Virtualizing Home Gateways to Reduce Energy Power Consumption

Jean-Patrick Gelas - Laurent Lefèvre
Teferi Assefa – Mulugeta Libsie
INRIA Grenoble / Université Claude Bernard- Lyon 1 – France
Addis Ababa University - Ethiopia

Collaborative work…

o Lyon, France

Addis Ababa, Ethiopia

Motivations

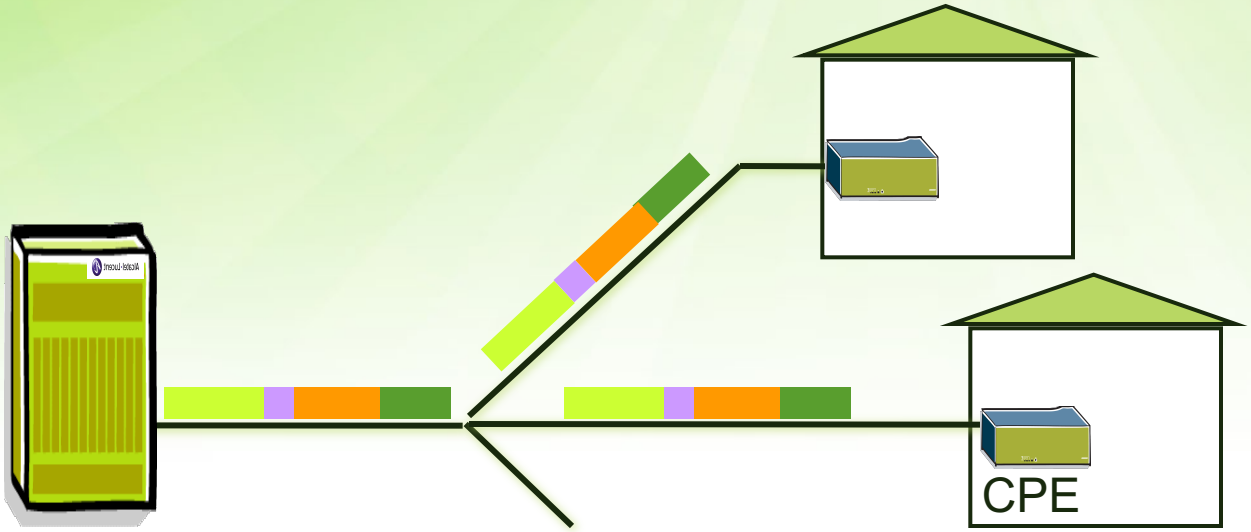
- GreenTouch initiative : Increase network efficiency by a factor of 1000 from current levels by 2015.
- Wireline networks : About 80% of the energy is consumed in the access network [1].
- Home gateway or Customer Premises Equipment (CPE):
  - Dissipate energy (11W-31W).
  - Provide services
    - Convert signal and protocols,
    - NAT, DHCP, …
    - Multimedia, Storage, Console game,…
  - Difficult to manage for network operators (ISP).

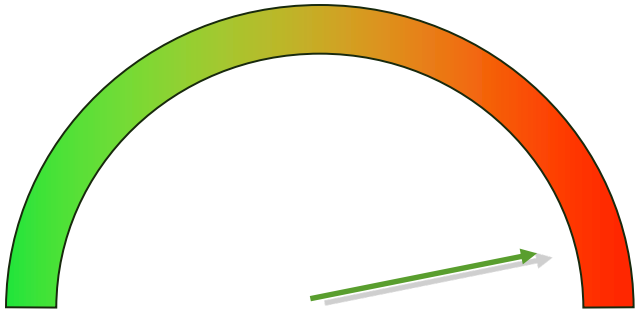
[1] Bolla R. et al. « The potential Impact of Green Technologies in Next Generation Wireline networks », Communications Magazine, IEEE, August 2011.

Objectives

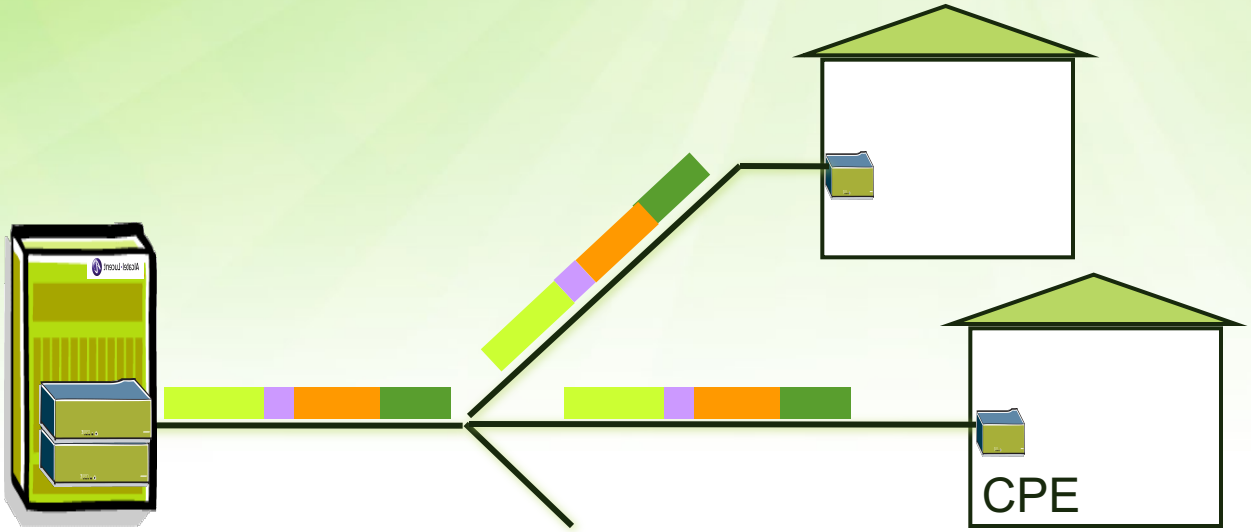
- Reduce Home Gateway (HG) power consumption
  - By virtualizing Home Gateways services towards some specific part of the network.
  - Run services on dedicated and shared facilities.
- Assumptions : Most end users will have triple-play services over a fiber link (FTTH). A fiber to Ethernet (or WiFi) converter is still required.
- Relocating network and application level services from HG into virtual Home Gateway (vHGW).
- If one server host around 1000 vHGWs we may achieve 300% energy saving in the overall wireline telecom access networks.

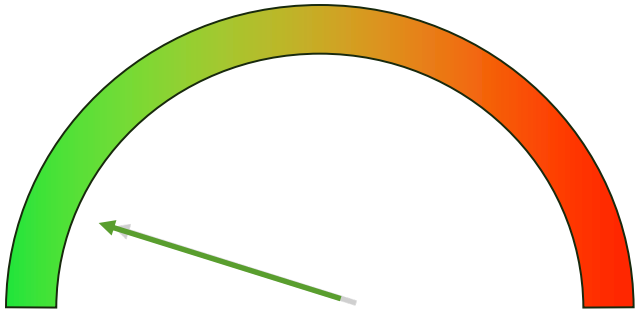
Standard Box replaced by low consumption box and services are virtualized





Standard Box replaced by low consumption box and services are virtualized

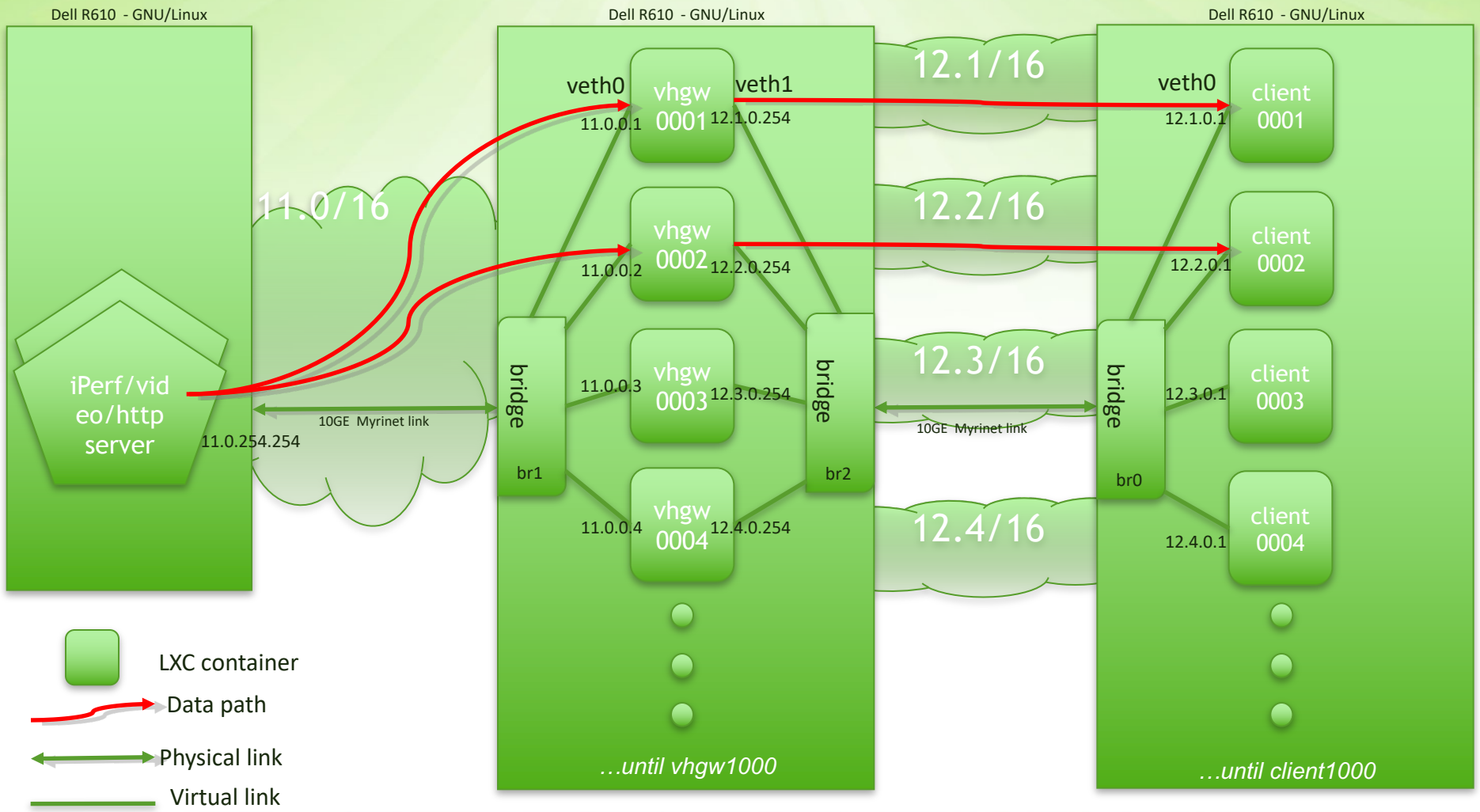




Experimental validation : Initial setup

- Three Dell R610 (Intel Xeon E5560)
- GNU/Linux Debian 6.0 (Squeeze)
- Consumption : 80W (idle)
- Networks (back-to-back): 10 Gbps Ethernet (Myrinet)
- Node role :
  1. Network traffic generator host (e.g video server)
  2. vHGWs host
  3. virtual clients host.

Experimental testbed (detailed)



GreenTouch™

Experimental validation : Software architecture

- Objective : Run as much as possible virtualized boxes !
- Extremely ligthweight virtualisation technology required (XEN, KVM, Vmware… does not fit!)
- Considering solutions based on simple isolation (vServer, LXC,…)

Introduction to LXC (Linux container)

- Lightweight virtualisation technology.
- Virtual ~~machine~~ environment with its own process and network space.
- Based on cgroups for restricted view of the host operating system.
- Smoothly integrated in Linux kernels.
- Provide great flexibility.
- Limited to Linux based guests.

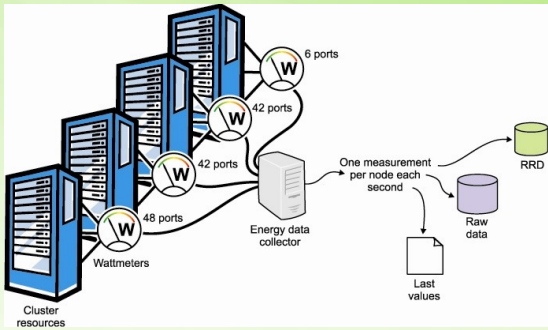
http://lxc.sourceforge.net

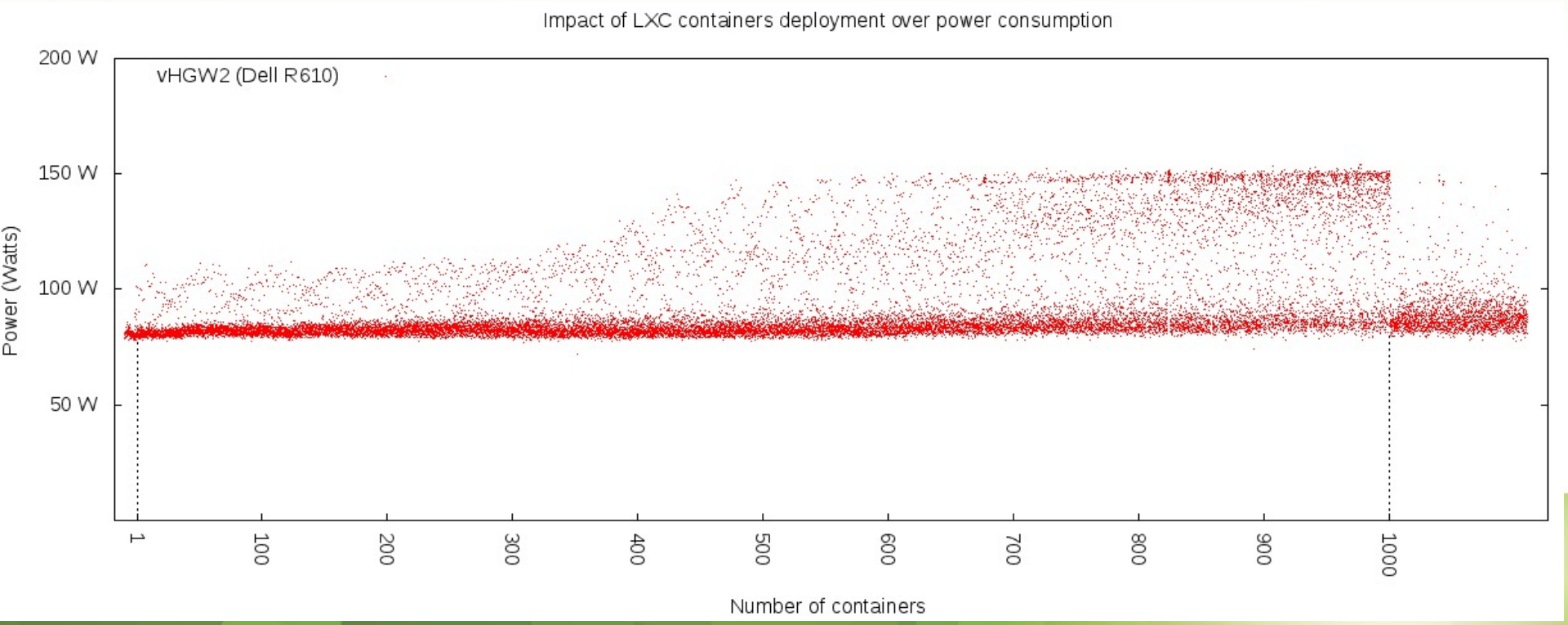
Experimental results

- The more we host vHGW, the more we save energy!
  - Run as much as possible LXC containers on one server (1000?)
- Problems :
  - Disk space : reduce file system size (213MB->140MB->20MB)
  - Launch all those containers (open file problem) (First try : 124 « only »)
  - Bad network performance (network neighbour table problem)
- Success! 1000 vHGWs are up and running on one node!

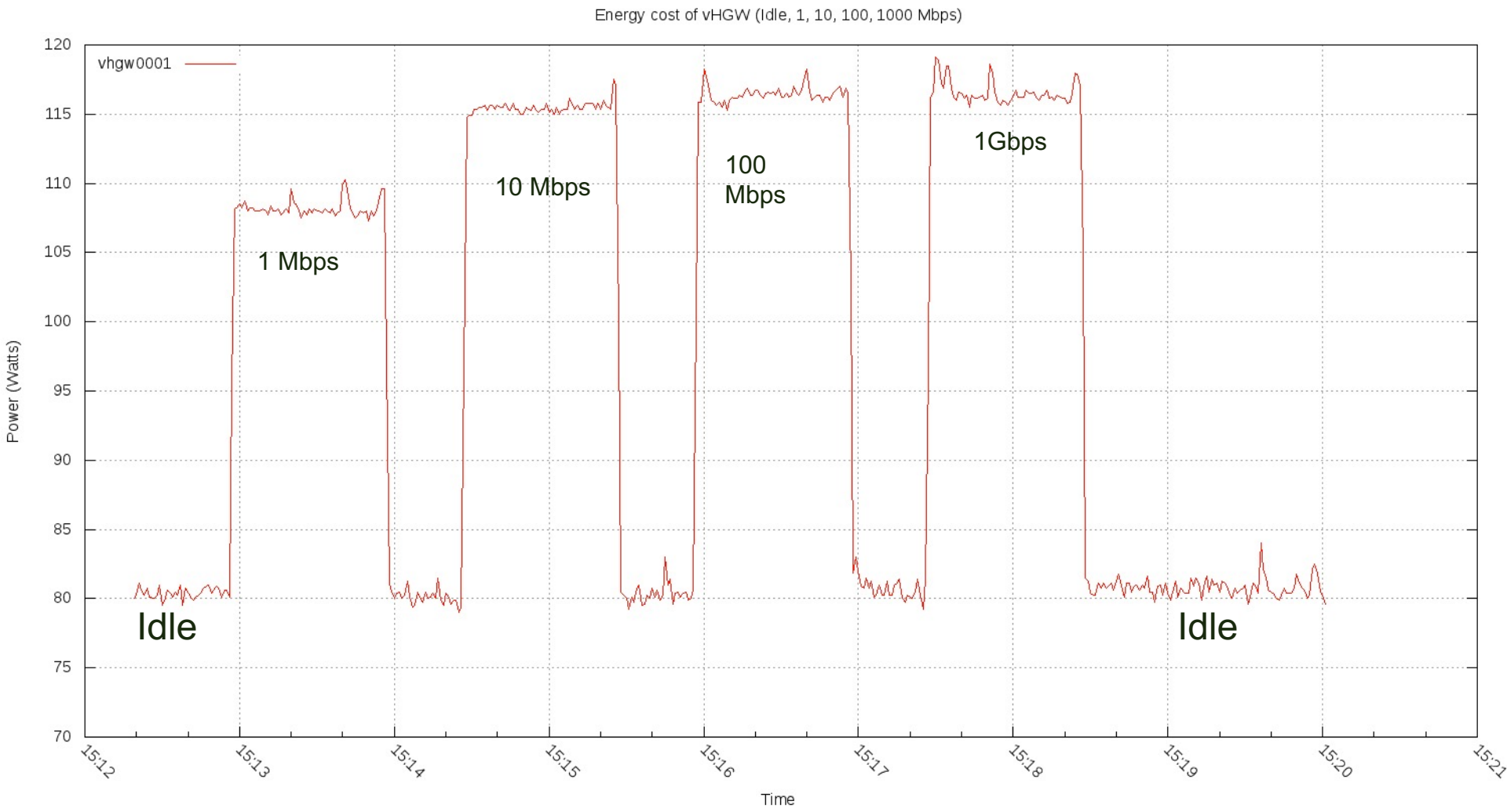
Power consumption of idle vHGWs

Scenario : Deploy one more vHGW (i.e. LXC container) each 20 seconds.
Experimentation duration : 6 hours.
Tool : External wattmeter (not ipmi)
Conclusion : If we ignore the clouds of point (due to the deployments phase) above the high density line, the power consumption is not really affected. However we can notice that once all the vHGW are deployed (then after 1000th), the average consumption is slightly increased.



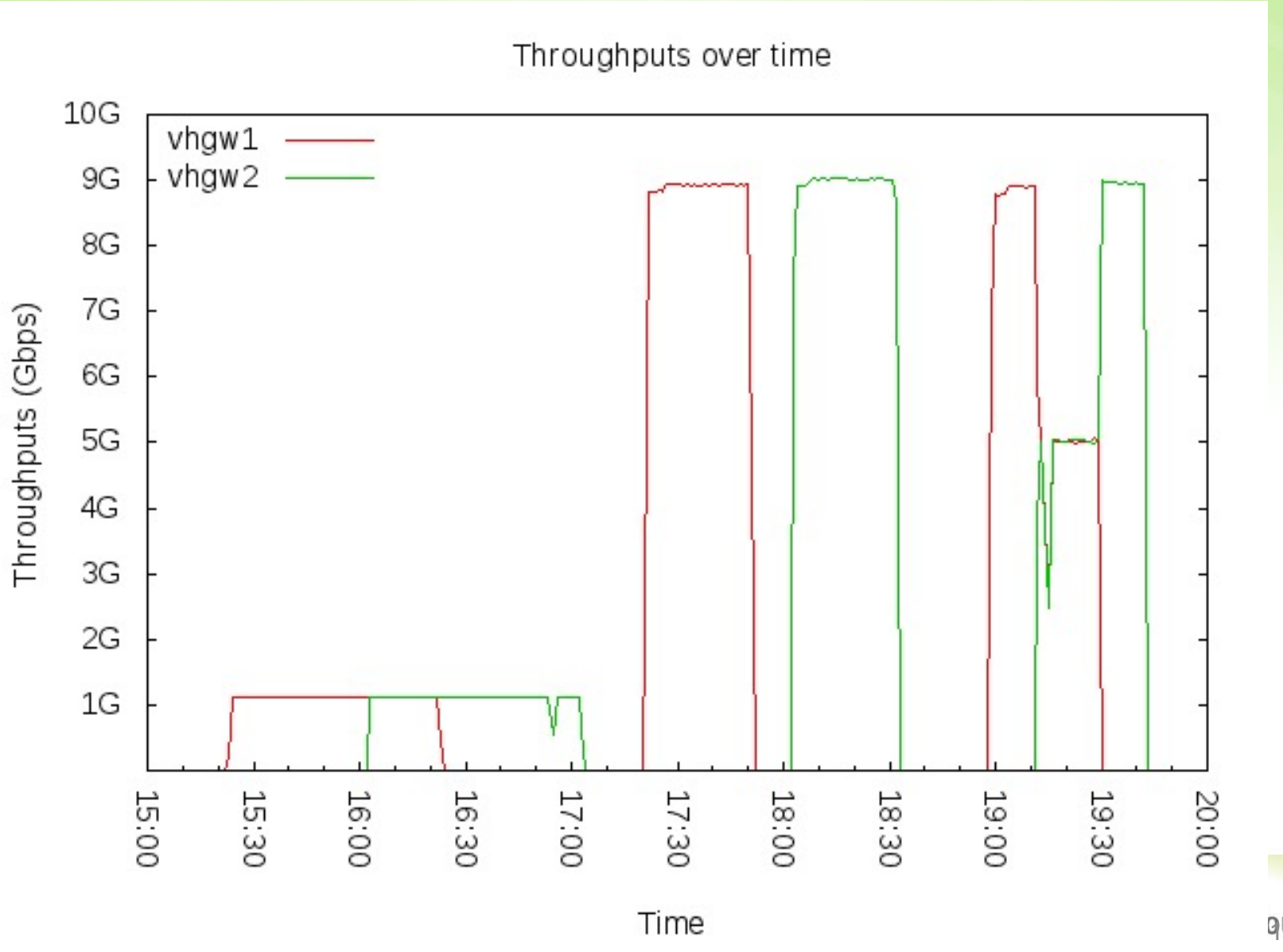


Energy cost of one vHGW forwarding data stream : 10W of difference only, then go fast to save energy!

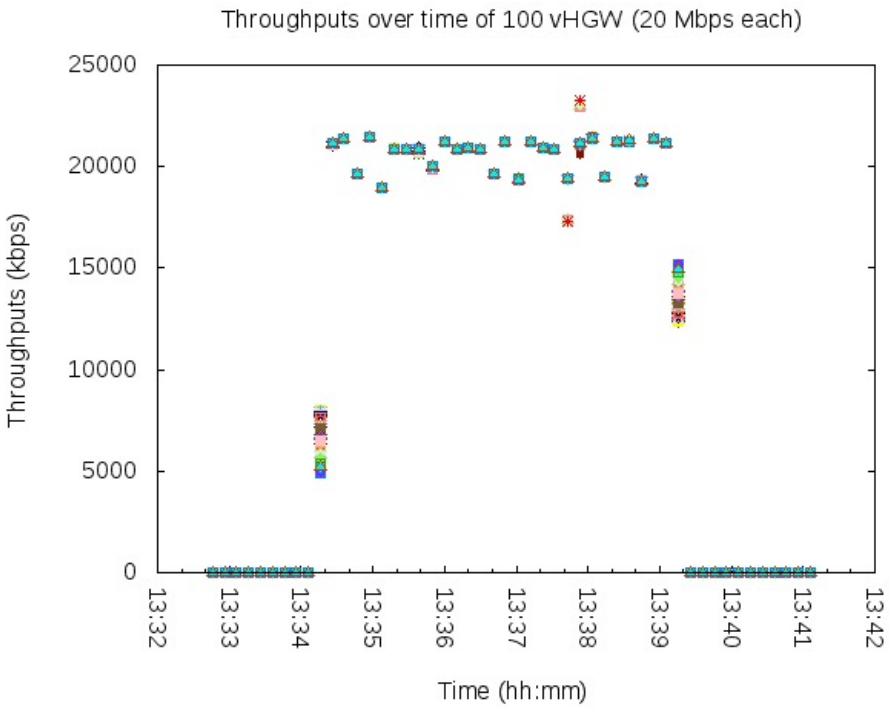


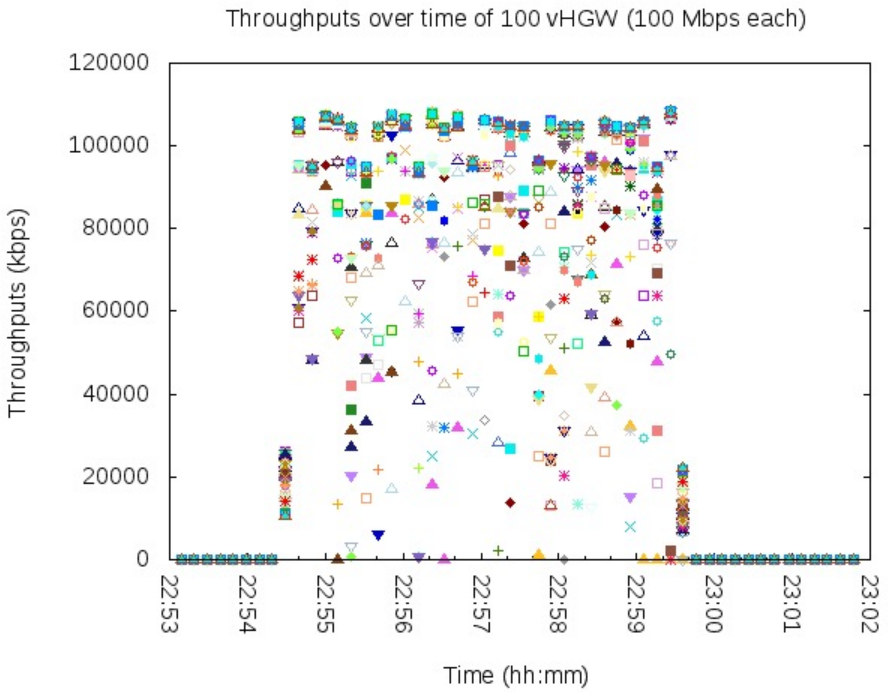
GreenTouch™

Network Bandwidth Sharing

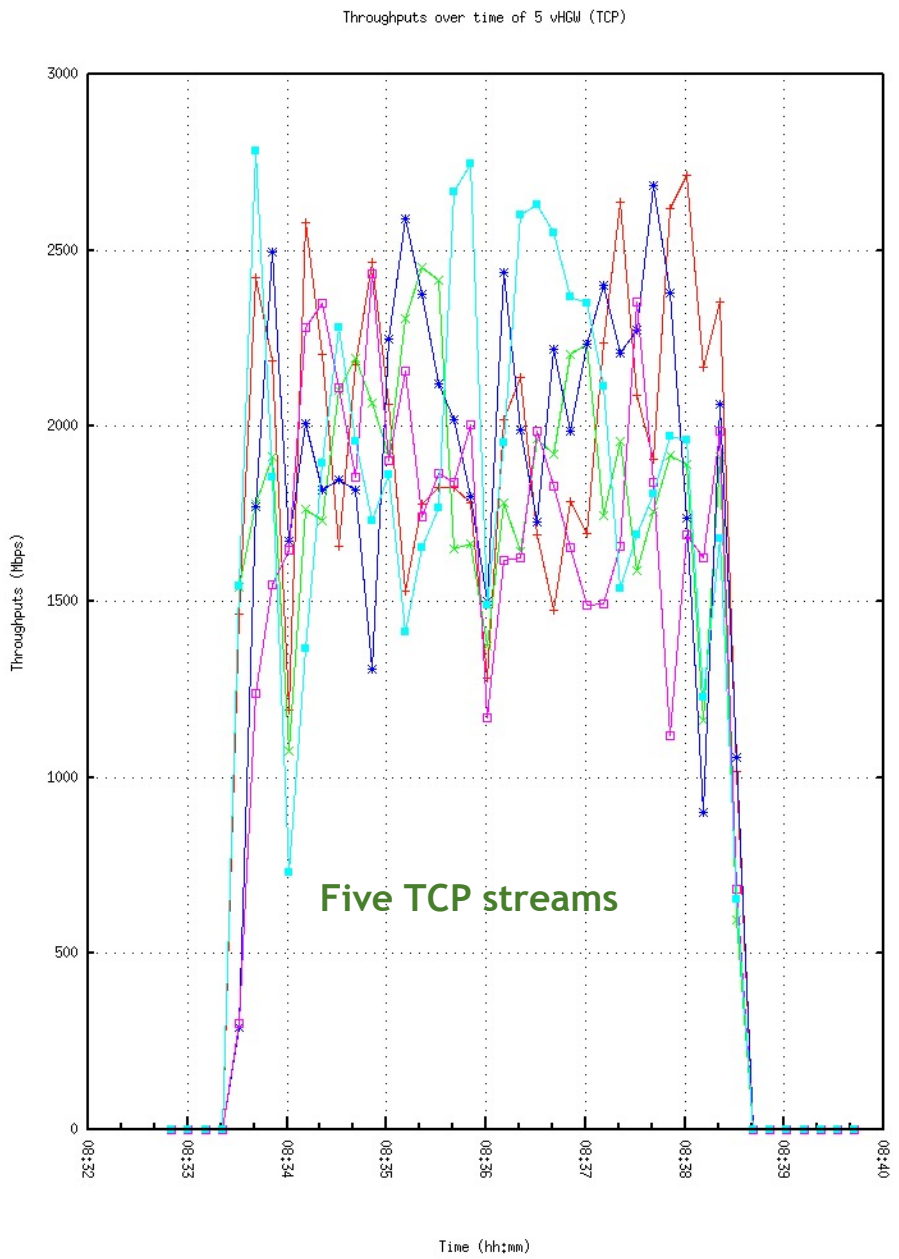


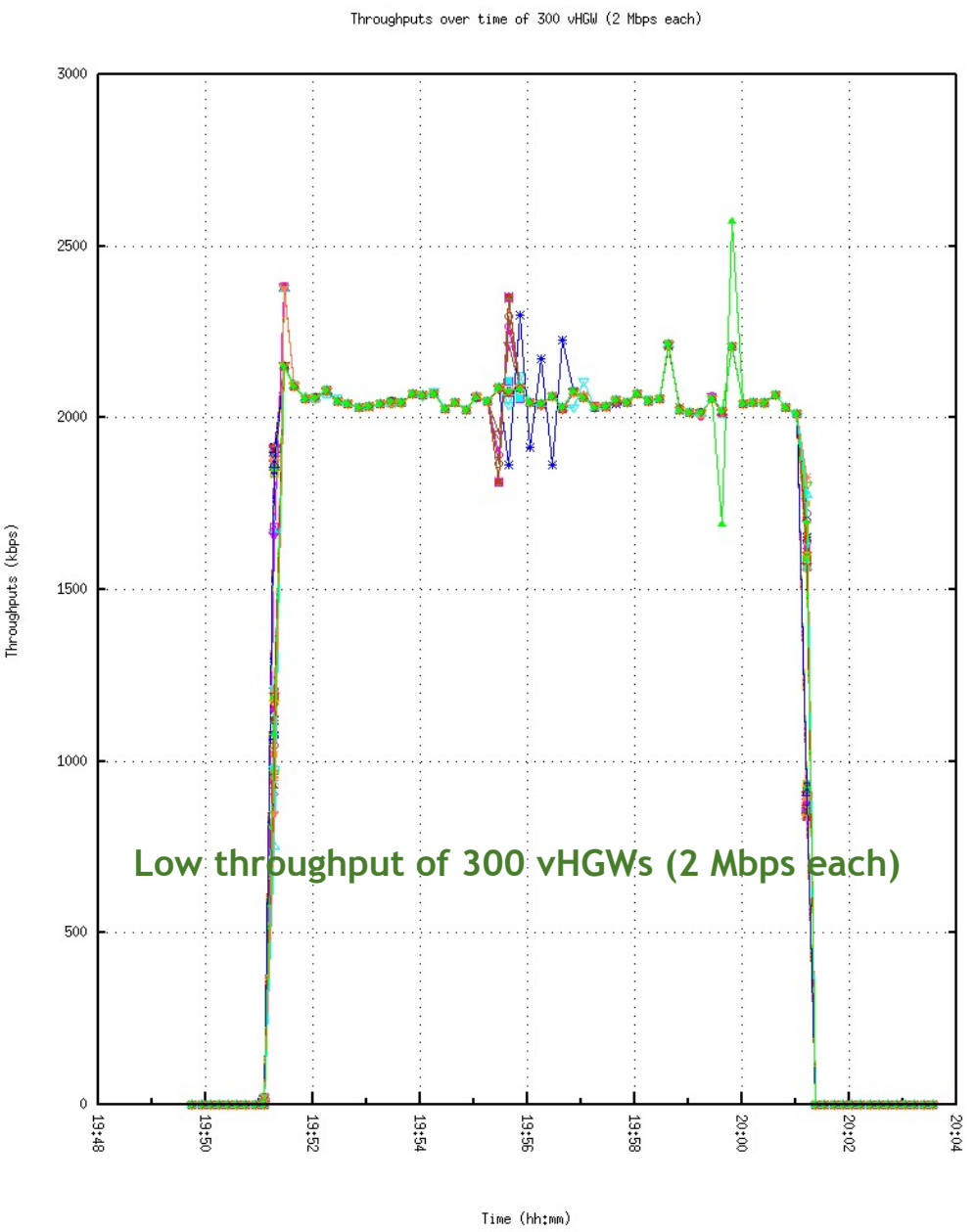
Bandwidth fairness between 100 vHGWs





GreenTouch





Bandwidth sharing is correct (but TCP by nature has difficulties to provide stability).

Here bandwidth sharing is good (but only 600 Mbps is used globally…)

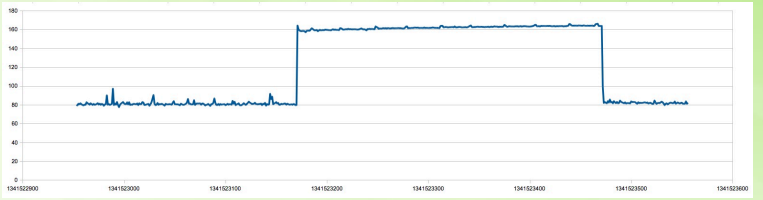
Latency and Jitter

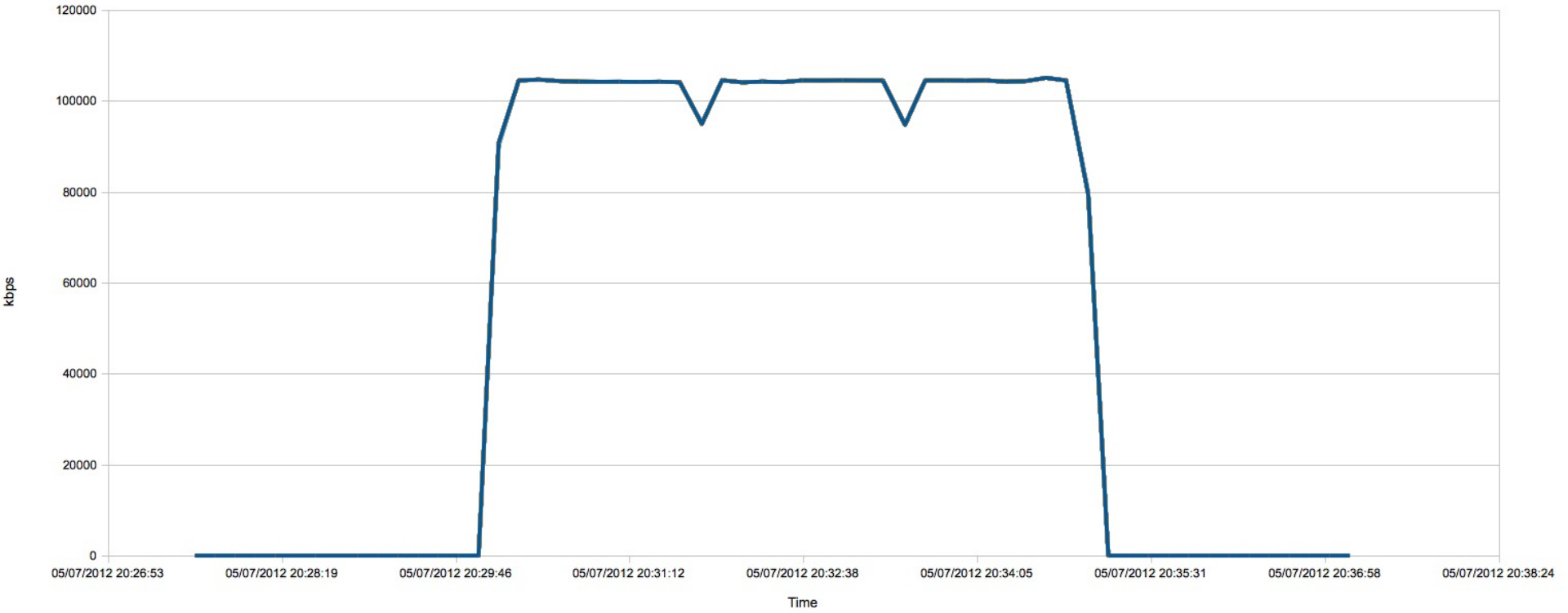
- Latency for crossing a vHGW on a server running 1 to 1000 vHGWs, and Jitter.
- Tools : Ping and iPerf
- Jitter remains stable (0.01 ms).

Number of running vHGW	Latency avg / mdev (ms)	Jitter (ms)
1	0.177 / 0.017	0.015
10	0.181 / 0.022	0.011
100	0.175 / 0.022	0.013
500	0.229 / 0.418	0.012
1000	2.330 / 5.845	0.013

25x100Mbps (163.94 Watts)

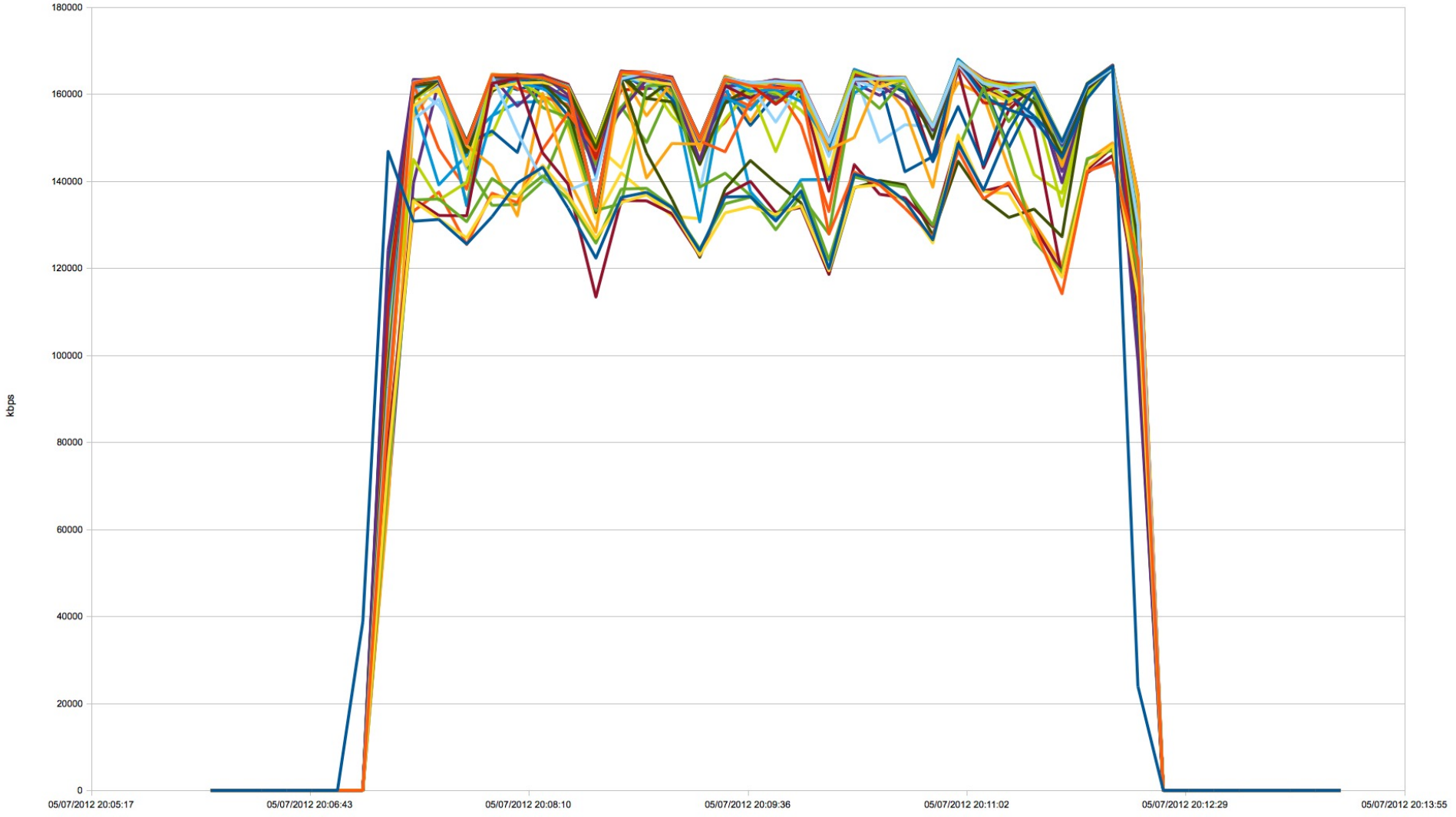
All the measures superimposed.
Fairness is perfect.



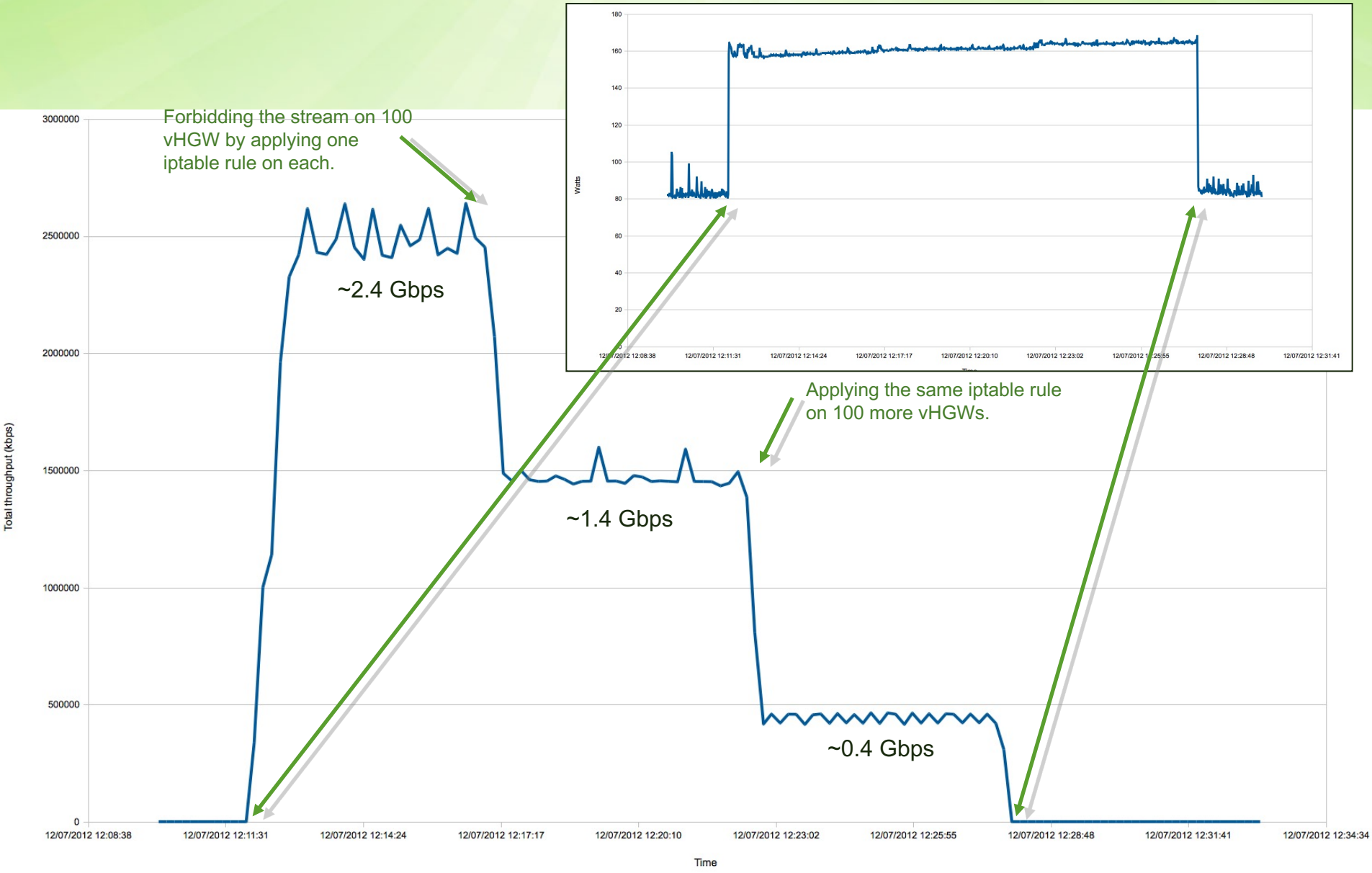


GreenTouch™

60x155Mbps (162.5 Watts)



Easy Management : Firewall rules scenario. 240, 140, 40 streams x 10Mbps



vHGW - Context Switch
(experimental results)

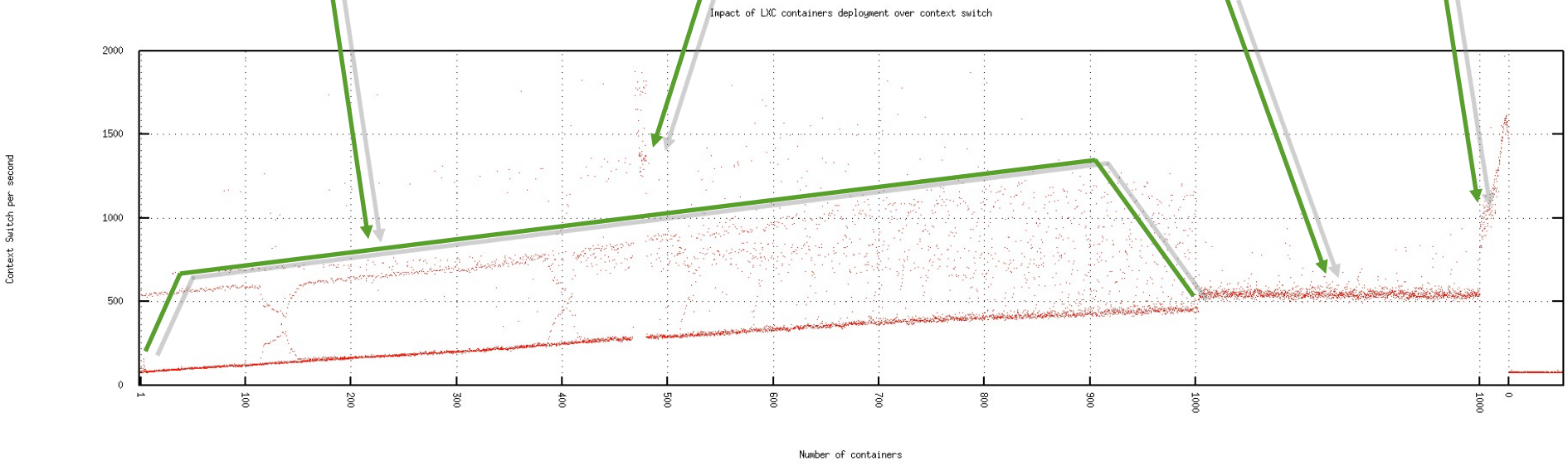
Scénario : Starting each minute a new vHGW.
(Time to start 1000 vHGWs ~18 hours)

The 1000 vHGWs are launched one by one…
While number of instances increase, number of context switch increase linearly.

Noisy area due to a *Logrotate* and an *updatedb* started at 6:30am.

Here all the 1000 vHGWs are launched and *Idle*. The number of context switch per second is about 500.

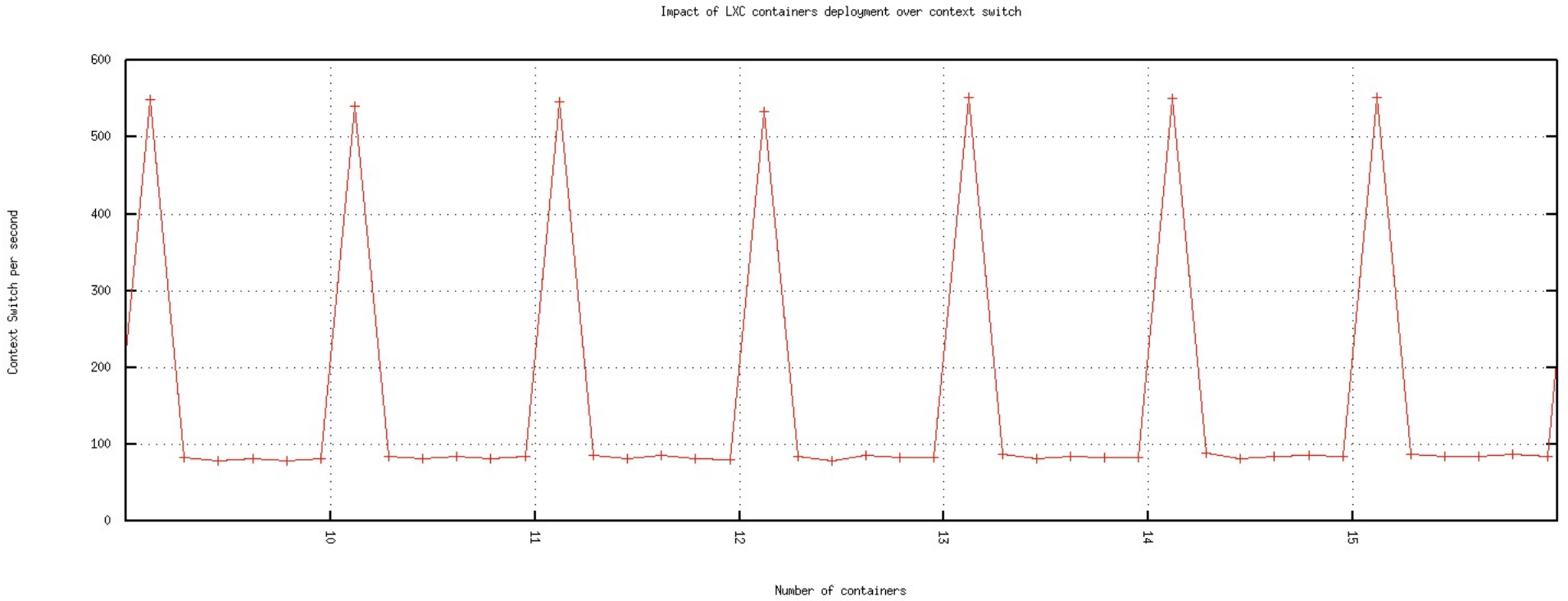
Here we stop all the vHGWs. It does generate an activity peak then fall back to ~80 context switch per second.



GreenTouch™

Zoom

Activity peak each time a new vHGW instance is launched.
(One peak every 60 seconds. One measuring point every 10 seconds)



GreenTouch™

Conclusion

- Considering in a near future that boxes will
  - Be reduced to the strict minimum.
  - Consume less.
  - Services will remain or become more complex.
  - Maintain QoS for the end users
- Proposed solution :
  - Virtualizing the Home Gateway
  - Aggregating complexity on dedicated servers.
- Gains in Energy and Management (and monitoring).
- Experimental validation
  (delivered with a set of experimental tools).
- Also show the limitations of this approach…

Thank you !

Jean-Patrick Gelas, Laurent Lefevre

Jpgelas, llefevre @ens-lyon.fr

http://www.greentouch.org

GreenTouch™

Gains and limitations…

- Gains :
  - Management
  - Power efficiency :
    - Box : 18W -> 6W
    - Server : 100W and 1000 vHGWs hosted.
    - 1000 x 18W = 18.000W
    - 1000 x 6W + 100 W = 6100W -> ~300% better!
- Limitations :
  - A system hosting a large number of vHGWs becomes busy even though vHGWs are idle.
  - A system forwarding a lot of streams is not fair (when aggregated throughput reach maximal throughput)
  - Single point of failure.
  - Security and privacy concern…