INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT
Int. J. Network Mgmt., **8**, 120–130 (1998)

Enabling Power Management for Network-attached Computers

Power management is an emerging area of interest for network management. This article reviews current developments and describes methods for enabling power management in network-attached computers. © 1998 John Wiley & Sons, Ltd.

By Kenneth J. Christensen and Franklin 'Bo' Gulledge*

Introduction

Electronic data processing equipment consumes between 25 and 65 TWh/yr in the USA, representing between 5% and 20% of the electrical load in office buildings.[17] The 1990 total energy consumption for office equipment in the US commercial sector is described as 58.3 TWh with 5.9 TWh and 5.0 TWh resulting from personal computer (PC) system units and monitors, respectively.[8] The Environmental Protection Agency (EPA) Energy Star program for office equipment[4] was announced in 1992 to develop methods of reducing this large power consumption. The program is based on creating voluntary partnerships between the EPA and industry. In 1993 an Executive Order was issued requiring all US federal government agencies to purchase EPA Energy Star compliant computers, monitors and printers. A PC compliant with the Energy Star PC/Monitor Memorandum Of Understanding (MOU)[3] has the ability to reduce its power consumption during periods of inactivity. To earn an Energy Star logo, the maximum allowed power consumption following a specified period of inactivity is 30 W for the monitor and also 30 W for the system unit. A Department of Energy (DOE) sponsored study at the Lawrence Berkeley National Laboratory projects that the Energy Star program for office equipment will save from a worst-case 6 TWh/yr to a best-case 16 TWh/yr in the year 2000.[8] At $0.08 kWh, which is the 1995 approximate cost, this represents savings of $500 million to $1.3 billion to US businesses. Other countries such as Sweden[14] have programs similar to the EPA Energy Star progam.

Kenneth J. Christensen received his PhD from the North Carolina State University in 1991. He is currently an Assistant Professor at the University of South Florida, working in performance modeling of Gigabit Ethernet and power management of network-attached computers is research interests are in the areas of computer network systems and architectures. He has over fifteen conference and journal publications, seven US patents, and is a senior IEEE member. Hompage: http://www.csee.usf.edu/~christen
Franklin 'Bo' Gulledge is a graduate student at the University of South Florida pursuing an Msc in Computer Science. His research is in the area of power management of network-attached computers. Homepage: http://www.csee.usf.edu/~gulledge

**Correspondence to Kenneth J. Christensen, Department of Computer Science and Engineering, University of South Florida, 4202 East Fowler Avenue, ENB 118, Tampa, FLA 33620, USA.*

 CCC 1055–7148/98/020120–13017.50

A challenge with Energy Star compliance is that network connections are lost if a power-managed network-attached computer powers-off its processor. Network connections are maintained by a higher-layer protocol, such as TCP/IP, executing in the processor of the computer. Existing computers, based largely on 80486 and older Intel Pentium processors, are able to maintain network connections during low power operation by slowing down the processor clock. This slow clock mode is sufficient to both achieve the necessary 30 W of power consumption and maintain all network connections. Newer computers, for example those based on the Intel PentiumPro processor, usually cannot achieve 30 W operation with a slow clock mode. PentiumPro processors running at 200 MHz do not support a slow clock mode and consume in excess of 30 W alone without even considering the rest of the components in the computer.[7] Thus, newer computers must power-off their processors to achieve Energy Star compliance. This trend of higher performance and higher power consumption by processors is expected to continue. However, the power consumption of monitors, which today consume the majority of power for a PC, will decrease if flat-screen Liquid Crystal Display (LCD) monitors replace existing Cathode Ray Tube (CRT) monitors.[12]

Lost network connections can result in lost user data, for example missing a file update on a server in the case of a disconnected client. It also inconveniences users because they need to repeat logon sequences and re-setup their previous connections. Remote access by other users is also adversely affected. As a consequence, many network administrators disable Energy Star features on network-attached computers.[11]. A case study found only 11% of PC system units with Energy Star fully enabled and working[13]. It is estimated that 90% of all office PCs are attached to a Local Area Network (LAN).[12] A solution to the network connection problem, as an overall component of network management, would be very valuable as a means of discouraging the disabling of Energy Star features and thus promoting increased energy savings.

The remainder of this paper is organized as follows. The next section reviews power management of computers. The third section presents recently developed methods of 'waking-up' power-managed computers on a LAN. The fourth section describes the network connection problem and presents a proxy server solution. This section also describes a prototype proxy server that is being developed at the University of South Florida. Extensions and future directions to the prototype proxy server are described in the fifth section. The final section is a summary, followed by references.

Power Management of Computers

A number of case studies have shown that most of the time that PCs are powered-on they are not in use.[17] During the day there are long periods of user inactivity and an estimated 20% of all PCs remain powered-off and inactive during nights and weekends.[13] A PC with power management features can greatly reduce its power consumption during such periods of user inactivity. User inactivity is detected by power management hardware as an absence of keyboard activity or mouse movements for a preset time period, which is typically 15 or 30 minutes. It is estimated that power management, for both the system unit and monitor, can result in an annual savings of around $30 per year per PC[17]. The larger implications of power management in the context of the Energy Star program are presented in reference 8. It is estimated that in the year 2000 the total US energy consumption of PC system units, which disregards energy consumption of monitors will be:

- 9.2 TWh if Energy Star is not adopted
- 7.5 TWh for a most likely case of Energy Star adoption
- 3.0 TWh for an advanced case of Energy Star adoption

At $0.08 kWh, this corresponds to $140 million savings for US businesses if the most likely case achieved, $500 million for the advanced case. The most likely case assumes that 50% of PC system units have Energy Star features enabled and working, the advanced case assumes 100% enablement. A 1% increase in Energy Star enablement between the not-adopted and advanced cases would result in 62 GWh, or $5.0 million, in additional savings. For the year 2010 projected savings are higher as a result of an expected growth in the number of installed computers.

—Background on Power Management—

Power management of computers first appeared in laptop computers as a means of extending battery life for mobile users. Power management can be included as part of the firmware and operating system in a PC.[2,10] There are four components to power management:

(1) The ability to monitor activity levels of input devices such as keyboard and mouse
(2) The use of timer circuits to determine when to shift to a lower-power mode of operation
(3) The ability to communicate power management status to affected devices
(4) The ability to resume a powered mode of operation on the detection of activity

Figure 1 shows the control flow of power management and the sequence of events is approximately as follows. The firmware periodically sends request signals (2) to the operating system to begin power management. If the operating system is enabled for power management and no activity is detected from the application (3), the operating system passes a signal (4) to the firmware to begin a power management inactivity timer. After a specified time with no activity detected (1), the firmware will initiate power management by sending signals (5) to some, or all, of the processor, peripherals (e.g. hard disk), and video card. These devices will then enter a low-power operating mode. An activity interrupt (1) or wake-up interrupt (6) will cause the firmware to signal (5) a return to an active and powered mode of operation.

The processor core, which includes the processor and system bus, is one of the components that can be managed. Two modes of operation are defined for the processor core:

- *Slow clock*—reduced-speed operation for the processor and system bus with reduced power consumption
- *Stopped clock*—processor and system bus are almost completely off and only a hardware interrupt can cause a restart (of the processor and system bus).

For newer high-performance processors, a slow clock mode is not supported. Emerging standards in power management, driven by a view of a PC as consumer appliance[10] support only a stopped clock mode.

'Waking-up' Power Managed Computers on a LAN

Computers that are in a stopped clock mode, or 'sleeping' state, cannot be remotely accessed due

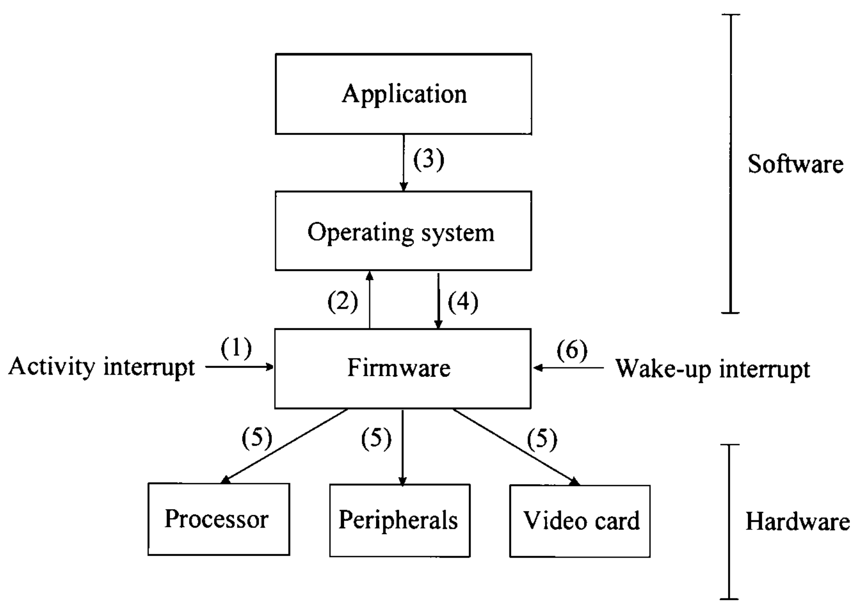

Figure 1. Power management components and control flow.[12]

to their LAN adapters being powered-off. A typical LAN adapter receives its power from the system bus, which is powered-off when in a sleeping state. A method of remotely 'waking-up' sleeping computers, primarily for purposes of remote management, has been developed. The method, called Magic Packet,[1] requires special LAN adapters to be installed in power managed computers that support both auxiliary power connections and an external wake-up interrupt signal. The special LAN adapters have three new functions:

Computers that are in a stopped clock mode, or 'sleeping' state, cannot be remotely accessed due to their LAN adapters being powered-off.

- Connection to external auxiliary power to allow partial adapter operation to continue when the system bus powers-off
- Connection to a wake-up interrupt line, which is external to the system bus
- The ability to receive all packets when operating on auxiliary power and recognize a special packet, called a Magic Packet. On recognition of a Magic Packet, a wake-up interrupt signal is generated.

A Magic Packet is defined as any packet that in a data field contains the adapter Media Access Control (MAC) address consecutively repeated 16 times. Preceding the sequence of MAC addresses is a synchronization field of 6 bytes of 1's. A Magic Packet can be sent with any protocol, for example TCP/IP, and can thus be delivered across any routed or bridged network. Figure 2 shows a Magic Packet LAN adapter. Within the MAC controller is a sequence detector circuit for detecting a Magic Packet. When in auxiliary power mode, the receiver circuitry in the physical (PHY) layer components and the sequence detector are powered, but the remainder of the adapter does not need to be powered. Powering the adapter receiver circuitry may require only several 10's of milliwatts. On detecting a Magic Packet, an interrupt signal is sent on the external wake-up interrupt line. The wake-up interrupt then causes the power-managed computer to begin its wake-up sequence.

The OnNow initiative[9,10] extends the Magic Packet approach to allow a LAN adapter to define a filter to match certain 'interesting' packets. For example, filters can be defined to generate a wake-up event on the detection of an Address Resolution Protocol (ARP) packet or TCP/IP Synchronize (SYN) packet intended for the sleeping PC. ARP packets broadcast by a TCP/IP host are used to find the LAN address of another TCP/IP host. A SYN packet is the first packet in a TCP/IP connection request. Thus, the OnNow approach to network devices may allow existing protocol packets to force a wake-up and thus not require applications or protocols to change if the PC can wake-up fast enough. The Magic Packet approach requires an application or protocol to first send a Magic Packet before establishing a standard TCP/IP connection for normal packet exchange. Both approaches require specialized LAN adapters (and computers with auxiliary power and external wake-up interrupt), as shown in Figure 2. The large majority of LAN adapters currently being installed do not support Magic Packet, and no adapters are yet available that comply with the emerging OnNow filtering specifications.

The Network Connection Problem and the Proxy Server Solution

A communications protocol, such as TCP/IP, forms of logical connection between two communicating computers. The purpose of a connection is to assure that transmitted packets arrive correctly at the destination. A receiving computer helps detect lost packets by sending a numbered acknowledgment (to the sending computer) for each received packet. If the sending computer does not receive an acknowledgment within a predetermined time period following a packet transmission, it assumes the packet is lost and resends the packet. If after a predetermined number of retransmissions, no acknowledgment has been received, the sending computer logically disconnects the connection. That is, it assumes that the receiving computer or network is down. When no application data is being sent, periodic 'are you alive' packets are sent (one packet roughly every few seconds) to insure that the connection is still viable. These packets must also be acknowledged.

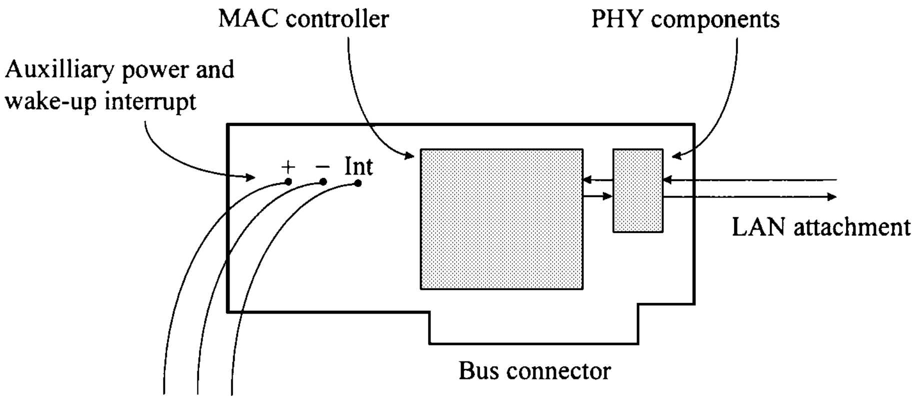

Figure 2. LAN adapter with auxiliary power and wake-up capability.

When a connection is disconnected, any associated applications 'clean-up' resources dedicated to that connection. The clean-up entails freeing all resources associated with the disconnected connection. For example, if a user on a client computer A is doing work on a server computer B, any unsaved work on computer B would be lost if the connection between A and B is disconnected. Further, to re-establish the connection at a later time the user at A would have to go through a time-consuming logon sequence.

If a power-managed computer with established network connections enters a stopped clock mode, protocol processing is also stopped. The LAN adapter, unless it has auxiliary power connections, is disconnected from the LAN and the higher-layer protocols executing in the system processor are halted. This results in disconnected network connections. The existing solution to the network connection problem requires a power-managed computer to identify that it has open connections and not enter a stopped-clock mode, and this prevents it from reducing its power consumption. This problem has been cited by network administrators as the reason they disable Energy Star features in LAN-attached PC's[11].

—Using a Proxy Server to Maintain TCP/IP Connections—

TCP/IP is the *de facto* standard higher-layer protocol for network communications. Ethernet and Token Ring are the two most common lower-layer LAN protocols. A solution to the network connection problem that requires TCP/IP or standard LAN adapters to change is probably not viable. For example, requring LAN adapters to execute higher-layer protocols, while also operating on auxiliary power, is unreasonable. By understanding how TCP/IP establishes and maintains a network connection, it can be shown that proxy server can be designed to maintain client–server TCP/IP connections even when a client is physically disconnected from the LAN.

TCP/IP connections use a client–server architecture. For peer-to-peer networking, one of the peers acts as the server, the other as the client. A server listens for requests for connections from clients. Once a client request is detected, a three-way handshake takes place to establish the connection. When the connection has been established, data can be transferred from client-to-server and server-to-client. Data transfer is assured via receiver acknowledgments for all received TCP data segments. If acknowledgment of a packet is not received by the sender, it will retry the 'lost' data segment for a fixed number of times before disconnecting the connection. At any time during a connection, the client or the server can advertise a 'zero window' condition. A zero window condition disallows a sender from sending data until a TCP segment advertising a non-zero window has been received. When a sender is in a zero-window state it has no knowledge if the receiver is maintaining its connection or if a non-zero window advertisement has been sent and lost. Thus, the sender transmits TCP probe segments, or packets, on a periodic basis to the receiver. If the probe packets are not acknowledged, which implies that the receiver is physically disconnected, then the connection is dropped. This probing of zero windows must be supported by a compliant TCP implementation.[15,16] Figure 3 shows the flow between a client and a server and illustrates a client-to-server zero window advertisement (1) followed by successive server-to-client probe packets (2). Each of the probe packets must be acknowledged (i.e. an acknowledgment packet, not shown in Figure 3, flows from client-to-server for each received probe packet).

A TCP zero window condition can be maintained indefinitely as stated in RFC 1122 ('Requirements for Internet Hosts—Communications Layers'):[15]

> A TCP MAY keep its offered receive window closed indefinitely. As long as the receiving TCP continues to send acknowledgments in response to the probe segments, the sending TCP MUST allow the connection to stay open.

The basis of the solution for the network connection problem is for a client preparing to enter a stopped-clock mode to advertise a zero-window on all its connections and then have some other computer respond to probe packets sent to the now sleeping client. The 'other computer' is a designated power management proxy server that always remains fully powered-up.

A power-management proxy server computer, which is really a 'proxy client' in the context that this computer manages connections for a power-managed sleeping client, is shown in Figure 4. The server shown in Figure 4 could also be a peer client. The protocol flow for entering a power managed state is as follows:

(1) A client inactivity timer expires and the client prepares to enter a low-power state. This includes disconnecting from the network.
(2) The client advertises a zero window on all open connections, (1) in Figure 4, and also notifies the proxy server that it is disconnecting from the network.
(3) The server closes the connection window blocking any further transmission of data packets, the server then begins to send periodic probe packets (2).
(4) The proxy server receives the probe packets addressed to the client and sends acknowledgments to the server (3).

The protocol flow for 'waking-up' is as follows:

(1) A client detects activity, which causes a transition to full-powered operation.
(2) The client notifies the proxy server that it is ready to resume it connection, the proxy server ceases to acknowledge packets sent to the client's address.
(3) The client advertises a non-zero window to the server.
(4) Data transfer can resume on the same connection.

The proxy server must be located on the same TCP/IP 'subnet' as the client. A subnet is usually a single LAN. An IP router delivers packets only to the specific subjet on which the client is located. Thus, this would typically mean that the proxy server must be attached to the same LAN as the managed clients. When a power-managed client goes to sleep the proxy server must be able to receive incoming packets originally destined for this client. If the proxy server is located on a different subnet than the client, it would not be able to 'see' the packets destined for the client.

—A Prototype Power Management Proxy Server—

The prototype power management proxy server and client are being developed for the Linux

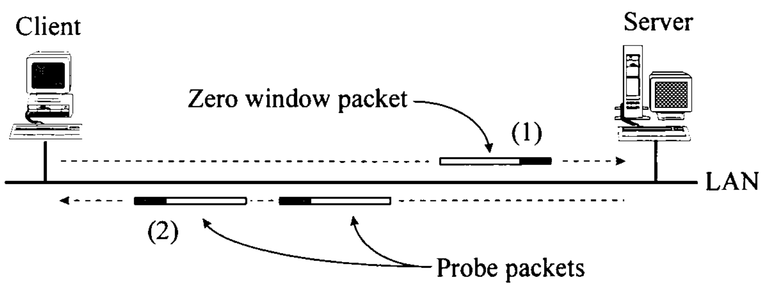

Figure 3. Client–server TCP flow showing probe segments.

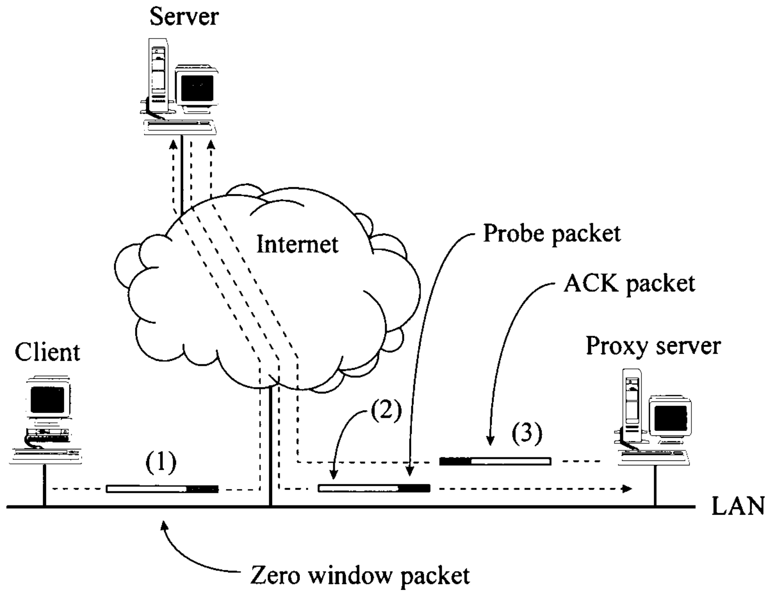

Figure 4. Client–server-proxy server TCP flow showing proxy server handling probe packets.

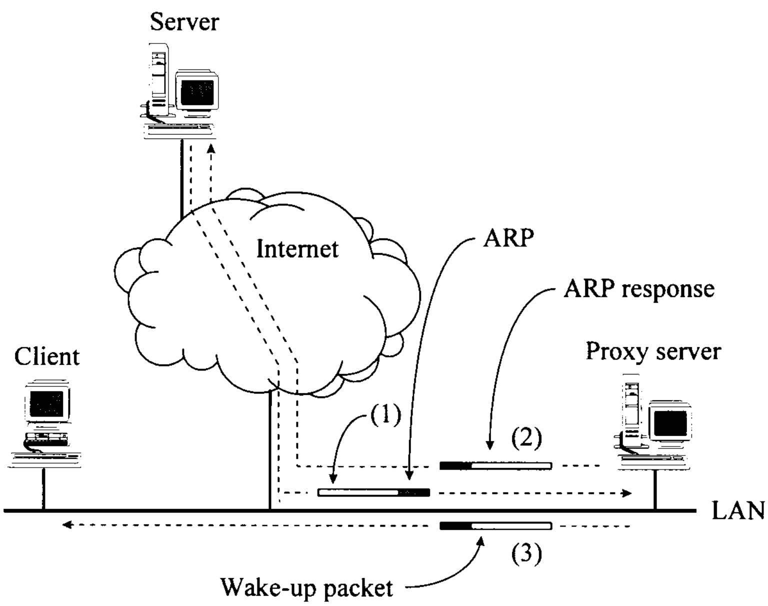

Figure 5. Proxy server initiating a Magic Packet wake-up of a sleeping client.

operating system. Unlike commercial operating systems, Linux has full source code available making it possible to make and test changes with a real operating system and its associated TCP/IP protocol. The prototye client simulates a power management event by a user-initiated action, not by an actual PC firmware-detected event. The prototype client does not power down, but operation of its TCP protocol suspends until the user initiates a simulated wake-up event which then reactivates the TCP protocol on the client. When the user initiates a simulated power management event on a client the TCP/IP connection is 'handed-off' to the proxy server. The following occurs for all client TCP/IP connections: the receive_window_size field for each Transmission

Control Block (TCB), is set to zero; a segment advertising a zero-window size is sent; the proxy server is sent a message that contains the TCB information for those connections to be proxied; the client's TCP protocol suspends and no longer responds to any network activity until reactivated by the user, and the proxy server promiscuously listens for packets bound for the suspended client. Any 'are you alive' messages are acknowledged by the proxy server and zero-window advertisements are sent using the proxied clients address in the IP source field. When the user ends a simulated power management event at a client, the following occurs for all the client's connections: the receive_window_size field for each TCB, is set to a non-zero value; a segment advertising a non-zero-window size is sent; the proxy server is sent a message to cancel proxy services; the client's TCP protocol resumes normal operation, and the proxy server cancels its proxy services for the client. Figure 6 shows the pseudocode for the power-managed client, Figure 7 for the proxy server.

Normally, TCP coordinates the activities of transmission, reception and retransmission for TCP connections through the TCB data structure. This data structure contains all the information about the TCP connection, including IP addresses, port numbers of the end-points of the connection, the round-trip estimate, sent and received data, acknowledgment and retransmission tracking information, and any statistics about the connection. This data structure is the basis for managing a TCP connection and therefore the key to power management. The prototype proxy server accomplishes proxying by having a copy of the proxied client's TCB, listening promiscuously for packets bound for the client, and responding with zero-window advertisements as if it was actually the client.

Extensions to the Proxy Server

The proxy server described in the previous section can be extended to support additional TCP/IP protocol capabilities on the behalf of a sleeping client. The first extension is to support ARP, that is, a proxy server responds to an ARP on behalf of a sleeping client. This ARP response allows the TCP/IP host sending the ARP to resolve the LAN address of the sleeping client. The second extension is to support incoming TCP/IP connections where the proxy server can intercept and respond to incoming connection requests on behalf of the sleeping client. Application-layer proxy services can also be implemented to handle incoming application-level requests for a sleeping client. This could apply to the emerging Internet 'push' applications. An existing application that already can use a proxy server is email. For example, the Eudora email program[5] can be set up to allow a single computer to handle e-mail for many client computers. If these client computers are awake, then the email is automatically forwarded to the appropriate client. If a client computer is sleeping, then the email proxy server holds the email until a later time when the client wakes up. If the client does not wake up, for example if it has been permanently removed from the network, then the e-mail proxy server 'bounces' the e-mail back to the original sender.

A power management proxy server can play an important role with the computers that support wake-up capabilities such as Magic Packet or OnNow packet filtering. In the case of Magic Packet wake-up, a proxy server can translate an existing protocol packet (e.g. an ARP) into a Magic Packet for the appropriate station. In the case of OnNow packet filtering and if the sleeping client cannot wake up quickly, in several milliseconds as opposed to several seconds, the proxy server can temporarily handle ARP responses and connection requests until the client is fully operational. Figure 5 shows a proxy server handling network requests (1) and (2) for a sleeping client and then initiating a Magic Packet wake-up (3) of the sleeping client. As in Figure 4, the server could also be a peer client.

The prototype proxy server is a standalone TCP/IP host implemented in a Linux PC. Dedicating a computer entirely for power management is probably not practical. The functions a proxy server provides can be co-located in existing network devices relieving the need for a dedicated computer. However, as described in the previous section the proxy server function must be on the same subnet as its managed clients. This restricts the devices into which the proxy server function can be located. Possible devices into which power management proxy serving can be integrated include LAN hubs in wiring closets and IP routers.


```
suspended = false
WHILE (suspended == FALSE) DO
  IF (suspend request received) THEN
     suspended = TRUE
     FOR (all connections) DO
        set TCB receive_window_size = zero
        send zero-window advertisement
        send TCB information and proxy request to proxy server
     ENDFOR
     WHILE (suspended == TRUE) DO
        IF (reactivate request received) THEN
           suspended = FALSE
           FOR (all connections) DO
              set TCB receive_window_size = non-zero
              send non-zero-window advertisement
              send message to proxy server to cancel proxying
           ENDFOR
        ELSE
           sleep x seconds
        ENDIF
     ENDWHILE
  ELSE
     . . .
     Normal TCP processing
     . . .
  ENDIF
ENDWHILE
```

Figure 6. Pseudocode for prototype power-managed client.

—Integrating Power Management Functions into TCP/IP and Network Applications—

Power management capabilities have been integrated into PC hardware, firmware, and operating systems. The next step must be to develop and integrate such capabilities into network protocols and network applications. For example, a client computer preparing to enter a sleeping state (e.g. due to inactivity) could advertise a zero window on all its existing connections and declare a 'sleep period' of N seconds. The N seconds represent the programmed sleep period of the computer, at which time it would wake up to handle any network-related tasks and then possibly return to a sleep state. A new TCP/IP packet could be defined to accomplish a 'sleep(N)' where during the sleep period the connection state would be maintained at the server (or peer client) and no probe packets would be sent. At the expiration of the sleep period, the server or peer client resumes the connection probing. Figure 8 shows the sleep(N) and delayed probing. The sleep(N) approach does not solve the problem of incoming network requests to a client. However, a client supporting OnNow packet filtering and a sufficiently fast wake-up time could detect and handle incoming requests.

Summary

Power management is an important part of network management for controlling operational costs of computing devices, especially PCs. Power


```
WHILE (proxying) DO
  Listen promiscuously
  FOR (each packet received) DO
      IF (packet destination == a proxied client) THEN
          send zero-window advertisement to source address
      ELSEIF (packet destination == proxy server) THEN
          IF (message requesting proxy service) THEN
              add client address to proxy list
              begin proxying for client
          ELSEIF (message canceling proxy service) THEN
              remove client address from proxy list
              end proxying for client;
          ELSE
              . . .
              Normal TCP processing
              . . .
          ENDIF
      ELSE
          do nothing
      ENDIF
  ENDFOR
ENDWHILE
```

Figure 7. Pseudocode for prototype power management proxy server.

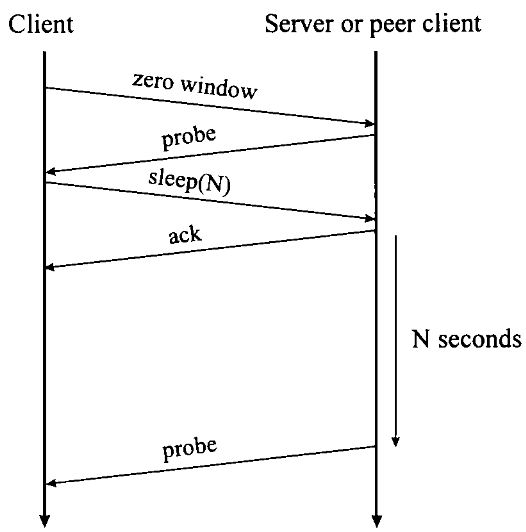

Figure 8. Packet flow for proposed TCP/IP zero-window with 'sleep(N)' and delayed probe.

management also has significant global environmental implications, for example reduced consumption of electricity translates into less air pollution from power plants. It is the environmental implications of power management that caused the EPA, and similar organizations in other countries, to produce power management guidelines and promote legislation that encourages use of these guidelines. High-end desktop computers require a complete shutdown of their processor to achieve Energy Star low-power operation. This complete processor shutdown causes network connections to be lost. To encourage continued use of Energy Star features in networked PCs, a proxy server to handle TCP/IP connections on the behalf of sleeping clients can be implemented. The functions of this proxy server can be extended to other TCP/IP capabilities, such as the ARP process, TCP/IP connection requests, and to applications.

Power management capabilities have been integrated into PC hardware, firmware, and operating systems.

Finally, future work can address additions to the TCP/IP protocol to directly support power management. With the very large quantities of PCs and the growing percentage of electricity that they consume, the methods described in this article represent an important direction for future research and development.

Acknowledgments

Thanks are due to Bruce Nordman and Jeff Harris, both from the Lawrence Berkeley National Laboratory, for their helpful and insightful comments during the course of this project.

References

1. *AMD – Magic Packet Technology*, Advanced Micro Devices, 1996. URL: http://www.amd.com/html/products/ind/overview/20212d.html
2. *Advanced Power Management (APM), BIOS Interface Specification*, Revision 1.2, Intel and Microsoft, February 1996. URL: http://www.intel.com/IAL/powermgm/apmv12.pdf
3. *PC/Monitor MOU – Version 2.0*, United States Environmental Protection Agency, October 1995.
4. *U.S. EPA Energy Star Office Equipment Program*, URL: http://www.epa.gov/office.html
5. Eudora Email Program, Qualcomm, Inc. URL: http://www.eudora.com/
6. J. Harris, Introduction to energy saving in office equipment, Nutek Energy Efficient Office Equipment and Home Electronics International Seminar, Panel Summary, September 1996. URL: http://eff.nutek.se/seminar/docs/J_Harris.html
7. *Pentium Pro Processor at 150-MHz, 166-MHz, 180-MHz, and 200-MHz*, Intel Corporation, 1996. URL: http://www.intel.com/design/pro/datashts/24276903.pdf
8. J. Koomey, M. Cramer, M. A. Piette, and J. Eto, Efficiency improvements in U. S. office equipment: Expected policy impacts and uncertainties, *LBL–3738*, Lawrence Berkeley Laboratory, December 1995.
9. *Device Class Power Management Reference Specification – Network Device Class*, Microsoft and Advanced Micro Devices, V 100, 1997. URL: http://www.microsoft.com/hwdev/download/netpmspc.doc
10. *OnNow and ACPI: Introduction and Specifications*, Microsoft Hardware Development—New PC Technologies, 1997. URL: http://www.microsoft.com/hwdev/onnow.htm
11. *Watchdog Dropped Workstations*, Novell Technical Information Document, Document ID 2906867, April 1996.
12. B. Nordman, M. A. Piette, K. Kinney, and C. Webber, User guide to power management for PCs and monitors, *LBNL-39466*, Lawrence Berkeley National Laboratory, January 1997.
13. B. Nordman, M. A. Piette, and K. Kinney, Measured energy savings and performance of power-managed personal computers and monitors, *Proceedings of the 1996 ACEEE Summer Study on Energy Efficiency in Buildings*, August 1996.
14. NUTEK, Swedish National Board for Industrial and Technical Development, Department of Energy Efficiency. URL: http://eff.nutek.se/engelsk.html
15. R. Braden (ed.), *Requirements for Internet Hosts—Communication Layers*, Internet Engineering Task Force, *Request For Comments 1122*, October 1989. URL: http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1122.txt
16. W. Stevens, *TCP/IP Illustrated*, Volume 3, Addison-Wesley, Reading, Massachusetts, 1996.
17. D. Tiller and G. Newsham, Switch off your office equipment and save money, *IEEE Industry Applications Magazine*, **2**, No. 4, 17–24, July/August 1996.

■

