solutions for wireless mesh network testbeds," *IEEE Communication Magazine*, vol. 46, no. 6, pp. 156 – 162, Jun. 2008.

[5] R. Riggio, K. Gomez, T. Rasheed, M. Gerola, and D. Miorandi, "Mesh your senses: Multimedia applications over wifi-based wireless mesh networks," in *Proc. of Secon*, Rome, Italy, 2009.