Cornell University
Laboratory for Elementary-Particle Physics

Assets server at LEPP

Inventory:OCSNG + GLPI

Monitoring: Zenoss 3

Inventory Management

- What are the systems?
  - Hardware
  - OS
  - BIOS
  - LEPP Pushed Packages (Windows)
- General Status of Computers
  - Did it check in today?
  - Where is it located?
  - Who purchased it
  - Who is the user or contact of record?

OCSNG

- Provides general “moment in time” Hardware and software inventory
- Can deploy software
- Limitations
  - Windows support is limited to 32 bit right now
  - Some issues with deployment on Windows 7
  - Note: new agent is planned for Windows 7 – in beta now
  - No manual data entry

GLPI

- Full lifecycle asset management
  - Tracks order
    - Supports product templates
    - Supplier suggested prices
    - Order placement
    - Delivery
    - Generation of item entries based on template and serial # once received
  - Tracks changes from OCSNG
    - Hardware changes
    - Specified license tracking
  - Supports manual entry
    - Location
    - User / contact of record
    - Notes
    - Status
      - Deployed
      - Storage
      - Computing Stock
      - Configurable

Monitoring

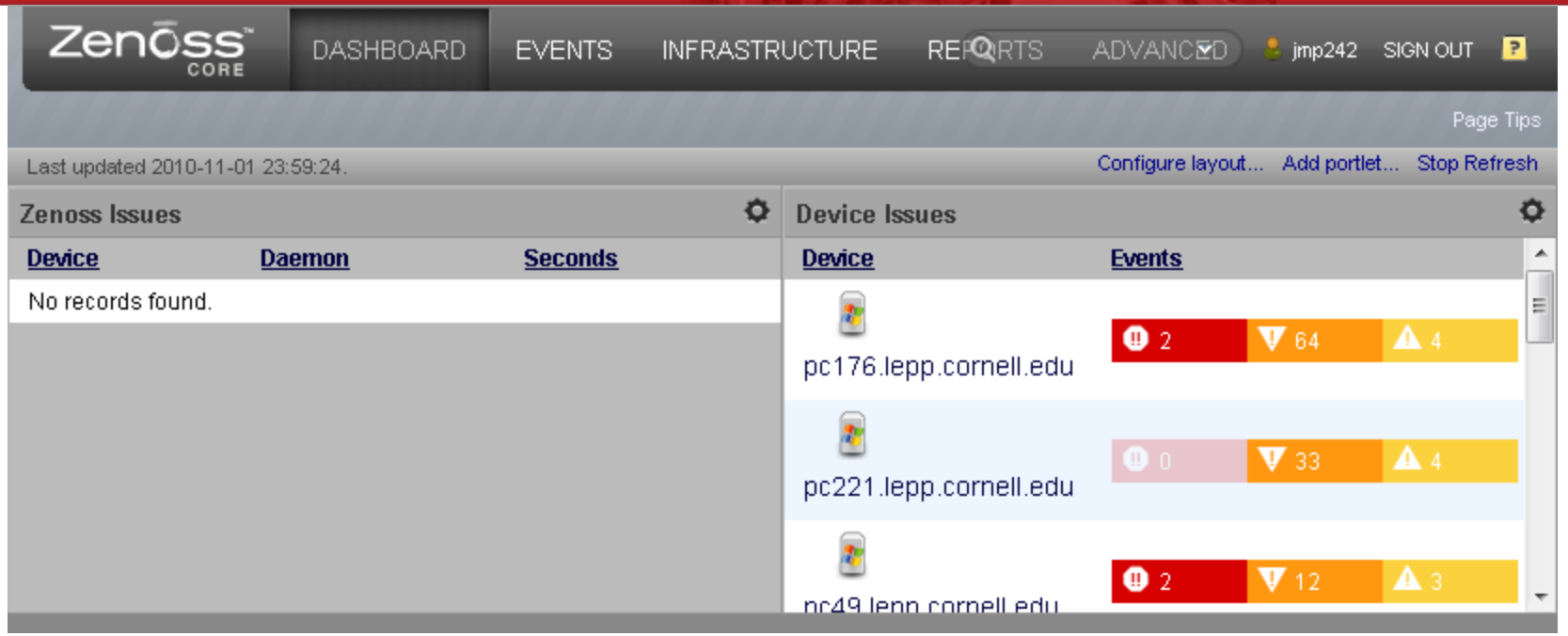
- Is the device pingable?
- Are specified services running?
- Graph performance metrics
  - CPU
  - Memory
  - HD I/O
  - Ping Response time
  - Custom information
    - Squid performance metrics
    - MySQL
- Custom monitoring
  - Is my script returning a 1?
  - Can I load index.htm from my web-server?

Zenoss

- Why Zenoss?
  - OSS
  - Company support
    - free in community forums and IRC
  - Active community
  - Agentless
  - Compatibility
    - Can talk many different “languages”
      - SNMP
      - WMI
      - SSH
      - Nagios Plugin
      - Cactai Plugin
      - Custom script / program
    - Targeted to run on RHEL5 and compatible – works great on SL5
    - Uses MySQL and ZOPE – two apps we already run for other services

Zenoss Limitations

- There are some limitations in Zenoss Core
  - No granular user access permissions
  - Major version upgrades (2.x to 3.x) are tricky
  - Currently much of the customization requires snippets (or more) of Python code.
  - Requires Apache reverse proxy to SSL the web interface
  - Agentless
    - If you've got a firewall between the Zenoss server and a host you want to monitor you'll probably need a VPN or to open ports



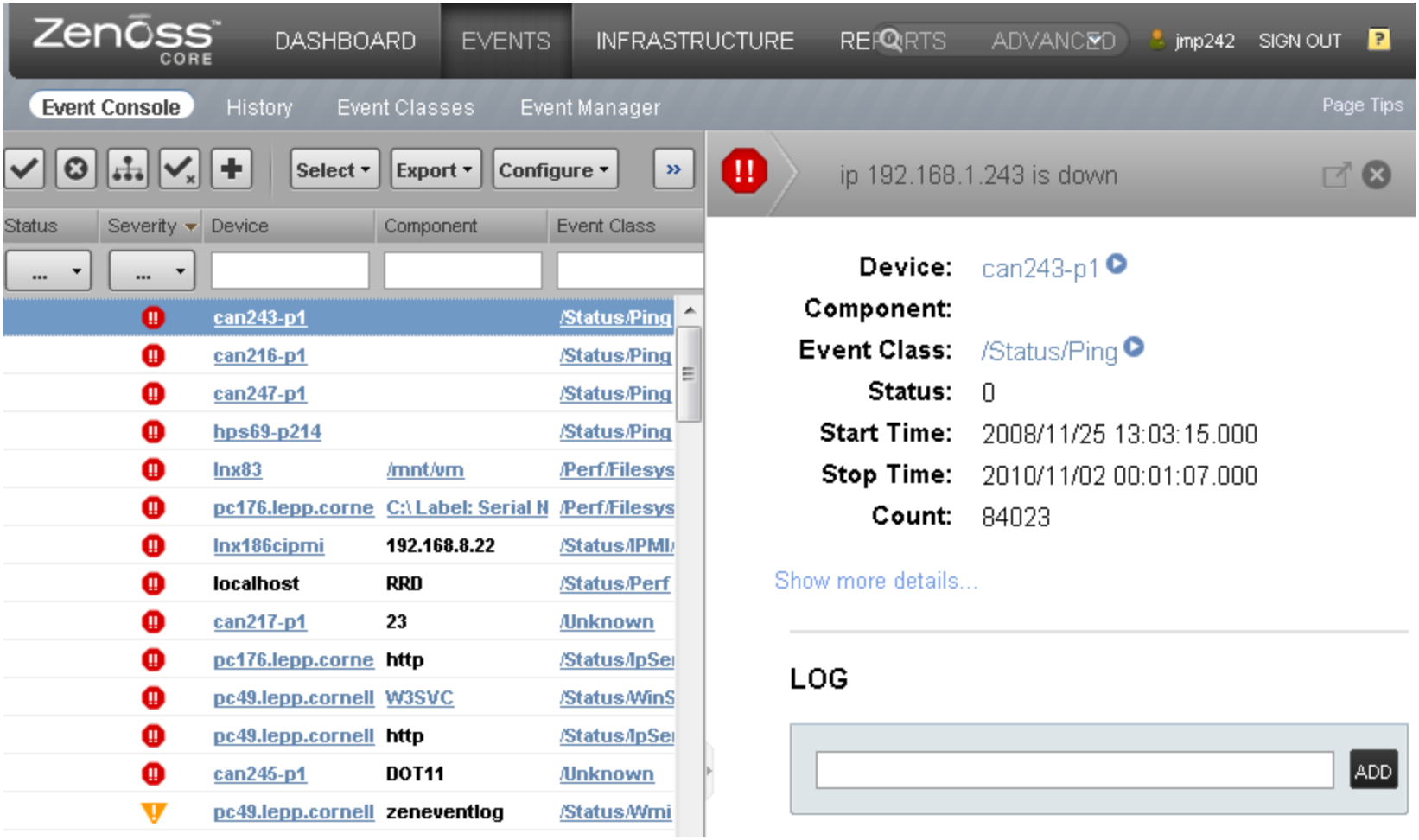
Zenoss Dashboard

Device issues gives a quick overview of events that are active

Also shows any internal Zenoss Issues

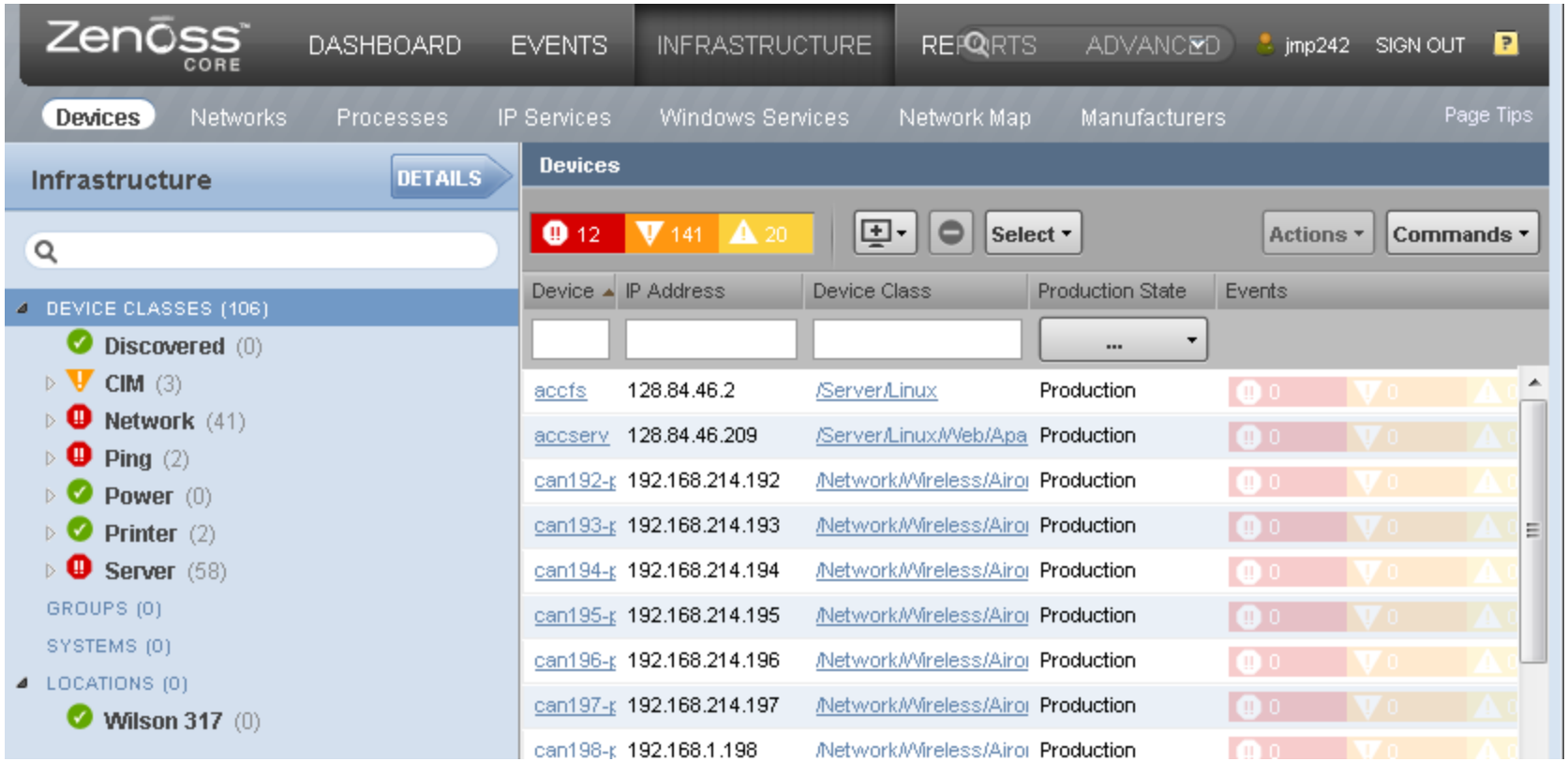
Portlets are configurable

Zenoss Events Display



- Shows all events, on the right is event details

Zenoss Infrastructure



- Devices is the major point of working with Zenoss for us
  - Device Classes are used to set monitoring templates
  - A Monitoring Template defines what is monitored and how it is monitored

How LEPP uses Zenoss

- The Web Interface is primarily for configuration and maintenance
  - You can use events right in the web view, but then you don't get quite as much filtering as easily
- Alerts are sent via e-mail
  - LEPP users usually don't have Zenoss up
- Zenoss can be configured to remediate selected events automatically
  - We have used this in the past to restart processes
  - We use this to pass off alerts to in-house systems via other methods than e-mail
  - You can get far more creative

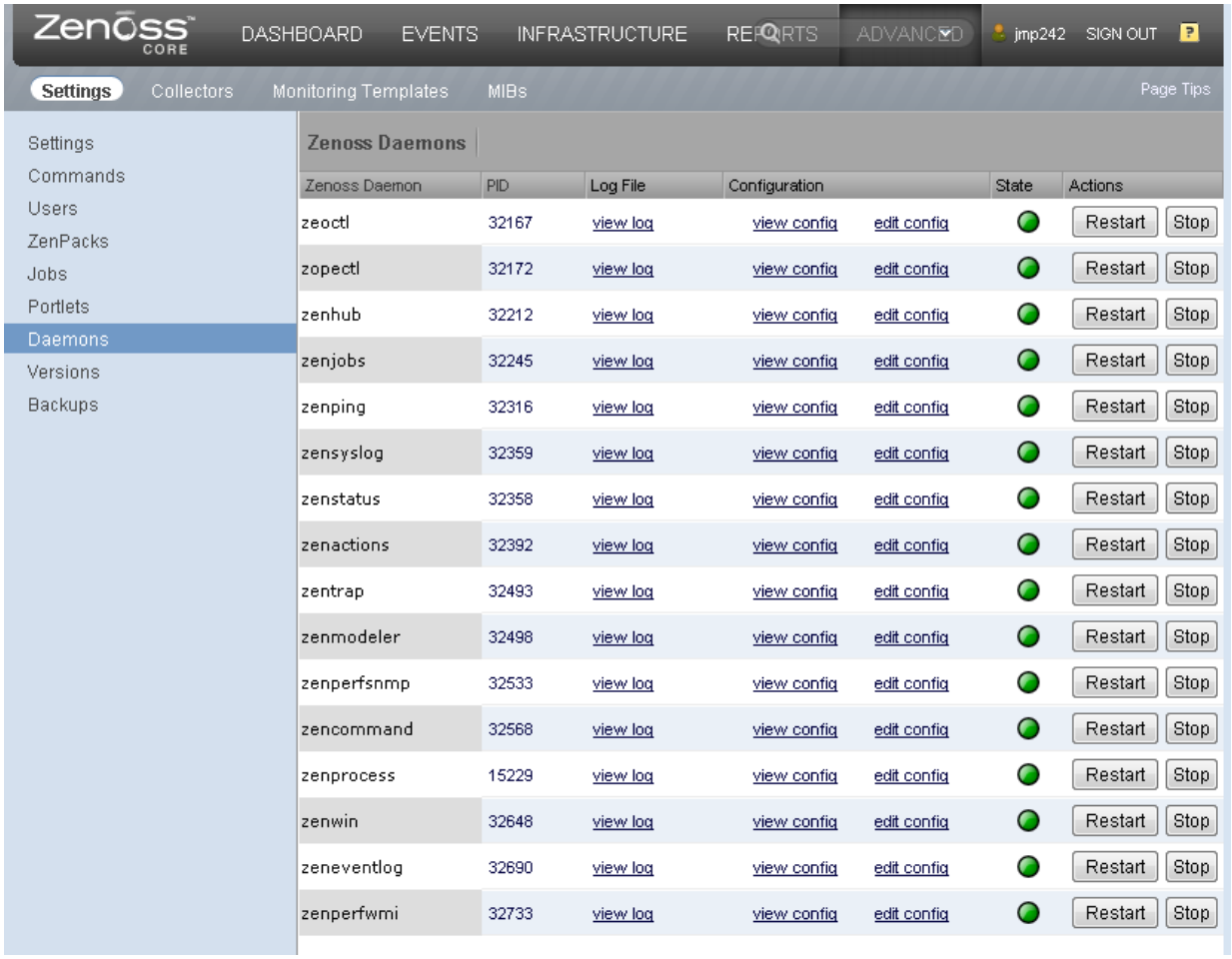
Example event lifecycle

- A monitored server goes down
- Zenoss runs a ping cycle every 60 seconds
- Zenoss notices the server is down
  - Marks as down in web UI
  - Generates an internal Event
    - May generate events for other monitored services on that server
- Zenoss polls new events every 60 seconds
  - Notices new critical server down event
  - Checks against user and group alerting rules (act like filters)
  - One matches this event
- Zenoss generates an e-mail
- User fixes problem, server back up
- Zenoss notices server is up
  - Generates a CLEAR event which closes out the existing critical event
  - Marks as up in web UI
- Zenoss e-mails user the event is CLEARed - resolved

Zenoss Scalibility

- Zenoss is quite scalable
  - Multiple Daemons – one for each step of the process
  - Central zenhub daemon can be configured to use multiple threads (n-1 where n = number of cores on Zenoss server)
  - Can support distributed collectors so you can spread out perf monitoring on multiple servers

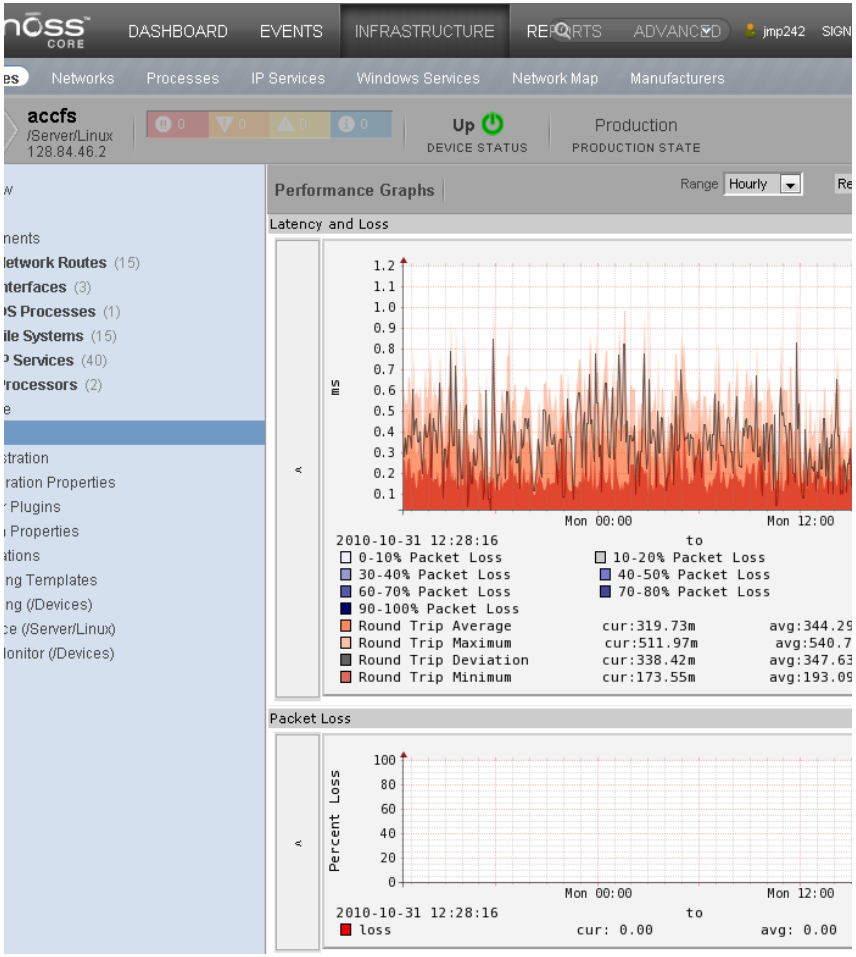
Zenoss Daemons

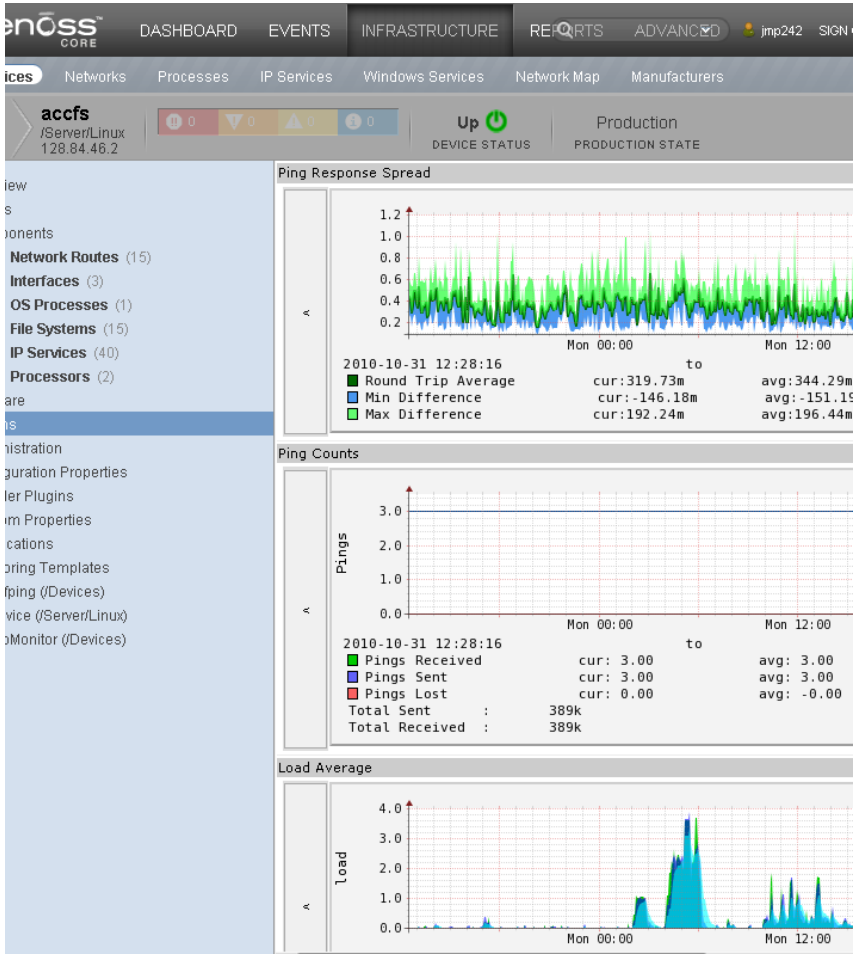


Extending Zenoss

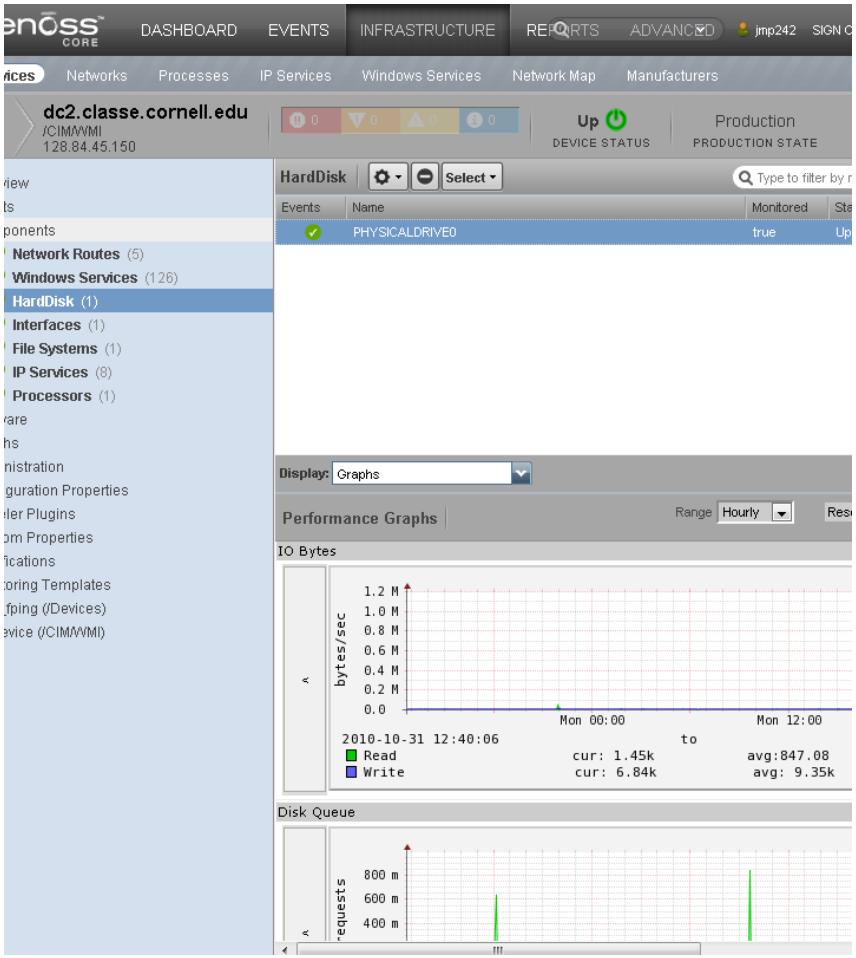
- Custom scripts
- Python code edits
- Writing your own daemon
- Zenpacks
  - Community
  - Zenoss, Inc
  - Internal

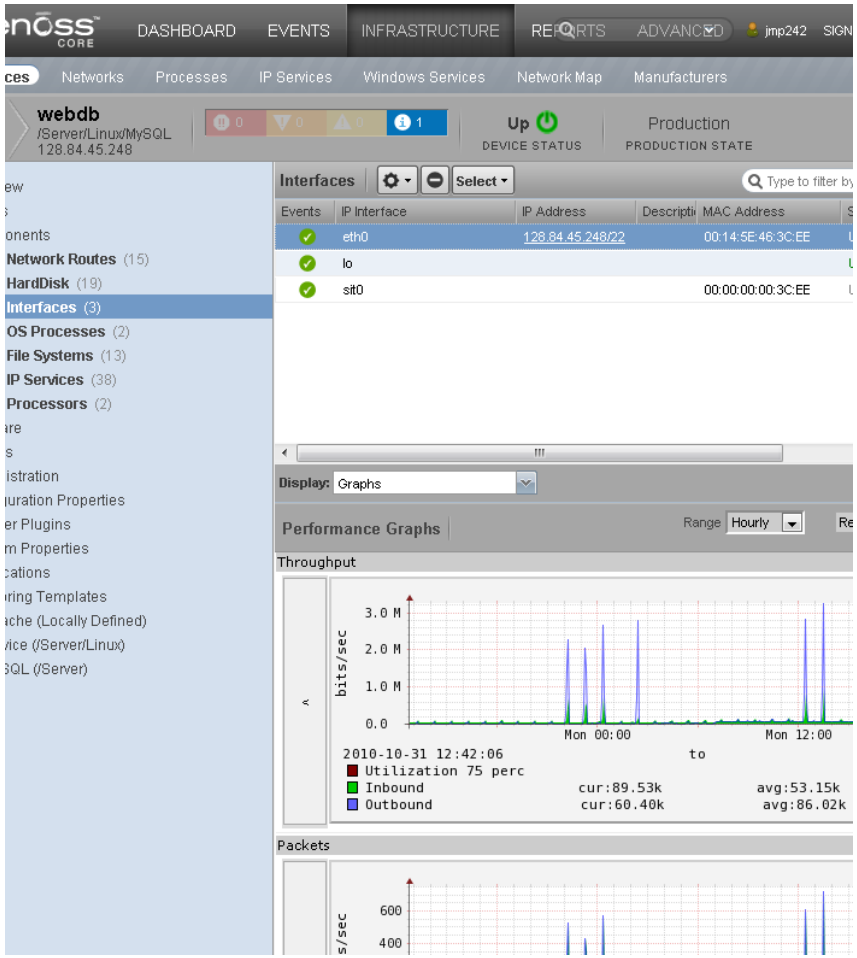
Screenshots





Screenshots





Links

- OCSNG
  - http://www.ocsinventory-ng.org/
- GLPI
  - http://www.glpi-project.org
  - English Forum:
    - http://www.glpi-project.org/forum/viewforum.php?id=6
- Zenoss
  - http://community.zenoss.org/index.jspa
  - http://community.zenoss.org/community/documentation
  - My community FAQs:
    - http://community.zenoss.org/docs/DOC-2445
    - http://community.zenoss.org/docs/DOC-4724