E-Net-Manager: A Power Management System for Networked PCs based on Soft Sensors

Simone Brienza, Fabio Bindi, and Giuseppe Anastasi
Dept. of Information Engineering
University of Pisa, Italy
simone.brienza@for.unipi.it, f.bindi@gmail.com, giuseppe.anastasi@unipi.it

Abstract — **The overall energy consumption due to ICT equipment has followed an increasing trend over the last years. A considerable fraction of the consumed energy is caused by user devices, such as Personal Computers (PCs) and displays. However, a large part of this energy is wasted due to an inefficient use. Users leave their PCs on for long periods even when unused, especially in workplaces. Hence, significant energy savings could be achieved just turning them off. However, it is not wise to rely on user collaboration, and, thus, automated tools are needed. In this paper, we present *E-Net-Manager*, a power management system for large environments, which turns unused PCs off and switches them on when the user is about to use them. To this end, E-Net-Manager leverages soft sensors, i.e., software/hardware tools already in use by the users, thus not introducing any additional cost. E-Net-Manager combines information provided by the users and data obtained from a number of these soft sensors. This way, it is possible to accurately determine the user presence/activity near her/his PC and, therefore, eliminate wastes also due to short periods of inactivity.**

Keywords — Power Management, Large Computing Environments, Wake-on-LAN, Soft Sensors, Low intrusiveness

I. Introduction

Information and Communication Technology (ICT) devices and services cause a significant energy consumption. Recent studies [1] highlight that almost 5% of the overall worldwide electricity consumption is due to ICT equipment. Although this percentage is not so high, its absolute value is very remarkable and has followed and increasing trend over the years [2]. A large fraction is due to user devices, such as Personal Computers (PCs), printers, and displays [3]. Indeed, even though the power consumed by a single device is limited, the total consumed energy is relevant, because of their large number and utilization time. In particular, the worldwide energy consumption of PCs and displays is currently estimated in about 300 TWh per year [1]. A significant fraction of this energy is wasted due to an inefficient utilization of infrastructure and user equipment. This is true especially in offices, where users generally do not pay so much attention to the energy problem. For instance, the authors of [4] show that PCs in an office environment are left unattended and switched on 28% of the time. [5] has found that 18% of PCs are never turned off at night and weekends.

Reasons for leaving the PC on are numerous [6]. The main ones are simply carelessness and laziness. Also, the user can leave her/his PC on, in order to be able to access it remotely (e.g., from home) or to avoid the hassle of long start times the following morning. Even when PCs have automated power management features enabled by default, many consumers disable them at some point and often forget to re-enable them [7]. For instance, operating system's power management policies (e.g., for automatic suspension or hibernation) are often disabled so as to avoid PCs to be switched off when the user is still in her/his workplace, or to not interrupt sensitive processes and applications.

To investigate the utilization pattern of PCs and other networked devices in working places, we monitored our university campus, for a period of two months. To this end, we used an approach similar to that described in [8]. Figure 1-left shows the number of active devices at different times of the day, for a working day and a holiday, respectively. These results confirm that a very large number of PCs are never turned off by their users. In addition, we have found that almost all PCs are always active during the working time, even when their users are far from the workspace. Figure 1-right shows three typical PC utilization patterns. The first user (PC1) never turns her/his PC off (e.g., to be able to access it from a remote location). The other two users switch off their PC regularly when they leave their office. In addition, the user of PC2 switches it off also during lunch time. However, in both cases the PC remains active almost continuously during the day, even when the user is busy with meetings and classes.

Considering that a typical desktop PC with a LCD display consumes roughly 115W when idle [9], it clearly emerges that significant energy savings could be achieved simply switching off unused PCs. As an alternative, modern desktops support the ACPI S3 (Sleep/Standby) energy state [10], which can reduce the power consumption by 95% or more, still providing quick resuming times [3]. Nevertheless, it is not reasonable to rely only on the user cooperation, especially in large organizations. Hence, automated power management systems are required to switch off PCs when they are not used, with minimum user involvement.

Ideally, the PC should be switched off whenever the user is not using it and resumed when the user is willing to re-use it. In practice, this behavior is impossible to implement. To approximate it, context-aware power management solutions have been proposed that rely on user's presence/activity detectors. Some of these solutions are based on sophisticated, and expensive, location systems [11]. Other solutions exploit low-power low-cost sensors that provide only approximate information about the user's locations [12][13]. Obviously, they are less expensive, however they may result in undesired

Work carried out in the framework of the SmartBuildings project funded by the Regional Government of Sicily (POR FESR Sicilia 2007-2013).

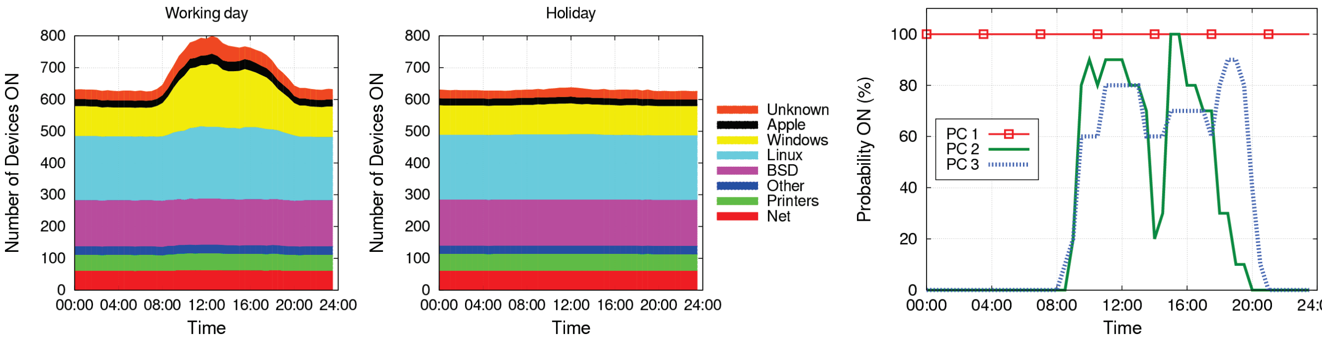

Fig. 1. Number of active devices during the day (left) and PC utilization patterns (right)

shutdowns/activations of the PC that are quite annoying for the user.

In this paper we propose *E-Net-Manager*, an automated system that leverages context-aware power management. It is based on a number of different *soft sensors*, i.e., devices and software tools already available in the environment – but used for other purposes – capable of providing information about the user's presence and/or activity. Such soft sensors include *attendance recorders*, *Bluetooth-enabled mobile phones*, and software utilities such as *Google Calendar* and *PC activity monitors*. By using different soft sensors, *E-Net-Manager* can be suitable for different kinds of user. In addition, by combining information from different soft sensors, it is able to minimize undesired shutdowns/activations of the PC. On the other hand, using more soft sensors does not introduce additional costs as they are already available in the environment. Our experimental results, obtained in a real environment, show that E-Net-Manager can significantly reduce the energy wastes, especially in offices.

The rest of the paper is organized as follows. Section II presents the related work whereas section III describes the features provided by E-Net-Manager. Section IV discusses the system architecture, while Section V provides some implementation details. Section VI shows the results obtained testing E-Net-Manager on a real test bed. Finally, section VII concludes the paper.

II. Related Work

The problem of reducing the energy waste for PCs and other networked devices has been widely dealt with in the literature. A detailed survey of the main solutions proposed in the literature can be found in [14].

For desktop computers, significant benefits – in terms of energy consumption – can be achieved by powering off the devices during idle times. Therefore, many systems have been proposed to automatically control the state of networked devices in order to turn them off when unused and switch them on when required. Some of the proposed solutions make use of proxy servers [15][16][17][18][19][20][21]. Basically, a dedicated proxy server acts on behalf of unused PCs, that can be turned off maintaining their network presence. If the proxy is able to manage the received request by itself, it immediately serves it. Otherwise, the proxy sends a wakeup message to the device and, then, forwards the received request. The main drawbacks of this approach concern the network architecture, since new devices must be added (dedicated PCs or ad-hoc embedded systems [22]). Furthermore, only a restricted set of applications and protocols can be handled by the proxy, thus limiting the efficacy of these solutions. For these reasons, some proxy-based solutions make use of virtualization. For instance, *SleepServer* [23] and *LiteGreen* [24] allow to migrate the image of the whole operating system of a PC to the proxy, whenever the user leaves her/his PC. This way, the PC can be turned off while being still accessible and completely operative in the network. The energy savings are remarkable, but these systems require significant setup times and procedures since they need a high-speed network infrastructure and, above all, a *Virtual Machine Monitor (VMM)* must be installed on each PC. In addition, virtual machine migration to and from the proxy server produces considerable waiting times. Therefore, a regular usage of these systems could become annoying for the users.

For these reasons, more unobtrusive power management systems have been developed, still providing significant energy savings. Among them, *PoliSave* [8] allows to schedule specific actions for automated power management of networked PCs. Basically, for each of her/his PCs, the user can specify the exact time when it should be switched off and on (e.g., when the user typically enters and exits her/his office). The controlled PC periodically queries a power manager to check for scheduled actions. If a shutdown or sleep operation is planned at that time, the PC immediately turns off. On the other side, if a wakeup is scheduled for a PC, the power manager exploits the *Wake-on-LAN (WoL)* technology, i.e., it sends a *Magic Packet* [25] to the PC. The PoliSave behavior is based on the definition of static rules, thus the user must specify in advance the exact time when the PC must be switched on/off. This results in reducing the energy wastes occurred during nights or weekends. However, PoliSave does not affect the energy wastes due to idle periods during the working time, e.g., when the user is attending a meeting. In this perspective, *Gicomp* [26] allows to automatically install/modify power management policies on PCs in large computing environments. Policies are decided by the organization management and installed by the network administrator through the system (i.e., without any user intervention). Specifically, Gicomp leverages power management mechanisms provided by all the modern operating systems. For instance, it allows to define the time to dim and turn off the display, the disk spin down timeout, the suspend timeout, the hibernate timeout. Once in sleep mode, PCs can be remotely woken up through WoL. Obviously, Gicomp is effective only if the power management strategies are decided centrally and are not disabled by users for the previously explained issues.

In order to achieve a more significant reduction of the energy consumption of PCs without bothering users, some solutions in the literature propose a context-aware approach. Essentially, these solutions exploit one or more user presence detection techniques so as to determine the position of the user with respect to her/his PC. Hence, if the user is moving from her/his workspace, the system can turn off her/his PC. Similarly, when the user comes back, it can be switched on. Some of these solutions rely on very accurate information about the user's location, obtained through sophisticated location systems or sensors. For instance, the solution presented in [11] exploits an ultrasonic system that provides the user's location with high accuracy. Obviously, although these solutions exhibit excellent performance, their complexity and costs are typically very high.

Other solutions exploit low-power, low-cost sensors that provide only approximate information about the user's position. For example, they check the radio connectivity of personal mobile devices (e.g. smartphones), to infer the presence/absence of the user in the working area [12][13]. Specifically, the solution proposed in [13] uses a policy called *Sleep/Wake-up On Bluetooth*. These systems are easy to implement, but are much less accurate than the ones based on expensive sensors. Hence, they may cause undesired shutdowns or activations of the PC.

E-Net-Manager also implements a context-aware approach. However, unlike previous solutions it relies on a number of different soft sensors, i.e., devices and software utilities already available in the environment (but used for other purposes) capable of providing information about the user's presence and/or activity. This makes it suitable for different kinds of user. In addition, by combining information from different soft sensors, E-Net-Manager is able to avoid (or minimize) undesired shutdowns/activations of the PC.

III. System Features

The goal of *E-Net-Manager* consists in reducing the energy consumption of networked PCs. Basically, it identifies the periods during which PCs are not used and turns them off. To this end, E-Net-Manager needs to know when the user is actually using her/his PC and when he/she is far from the workspace. To derive this information, the system exploits data obtained both from users and soft sensors. E-Net-Manager also allows users to remotely check and control the state of their PCs. Specifically, E-Net-Manager provides the following basic features

- *Static Control.* Each user has an account on the system and can, thus, define rules for all her/his associated PCs. The user can define rules to specify when the PC must be switched on, suspended or turned off. The system automatically executes the corresponding actions as soon as the rules become active, without any user intervention. Therefore, by properly configuring the system, it is possible to keep the PC on only during working hours, while avoiding, at the same time, to annoy the user with undesired actions. For instance, in case of shutdown or suspension, the system preliminary checks whether the user is still using the PC through the activity soft sensor (see below). If so, the operation is ignored or deferred. It is possible to define two kinds of static rules:
    - *Immediate rules.* The user specifies the specific time of the day to perform an action (e.g., "Switch off PC1 at 6 pm in Working days").
    - *Extended rules*. The user specifies a time interval in which her/his PC must be in a certain state (e.g., "Switch off PC1 from 6 pm to 8 am in Working Days"). The system constantly controls that the state of the PC is consistent with the rule for all its duration.
- *User Presence/Absence Detection*. E-Net-manager uses several soft sensors to detect the user's presence and activity within her/his working area. Then, this information is exploited to suspend/power off the PC also during those idle periods that occur sporadically (e.g., due to a meeting or a break) and cannot be predicted in advance and reflected into the system through specific control rules. Specifically, whenever the system detects that the user is far from her/his workspace, it sends a suspension command to her/his PC.
- *Remote Control*. E-Net-Manager also allows users to remotely control their own PCs. To this end, each user can access her/his personal account on the system and check the state of each registered PC (i.e., on, off, or sleeping). In addition, the user can perform specific actions, depending on the state of the PC. For instance, it is possible to turn off or suspend an active PC, or switch on a sleeping PC.

In addition, since a user may have more than one PC, E-Net-Manager allows users to create and organize *groups of PCs*. This way, the user can define static rules or perform remote-control requests that affect all the PCs in the group. In these cases, the system sends the proper command to all the PCs of the group at the same time.

E-Net-Manager also provides users with *statistics* concerning the utilization of their PCs. For each PC, the user can view how much time and when it was switched off, turned on or suspended in a specific day or on a monthly basis.

In conclusion, E-Net-Manager also exhibits the following features, resulting in a functional, secure, and unobtrusive power management system.

- *Security and privacy.* Each communication within the system is secured through encryption and authentication so as to prevent malicious users from accessing critical information (e.g., presence of a user or state of a PC).
- *Low intrusiveness*. The system operates transparently to the user (i.e., without any interaction) while trying, at the same time, to avoid any possible discomfort to her/him.
- *Interoperability*. The system can manage any PC, irrespective of its operating system.

IV. System Architecture

In this section we describe the architecture of *E-Net-Manager*. The system is supposed to work in the context of a large network composed of many hosts and organized into several subnets. In addition, the network can span multiple buildings, such as a university or corporate campus. Finally, the power management system is expected to work also in networks using dynamic assignment of IP addresses (DHCP), private IP addresses (NAT), or in the presence of firewalls. The system architecture follows the client-server paradigm and includes several components, as shown in Figure 2.

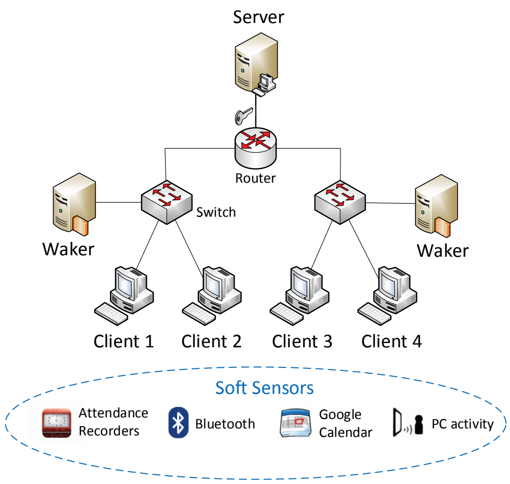

Fig. 2. Architecture of E-Net-Manager

All the various system components (i.e., controlled PCs, server, and wakers) communicate via the *eXtensible Messaging and Presence Protocol* (XMPP) [27]. It is a message-oriented communication protocol based on XML. Specifically, each PC is an XMPP independent node with its own identifier (called *JID*). Similarly, the server has its own JID. Hence, all information exchanged within the system forms XML streams. In XMPP, connections are started by clients and are persistent. This allows a bidirectional communication also in networks with NAT servers or firewalls. The connection between a PC and the Server can be created with both SSL and TLS encryption, thus achieving communication confidentiality and server authentication. Clients are authenticated using SASL (*Simple Authentication and Security Layer*) through JID-password pairs.

A. Server

The *Server* side provides the core functionalities of E-Net-Manager. In detail, the offered features are supplied by four distinct components (Figure 3).

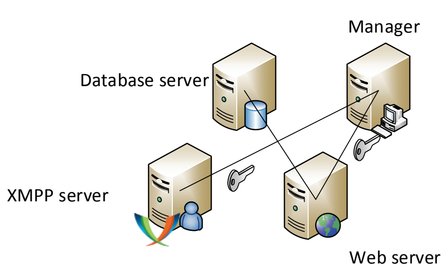

Fig. 3. Server components

Database server. Stores data about users, PCs, Wakers (see below), soft sensors, and PC groups. It also stores the static rules defined by the users and the usage statistics for each PC.

Web server. Provides the Web interface of E-Net-Manager, allowing users to (**i**) check the state of their PCs, (**ii**) control them remotely, (**iii**) define rules for "power on", "suspension" and "power off" actions, (**iv**) create/organize groups of PCs, (**v**) view soft sensors current settings, and (**vi**) view usage statistics.

XMPP server. Allows the communication between the server and the PCs to control. Specifically, it manages all the XMPP accounts and handles the message delivery among the various components of the system.

Power Manager. Manages and executes static rules introduced by the users and remote control requests. Basically, it periodically (e.g., once in a day) obtains the static rules from the database and activates specific timers for triggering commands at scheduled times. When a timer expires the Power Manager sends a power-off (suspension) command or a wakeup command to the involved PC. In both cases, the XMPP protocol is used. In case of extended rules, the Power Manager also sends the expiration time of the rule, together with the command to be executed. In addition to the static rules, the Power Manager also handles the remote-control requests that users perform via the Web interface. In practice, it waits for remote commands from the web server, listening on a specific socket. To conclude, the Power Manager is also used to create usage statistics for all the associated PCs. To this end, it exploits some XMPP capabilities. In fact, all the controlled PCs have an XMPP account and are part of the roster (a sort of contact list) of the Power Manager. Therefore, the Power Manager is always aware of the state of the controlled PCs. Basically, it knows when they are connected (i.e., turned on) and when they are disconnected (i.e., switched off). Thus, the Power Manager stores this information into the database and uses it to derive usage statistics, when requested by the users through the Web interface. Besides, this information is also used to apply extended rules. In fact, whether a PC changes its state while an extended rule is active (e.g., a PC is turned on when it should be off or vice versa) the Power Manager promptly sends the proper command to put the PC in the right state.

B. Waker

In order to remotely turn on suspended or switched off PCs, E-Net-Manager relies on Wake-on-LAN [25], a mechanism specifically designed for Ethernet network interface cards (NICs). Basically, part of the NIC circuitry remains always active – only consuming a small amount of energy – and able to receive packets. However, if the PC is switched off or in sleep mode, the NIC ignores all packets but Magic Packets. Specifically, a Magic Packet is a particular UDP datagram directed to the port 9 of the subnet broadcast address of the target PC, and its payload consists of 16 repetitions of the MAC address of the target PC. Therefore, when a NIC receives a Magic Packet containing its MAC address, the PC turns on.

The WoL mechanism is very common and supported. However, to avoid security issues, Magic Packets are usually not forwarded by routers, unless they are properly configured. Therefore, the sender and the receiver of a Magic Packet must be located on the same subnet. For these reasons, if the network managed by E-Net-Manager consists of many subnets, a *Waker* must be placed in each subnet, so as to remotely turn on PCs in that subnet. A *Waker* is a host with a permanent connection to the Server, that receives wakeup commands from the Power Manager (e.g., when a power-on rule becomes active) and sends Magic Packets to target PCs.

C. *Client*

The Client part of the system consists of all PCs that are under the control of E-Net-Manager. Our system requires that a simple software module (*client module*) is installed on each PC. The client module is in charge of receiving and executing suspension and poweroff commands from the Power Manager. Its behavior is fully transparent to the user.

Specifically, whenever a static rule becomes active for a PC, or a remote-control request for a PC is received, the Power Manager sends a poweroff or suspension command to the PC, through the XMPP protocol. The client module is always connected to the XMPP server, thus, the command is immediately received. Before executing the command – in order to limit the intrusiveness of power management – the client module preliminary checks whether the user is still using the PC through the Activity soft sensor (see below). If the Activity soft sensor confirms that the user is no longer using the PC, the command is executed. Otherwise, the received command is just ignored or postponed, trasparently to the user. Specifically, commands derived from immediate rules are ignored. In case of extended rules, instead, the client module waits for the user to leave the PC. As soon as the Activity soft sensor determines that the user is no longer active – if the rule is still valid – the requested operation is executed. For instance, if an extended rule was created to turn a PC off between 6 and 9 pm but the user keeps using it till 7 pm, the client module receives the poweroff command at 6 pm but executes it after 7 pm.

D. *Soft sensors*

In order to detect the presence/absence of the user within/from the working area, as well as her/his activity, E-Net-Manager exploits some soft sensors, i.e., hardware or software tools that are already available in the environment or already used by the user and, thus, do not require setup or installation. In principle, any device or software tool that is able to provide information about the user activity and/or presence can be used as a soft sensor and integrated in the system. In the current implementation, we exploit *Bluetooth smartphones*, *attendance recorders* and software tools like *Google Calendar* and *PC activity monitors*. Specifically, E-Net-Manager switches off a PC when the user punches out, and, conversely, turns it on when the user punches in. Similarly, the user is supposed to be far from the workspace if no mouse/keyboard actions are detected, and, at the same time, an event in the calendar starts, or the Bluetooth smartphone is out of the coverage area of the PC. In these cases, the user's PC can be suspended. We provide below a short description of the considered soft sensors.

Attendance soft sensor. This soft sensor is effective for all those users that are requested to punch in (out) at the beginning (end) of the working day. Basically, whenever a user clocks in or out, a special message – containing the user id and the performed action (i.e., clock in/out) – is sent to the Power Manager. Since the user is entering/exiting his working area, the Power Manager will send an appropriate command to the user's PC (or corresponding Waker) in order to switch it on/off.

Bluetooth soft sensor. This soft sensor allows to detect the presence/absence of the user within the working area. The used approach is similar to that proposed in [13] and exploits the Bluetooth discovery mechanism. We assume that the user has at least one Bluetooth-enabled device, e.g., a smartphone, that she/he always brings with her/him. In addition, the PC must be equipped with a Bluetooth interface. Once the user has selected the Bluetooth device to use (through the graphical interface), the PC starts periodically sending *request Bluetooth messages* to this device, waiting for *response messages*. There is no need that the device is paired to the PC. If no answer is received, after a certain time interval, the smartphone is considered out of range. This means that the user is far from her/his PC (beyond the coverage area of Bluetooth) and the PC can be switched off. Thus, a pop-up is displayed for a short time, to give the user the opportunity to intervene and prevent the suspension operation. Without any user intervention, the PC is put in sleep state. The user can quickly resume it (manually) when she/he comes back, in just few seconds. The Bluetooth soft sensor mechanisms is shown in Figure 4.

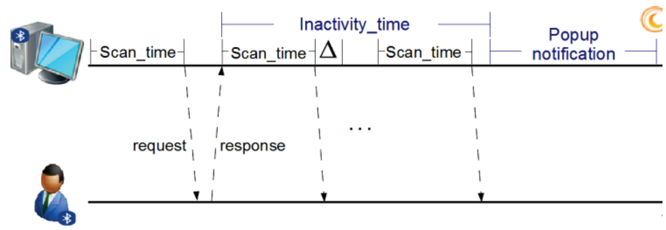

Fig. 4. Bluetooth soft sensor working scheme

Google Calendar soft sensor. The events stored in the user's calendar can be used to trigger appropriate power management actions. In particular, the user's PC can be automatically turned off when an event starts. To this end, while inserting an event to her/his calendar, the user has to enable the email reminder. This way, an email will be sent to the user's Gmail mailbox some minutes before the event. The Google Calendar soft sensor software simply performs a periodic check of such mailbox. That is, it downloads the emails from the Gmail server and parses the recently received emails to determine if they are Google Calendar reminders. In that case, the starting time of the event is extracted and a timer is activated. The timer expires some time (defined by the user) after the beginning of the event in order to put the PC in sleep mode.

Activity soft sensor. This soft sensor allows to detect whether the user is using her/his PC. This information is used to decide if a suspension or poweroff action can be executed without bothering the user. It is a software that constantly monitors the mouse/keyboard activity of the user. Basically, it stores when the user has performed the last action, i.e., a mouse movement or a key pressure. When a poweroff or suspension command is received (due to a static rule, a remote-control request, or other soft sensors) the software calculates the time past since the last user action. If it is over a certain pre-defined threshold (e.g., 5 minutes), the user is considered inactive and the command is executed. Otherwise, it is assumed that the user is still using the PC and the received

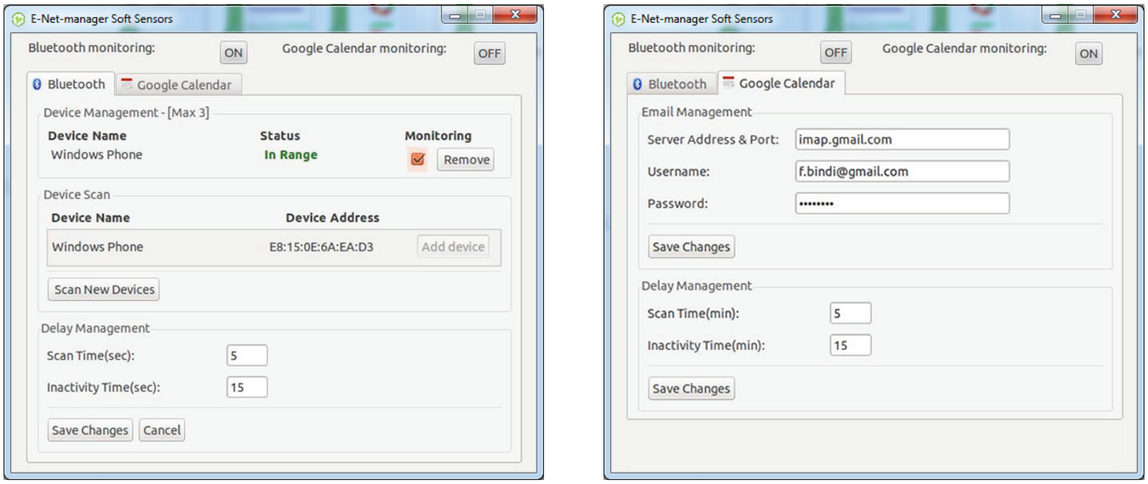

Fig. 5. Graphical User Interface for Bluetooth (left) and Google Calendar (right) soft sensors

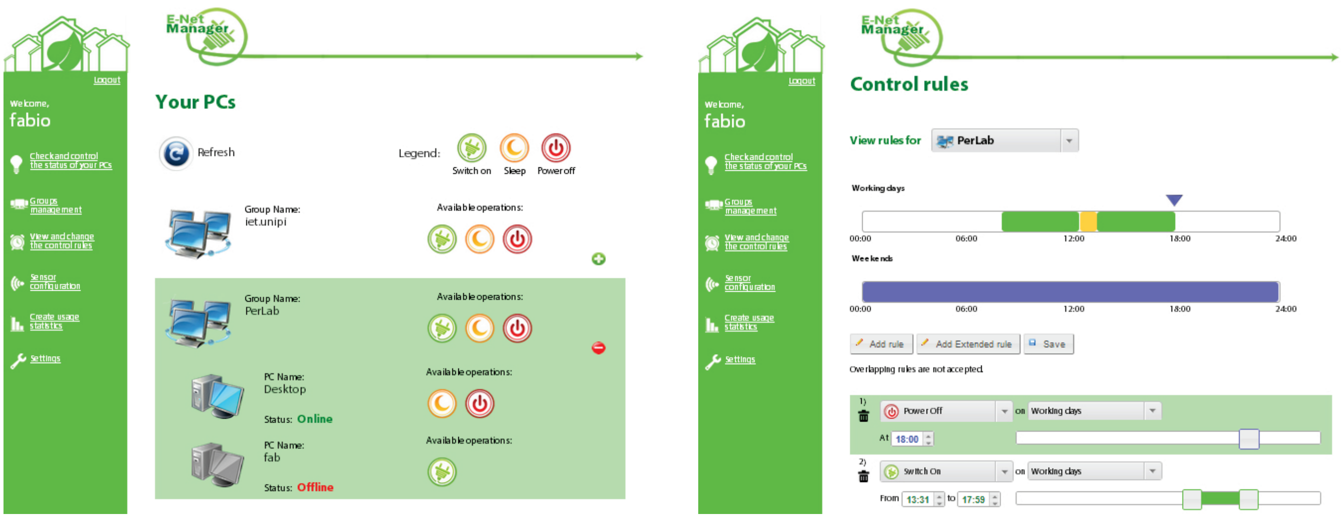

Fig. 6. Examples of webpages

command is just ignored or postponed, trasparently to the user.

V. Implementation

In this section, we provide some details about the system implementation.

A. *Server*

The server part of the system has been implemented as four distinct software processes (see section IV). This is a scalable solution, since the server part can be hosted by just one machine or by different machines according to the number of controlled PCs. In particular, we used *MySQL* as relational database management system. The web server we used is *Apache,* with the *mod_ssl* module installed. Hence, the site is completely hosted over HTTPS. Further details about the Web interface are provided in the specific subsection. The XMPP service is provided by *eJabberd*, a popular open-source XMPP server, written mainly in *Erlang* and distributed under the *GNU General Public License*. We have realized the Power Manager as a *multithreaded* program, written in C++. It uses the *Boost Asio* library to create timers, that are handled by multiple threads. It also uses the *Gloox* library to implement the necessary XMPP functionalities.

B. *Waker*

The Waker is the simplest part to implement. Basically, it is a process (written in C++) that – once started – registers itself to the Power Manager and communicates its own subnet address. Then, it waits for wakeup commands from the Power Manager and sends Magic Packets to target PCs.

C. *Client*

On each controlled PC, a simple software module – i.e., the *client module* – is required. This software can be downloaded from the E-Net-Manager website.

In order to associate the PC to the system, the client module requires some information during the installation wizard. Specifically the user is requested to enter her/his username and password or to create a new account whether she/he does not have one (it is needed to use the Web interface). The user must also specify a PC name. In addition, the client module automatically obtains the MAC address and the IP Broadcast subnet address of the network interface, since they are required for the WoL mechanism. Finally, before

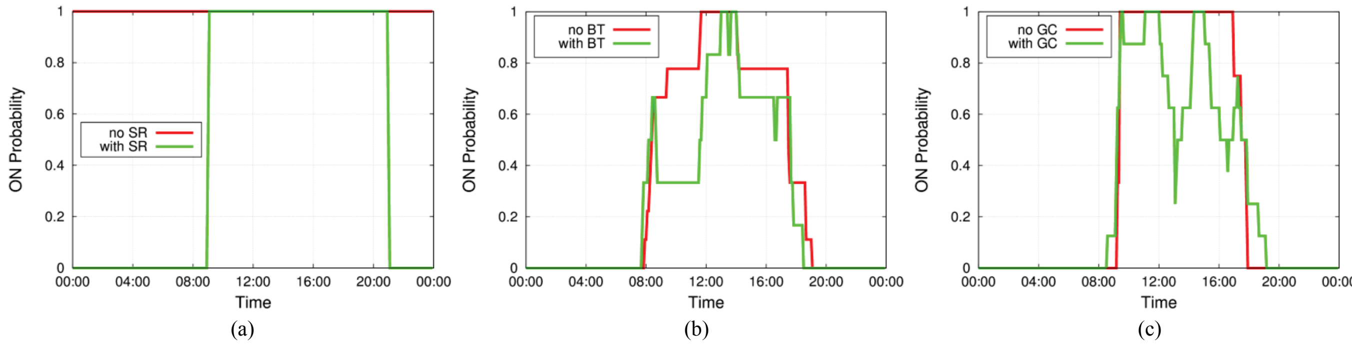

Fig. 7. PC activity in a working day with and without the usage of *Static Rules* (a), *Bluetooth soft sensor* (b) and *Google Calendar soft sensor* (c)

closing the installation phase, the client module automatically creates an XMPP account with the XMPP server, in order to enable the XMPP communication.

The client module is a *multithreaded (C++)* process. It exploits the *Boost* library for timers and the *Gloox* library for XMPP communications. It is executed in background at the PC startup (before the user login) and does not require any user interaction to work. Several versions have been implemented for the most common operating systems:

- *Windows (XP, Vista, 7, 8, and 8.1)*. In this case the client module is implemented as a *Windows Service*.
- *Linux (Ubuntu ≥ 12.04, Debian ≥ 6.0)*. The client module is implemented as a *Daemon*.

D. Soft sensors

In order to control Bluetooth, Google Calendar and Activity soft sensors, a specific software module is installed directly on the users' PCs, together with the Client module. Once again, we have implemented different versions, for several operating systems. The soft sensor module is a *multithreaded* process, written in *C++*, automatically launched at the operating system startup in Linux, or after the user login in Windows. A *Graphical User Interface (GUI)* – realized through the *GTKmm* library – has been provided, to allow the user to configure and enable the soft sensors (Figure 5). As regards soft sensors, some implementation considerations are necessary.

Bluetooth soft sensor. The message passing via Bluetooth interface has been implemented through the *bluez* library in Linux and the *Microsoft SDK API* in Windows.

Google Calendar soft sensor. In order to download emails from the Gmail server we have used the *Imap* protocol, together with *TLS* to crypt the communication.

PC Activity soft sensor. In Windows systems, we obtain the timestamp of the last input event through the *GetLastInputInfo()* system call. In Linux operating systems, instead, mouse and keyboard activity are periodically checked by reading the correspondent files in the */dev/* folder.

E. Web Interface

The user can interact with E-Net-Manager through a simple Web interface. All the web pages are written in *PHP* and *Javascript* (using *jQuery*). Users can log in to the website, using their accounts, and perform the following actions: (**i**) check the state of their PCs and remotely control them (Figure 6-left); (**ii**) create/organize groups of PCs; (**iii**) view/change the static rules defined for their PCs (Figure 6-right); (**iv**) view soft sensors current settings; (**v**) view usage statistics (daily and monthly). Whenever the user creates or modifies some rules or settings, the Database is updated and the Power Manager is notified, in order to make them promptly effective.

VI. Experimental Results

In order to measure the energy savings deriving from the usage of E-Net-Manager, we tested the system on a real environment in our university campus. We considered 20 PCs, belonging to 16 different users (13 PCs were running Windows, the others Linux). All the PCs were part of the same IP subnet, so that only a Waker was required.

We monitored the activity of the PCs for two weeks, so as to profile the user's normal behavior, i.e., without any active power management policy. Then, we asked the users to enable and use (for two further weeks) the power management features provided by E-Net-Manager, according to their needs and habits.

Since the server stores the energetic transitions of each associated PC (from powered on to powered off and vice versa), we were able to measure the utilization time of each PC, in order to assess the improvement in terms of energy consumption. Basically, starting from the gathered information, we have derived the probability that each PC was on in various moments of the day. Simply, for each considered time slice, we have calculated the fraction of times the PC was active over the total number of considered days. Therefore, we have compared the data obtained during the two weeks of profiling and the two weeks of usage of E-Net-Manager. For the sake of space, we will present only the results referring to working days, since they are more significant than holidays.

Obviously, the savings introduced by the system strictly depend on the users' behavior. For instance, great savings were obtained with those users that regularly leave their PCs always on. In this case, just setting few simple rules, it is possible to cut the wastes due to the PCs active during nights, weekends and holidays. Figure 7 (a) clearly shows that – for careless users – significant savings (50%) can be achieved with proper static rules for working days.

However, some users pay more attention to the energy issue, providing for switching off their PCs at the end of the working day. In this case, static rules do not introduce significant benefits, since users act on their own. However, thanks to soft sensors, E-Net-Manager is even able to significantly reduce the energy consumption of these virtuous users. Combining the presence information provided by all the soft sensors, even short idle times (e.g., due to a meeting or a break) can be detected. Thus, E-Net-Manager can provide a fine-grained power management. Figure 7 (b) and (c) show the improvements in PC utilization obtained by two different users using Bluetooth and Google Calendar soft sensors, respectively. Overall, considering all the involved users, the activity times (and, thus, the energy consumption) of PCs have been reduced by 9.4% and 10.3% using Bluetooth and Google Calendar soft sensors, respectively.

VII. Conclusions

In this paper we have presented *E-Net-Manager*, an automated system for power management of networked PCs in large environments. The innovative contribution of E-Net-Manager mainly consists in combining information provided by the users and data gathered from a number of soft sensors. Specifically, the system can decide when to switch off/on PCs basing both on static rules explicitly defined by the users, and on presence information derived by hardware and software already used by users for other purposes. For instance, E-Net-Manager leverages Google Calendar, Bluetooth smartphones, PC activity monitors and attendance recorders as soft sensors. These soft sensors are exploited to obtain presence/absence information so as to determine whether the user is currently near her/his PC. Through the use of soft sensors and user-defined rules, it has been possible to realize a low cost, low intrusive power management system. Experimental results, conducted on a real test bed in our university campus, clearly show that significant energy savings can be achieved even in the case of short occasional idle periods.

References

[1] W. Van Heddeghem, S. Lambert, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, "Trends in worldwide ICT electricity consumption from 2007 to 2012", *Computer Communications*, vol. 50, pp. 64-76, Sep. 2014.

[2] P. Bertoldi, B. Hirl, and N. Lab, "Energy Efficiency Status Report 2012 – Electricity Consumption and Efficiency Trends in the EU-27", *JRC Scientific and Policy Reports*, European Commission, Joint Research Centre, Institute for Energy and Transport, 2012.

[3] A. Kamilaris, D. T. H. Ngan, A. Pantazaras, B. Kalluri, S. Kondepudi, and T. K. Wai, "Good Practices in the Use of ICT Equipment for Electricity Savings at a University Campus", *Proc. of the 5th International Green Computing Conference (IGCC)*, Dallas, Texas, USA, Nov. 2014.

[4] iMinds, "D8.1. Overview of ICT Energy Consumption", FP7‑288021, EINS (Network of Excellence in InterNet Science) Consortium, 2013 [Online]. Available at: www.internet‑science.eu/sites/ internet‑science.eu/files/biblio/EINS_D8%201_final.pdf.

[5] National Energy Foundation (NEF), S. Karayi, "The PC energy report", 1E, London, 2007 [Online]. Available at: http://www.1e.com/ energycampaign/downloads/1e_reportfinal.pdf.

[6] National Energy Foundation (NEF), "Assessment of Potential for Energy Savings from PC Software Management", 1E, 2006 [Online]. Available at: http://www.1e.com/energycampaign/downloads/ 1ENEFReport.pdf.

[7] International Energy Agency (IEA), "More Data, Less Energy: Making network standby more efficient in billions of connected devices", 2014 [Online]. Available at: http://www.iea.org/publications/freepublications/ publication/moredata_lessenergy.pdf

[8] L. Chiaraviglio, and M. Mellia, "Polisave: efficient power management of campus PCs", *Proc. International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2010)*, Split,, Croatia, Sep. 23-25, 2010.

[9] B. Nordman, and K. Christensen, "Greener PCs for the Enterprise", *IEEE IT Professional*, vol. 11, pp. 28-37, 2009

[10] "Advanced Configuration and Power Interface Specification, revision 5.0", Hewlett-Packard, Intel Corporation, Microsoft, Phoenix Technologies, Toshiba, December 6, 2011 [Online]. Available at: http://acpi.info/DOWNLOADS/ACPIspec50.pdf

[11] R. K. Harle, and A. Hopper, "The Potential for Location-Aware Power Management", *Proc. International Conference on Ubiquitous Computing (UbiComp 2008)*, Seoul, South Korea, Sep. 21-24, 2008.

[12] M. Youssef, and A. Agrawala, "The Horus WLAN Location Determination System", *Proc. International conference on Mobile Systems, applications, and services (MobiSys 2005)*, Seattle, Washington, USA, 6-8 Jun. 2005.

[13] C. Harris, and V. Cahill, "Power Management for Stationary Machines in a Pervasive Computing Environment", *Proc Hawaii International Conference on System Sciences (HICSS 2005)*, Hawaii, USA, Jan. 3-6, 2005.

[14] S. Brienza, G. Anastasi, G. Lo Re, and M. Ortolani, "Energy Efficient Protocol Design", *Green Communications: Principles, Concepts and Practice* (P. Rost, A. Maeder, M. Meo, C. Verikoukis, K. Samdanis, Editors), John Wiley & Sons Inc., to appear (Nov. 2014). ISBN: 9781118759264.

[15] B. Nordman, and K. Christensen, "Improving the Energy Efficiency of Ethernet-Connected Devices: A Proposal for Proxying", White Paper, Version 1.0, Ethernet Alliance, October 2007.

[16] K. Christensen, and F. Gulledge, "Enabling Power Management for Network-Attached Computers", *International Journal of Network Management*, vol. 8, n. 2, pp. 120-130, Mar.-Apr. 1998.

[17] K. Christensen, B. Nordman, and R. Brown, "Power Management in Networked Devices", *IEEE Computer*, vol. 37, n. 8, pp. 91-93, Aug. 2004.

[18] C. Gunaratne, K. Christensen, and B. Nordman, "Managing Energy Consumption Costs in Desktop PCs and LAN Switches with Proxying, Split TCP Connections, and Scaling of Link Speed", *International Journal of Network Management*, vol. 15, n. 5, pp. 297-310, Sep.-Oct. 2005.

[19] R. Khan, R. Bolla, M. Repetto, R. Bruschi, and M. Giribaldi, "Smart Proxying for Reducing Network Energy Consumption", *Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2012)*, Genoa, Italy, July 8-11, 2012.

[20] S. Cheshire, "Method and Apparatus for Implementing a Sleep Proxy for Services on a Network", United States Patent N. 7,330,986, February 12, 2008.

[21] S. Nedevschi, J. Chandrashekar, B. Nordman, S. Ratnasamy, and N. Taft, "Skilled in the Art of Being Idle: Reducing Energy Waste in Networked Systems", *Proc. USENIX Symposium on Networked System Design and Implementation (NSDI, 2009),* Boston, Massachusetts, USA, Apr. 22-24, 2009.

[22] Y. Agarwal, S. Hodges, J. Scott, R. Chandra, P. Bahl, and R. Gupta, "Somniloquy: Augmenting Network Interfaces to Reduce PC Energy Usage", *Proc. USENIX Symposium on Networked System Design and Implementation (NSDI, 2009)*, Boston, Massachusetts, USA, Apr. 22-24, 2009.

[23] Y. Agarwal, S. Savage, and R. Gupta, "Sleepserver: a software-only approach for reducing the energy consumption of pcs within enterprise environments", *Proc. USENIX Annual Technical Conference (USENIXATC)*, pp. 22–22, Berkeley, CA, USA, 2010.

[24] T. Das, P. Padala, V. N. Padmanabhan, R. Ramjee, and K. G. Shin, "Litegreen: Saving energy in networked desktops using virtualization", *Proc. of the USENIX Annual Technical Conference (USENIXATC)*, 2010.

[25] Magic Packet Technology, White Paper, Publication# 20213, Rev: A, Amendment/0, November 1995.

[26] K. Kurowski, A. Oleksiak, and M. Witkowski, "Distributed power management and control system for sustainable computing environments", *Proc. International Conference on Green Computing (IGCC 2010)*, Chicago, Illinois, USA, Aug. 15-18, 2010.

[27] P. Saint-Andre, "Extensible messaging and presence protocol (XMPP): core", RFC 3920, Internet Engineering Task Force, Oct. 2004 [Online]. Available at: http://www.ietf.org/rfc/rfc3920.txt