Enabling an Energy-Efficient Future Internet Through Selectively Connected End Systems

Mark Allman[†], Ken Christensen[‡], Bruce Nordman[¶], Vern Paxson[†,¶]
[†]International Computer Science Institute, [‡]University of South Florida
[¶]Lawrence Berkeley National Laboratory

This material is based on work supported by the National Science Foundation under grant numbers 0721933 and 0721858.

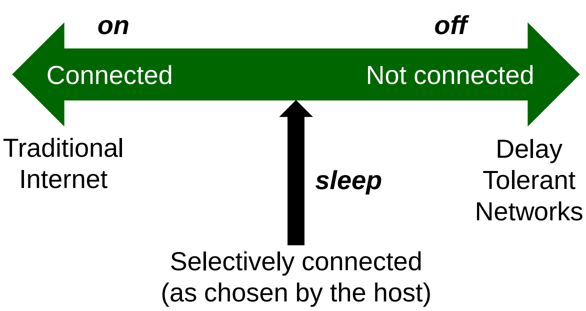
Abstract

We offer an initial exploration of the architectural constructs required to support *selective connectivity*, whereby a host can choose the degree to which it maintains a network presence, rather than today's binary notion of "connected" or "disconnected". The driver for our thinking is to allow hosts to go to sleep to realize energy savings while not sacrificing their entire standing in the network.

The main idea

Most hosts are inactive (idle) most of the time.

Need a state between connected and not connected.



This *selectively connected* state [1] can facilitate:

- Controlled preservation of network state for sustained operation across sleep and wakeup.
- Administrator access for maintenance.
- User remote access.
- Occasional low-level activity

Key example – Instant Messaging (IM)

Our idea helps powered-down hosts by re-connecting after a host returns from sleep.

Our goal: to enable reception of messages while a host is selectively connected.

- Perhaps even from a select set of "buddies"
- That is, while sleeping the host retains its standing in the network

Network chatter

What prevents hosts from going to sleep?

Hosts incessantly engaged in network dialogs.

- Our measurements: 4 to 6 pkts/sec for hosts with no direct user activity

How much of this is vestigial/extravagant and how much reflects a required network presence?

Key near-term effort is a detailed end-system trace analysis of network traffic from/to several hundred systems each recorded for 24 hours.

Proposed architectural elements

We can view end systems as having three basic states: *on*, *off*, and *asleep,* where asleep reflects a voluntary form of off for which end system can prepare and is willing to forgo or be awaken from if need be. To support this we explore:

Assistants – Entities that stand in for a host while the host sleeps.

Exposing selective connectivity – Providing visibility into a host's level of connectivity across different layers and peers.

Evolving soft state – Refining soft state to support persistence for selectively-connected end systems.

- *Proxyable state*: Soft state that an assistant or proxy can maintain [2].
- *Limbo state*: Allows sufficient information exchange that participating machines can recognize distinction between being "*inexplicably gone*" and simply "*asleep*".

Host-based control – Leave the end system in control of how others in the network react to the system's selective connectivity. We seek to support a range of user/operator-designated *policy decisions*.

Application primitives – As a way to empower assistants over a range of network activity, our design calls for the development of generic primitives that could be shared across applications.

Security – Issues relating to secure operation cut across all of the architectural components.

- What information is exposed externally?
- How do we layer cryptography to expose enough protocol machinery to assistants without exposing sensitive data?
- How can we securely delegate tasks to assistants?
- How can a peer understand that an assistant is authoritative?

Selected references

[1] M. Allman, K. Christensen, B. Nordman, and V. Paxson. Enabling an Energy-Efficient Future Internet. *ACM SIGCOMM HotNets*, November 2007.

[2] 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*, 15(5):297–310, September/October 2005.