Cooperative Proxying: An Approach to Reduce Network Energy Waste

Bolla, R., Giribaldi, M., Khan, R., & Repetto, M. (2013). Cooperative Proxying: An Approach to Reduce Network Energy Waste. IEEE INFOCOM 2013, Turin, Italy.

Cooperative Proxying: An Approach to Reduce Network Energy Waste

Raffaele Bolla* , Maurizio Giribaldi† , Rafiullah Khan* and Matteo Repetto‡
* *DITEN Dept. University of Genova (UNIGE), 16145 Genova, Italy,* Email:{raffaele.bolla, rafiullah.khan}@unige.it
† *Infocom s.r.l, P.zza Alessi 2/7, 16128 Genova, Italy,* Email: maurizio.giribaldi@infocomgenova.it
‡ *CNIT, Research Unit of University of Genova, 16145 Genova, Italy,* Email: matteo.repetto@cnit.it

***Abstract*—A network connected host is expected to generate/respond to applications and protocols specific messages. Billions of Euro of electricity is wasted to keep idle hosts powered up 24/7 just to maintain network presence. This short paper describes the design of our cooperative Network Connectivity Proxy (NCP) that can impersonate sleeping hosts and responds to packets on their behalf as they were connected and fully operational. Thus, NCP is in fact an efficient approach to reduce network energy waste.**

***Index Terms*—Green networking, energy efficiency, network connectivity proxy, power measurement.**

I. Introduction

Low-power states are effective mechanisms for reducing network energy waste by PCs but their use is hindered just because of the need to maintain network presence [1]. A low-power entity called "Network Connectivity Proxy" (NCP) can be designed that impersonates the hosts during idle periods and allows them to sleep [2]. It hides sleep state of covered hosts and make them appear as fully operational and connected [3]. The NCP maintains host's: *(i) Link Layer Presence* e.g., responding to ARP requests, *(ii) Network Layer Presence* by impersonating its IP addresses through gratuitous ARP and replying to network presence and management packets e.g., ICMP ping, DHCP etc, *(iii) Application Layer Presence* by responding to routine application messages.

Hosts should register their desired behaviors for different applications, protocols and packet types at the NCP before entering into sleeping state. The NCP then starts intercepting network traffic addressed to sleeping hosts and takes appropriate actions [4]. It wakes up the hosts only when their resources are required e.g., a new TCP connection attempt. Moreover, the NCP could buffer packets to give enough time to the hosts to restore their full power state and to avoid packet loss. This short paper briefly presents the NCP architecture and main functionality of different components.

II. Design of Network Connectivity Proxy

The NCP service has three basic requirements: (i) Awareness of host's power state, (ii) Ability to sniff and process packets addressed to sleeping hosts and (iii) Knowledge about the host's registered rules. The generic reference architecture of the NCP is depicted in Fig. 1. The NCP provides a set of rules that hosts can register. Rules specify actions for each received packet e.g., direct respond, discard, buffer, wake up sleeping host etc. The NCP activates the rules once the host switches to a low power mode and deactivates them when it wakes-up. The packet filtering block is responsible to sniff and identify packets addressed to the sleeping hosts based on header information (e.g., source and destination IP and port numbers, protocol-specific flags and packet content pattern matching). The packet processing block performs specific actions for each received packet based on the rules registered by sleeping host. The NCP design has three main requisites: rules structure, traffic diversion and packet classification engines.

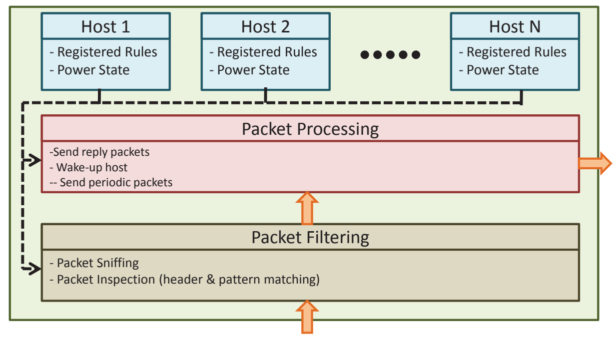

Figure 1. A generic NCP architecture.

A. Rules Structure

NCP rules can be classified into *(i) Network connectivity:* protocols that maintain host presence e.g, responding to ARP and ping requests, DHCP lease renewal etc, *(ii) Packet management:* dealing with packets that require some response from the host, *(iii) Heart-beating:* sending periodic messages just to refresh some information e.g., keeping alive a TCP connection, keeping active a soft state, etc.

Each rule is composed of four parts: host identification, power state, filter specification and an action. Universally Unique Identifier (UUID) is a better choice for host identification instead of IP/MAC address as hosts may have multiple NICs. The power state indicates when a rule should be activated. The filter identifies the packets, based on their header and/or data fields, for which the rule applies. It consists of a set of conditions which may be based on IP address, port number, network protocol, payload pattern etc. Finally, the action specifies the NCP behavior when a packet matches the filter specification.

Table I
SUMMARY OF BASIC NCP RULES.

		ARP	Ping	DHCP	WoC	WoP	SRoP
Type	Network Connectivity	X	X	X			
	Packet Management				X	X	X
	Heat-beating			X			
Action	Wake-up the host				X	X	
	Send predefined packet						X
	Build packet	X	X	X			
	Build packet by template						

NCP actions can be classified into four types: *(i) Wake-up the host:* when connection attempt at specific protocol and port is received (*Wake on Connection (WoC)*) or when received packet matches specific pattern (*Wake on Packet (WoP)*). *(ii) Send predefined packets:* send a given reply when received packet matches specific pattern (*Send Reply on Packet (SRoP)*). Hosts create the reply packets in advance and load them at NCP during rule registration. *(iii) Building packets:* the NCP creates reply packet on its own. *(iv) Building packets by templates:* the NCP builds packets starting from templates supplied by hosts. Table I shows the basic rules our NCP design provides so far.

B. Traffic Diversion

A basic prerequisite for NCP operation is the capability of getting traffic addressed to sleeping hosts. This obviously happens if the NCP lays on the path between each of its client devices and all of their peers, but this is not always feasible and also filtering could slow down packet forwarding operations if the traffic volume is high. Thus, traffic diversion mechanism is necessary to redirect traffic addressed to sleeping hosts towards the NCP.

Traffic diversion in local networks implies binding the NCP's MAC address to the IP addresses of sleeping hosts. Thus, the NCP's MAC replaces host's MAC when the host transition into sleep state and host's MAC is brought back when it wakes up. This job is accomplished by using "Gratuitous ARP", an unsolicited ARP message that holds the IP address and its associated MAC address. The gratuitous ARP enables the NCP to *(i)* answer ARP requests on behalf of sleeping hosts providing its own MAC address, *(ii)* update ARP caches of local hosts when its clients enter or exit the low-power mode to avoid sending packets to obsolete MAC addresses in caches.

C. Packet Filtering/Classification

Packet classification is the task of filtering and inspecting packets in order to pick up those matching any of the currently registered active rules. It turns out to be a critical task when the traffic load is high compared to the processing power of the device hosting the NCP service. These issues can be overcome by: (i) limiting the traffic load (handle only traffic that is addressed to sleeping hosts), (ii) using hardware based packet classification.

Software classification engines e.g., Berkeley Packet Filters, RF_RING, Pcap libraries etc. are flexible, portable and cost-effective solutions under low traffic load. However, hardware packet classifiers e.g., TCAM based classifiers come with several benefits: performance improvement, scalability and additional power savings. Packet classification is usually based on simple logical structures (decision trees, finite state machines) which are easy to realize in hardware. The CPU can thus run in low power modes until packets matching any rule are found.

Table II
ANNUAL SAVINGS WORLD-WIDE.

	World-wide Savings			
	TWh/year		Billion Euro/year	
	Always ON	Sleep when idle	Always ON	Sleep when idle
Desktop Computers	329.84	94.34	72.56	20.76
Laptops	20.26	6.19	4.46	1.36

III. EXPECTED ENERGY SAVINGS

Table II forecasts the global energy savings if NCP was fully deployed considering 815 million desktop PCs and 164 million laptops connected to the Internet. It uses the default data from Energy Star office equipment calculator that uses about 6.5 hours/day as average PCs usage and 17.5 hours as idle period. Thus, the global energy savings will account to almost 250 TWh annually. Similarly, considering 22 cent/kWh as the average electricity cost in Europe, the NCP has potential to provide almost 55 Billion Euro savings annually.

IV. CONCLUSIONS

The NCP is a quite useful approach that maintains the virtual presence of network hosts and allows them to sleep during idle periods. This short paper has addressed the generic NCP requirements, its basic functionalities and a reference architecture. It briefly presents our NCP design and focuses on three main components: rules structure, traffic diversion and packet classification engines. The NCP has potential to provide about 60% network energy savings depending on the hosts time usage model.

ACKNOWLEDGMENTS

This work was partially supported by the European Commission in the framework of the FP7 ECONET (low Energy COnsumption NETworks) project (contract no. INFSO-ICT-258454).

REFERENCES

[1] R. Khan, R. Bolla, M. Repetto, R. Bruschi, and M. Giribaldi, "Smart Proxying for Reducing Network Energy Consumption," in *IEEE International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, July 2012.

[2] K. Christensen, P. Gunaratne, B. Nordman, and A. George, "The Next Frontier for Communications Networks: Power Management," in *Computer Communications*, Vol. 27, No. 18, pp. 1758-1770, Dec. 2004.

[3] M. Jimeno, K. Christensen, and B. Nordman, "A Network Connection Proxy to Enable Hosts to Sleep and Save Energy," in *IEEE International Conference on Performance, Computing and Communications Conference (IPCCC)*, Dec. 2008.

[4] R. Khan, R. Bolla, and M. Repetto, "Design of UPnP based Cooperative Network Connectivity Proxy," in *The Second IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT 2012)*, Oct. 2012.