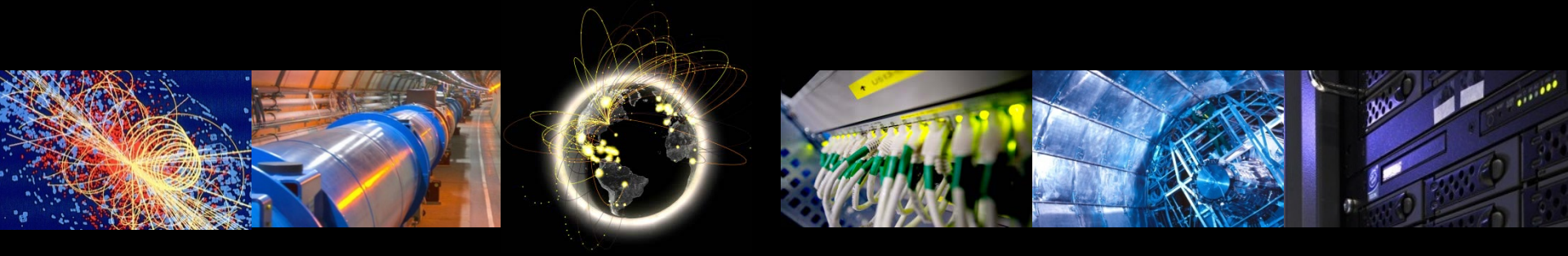
Containers: Security point of view

Vincent BRILLAULT, CERN/EGI-CSIRT

GDB April 2017, CERN



WLCG
Worldwide LHC Computing Grid

CERN

Containers: direct benefits

- De-couple provisioning and VOs:
  - OS/library independent* from VOs
    - No extended validation required
    - Less breakage from updates
  - No VOs libraries *leaking* to provisioning
    - No HEP_OSlibs package
- Better isolation than UID switch:
  - WN processes invisible/not accessible
  - WN files invisible/not accessible
  - cgroups for memory management

Containers: not a perfect solution

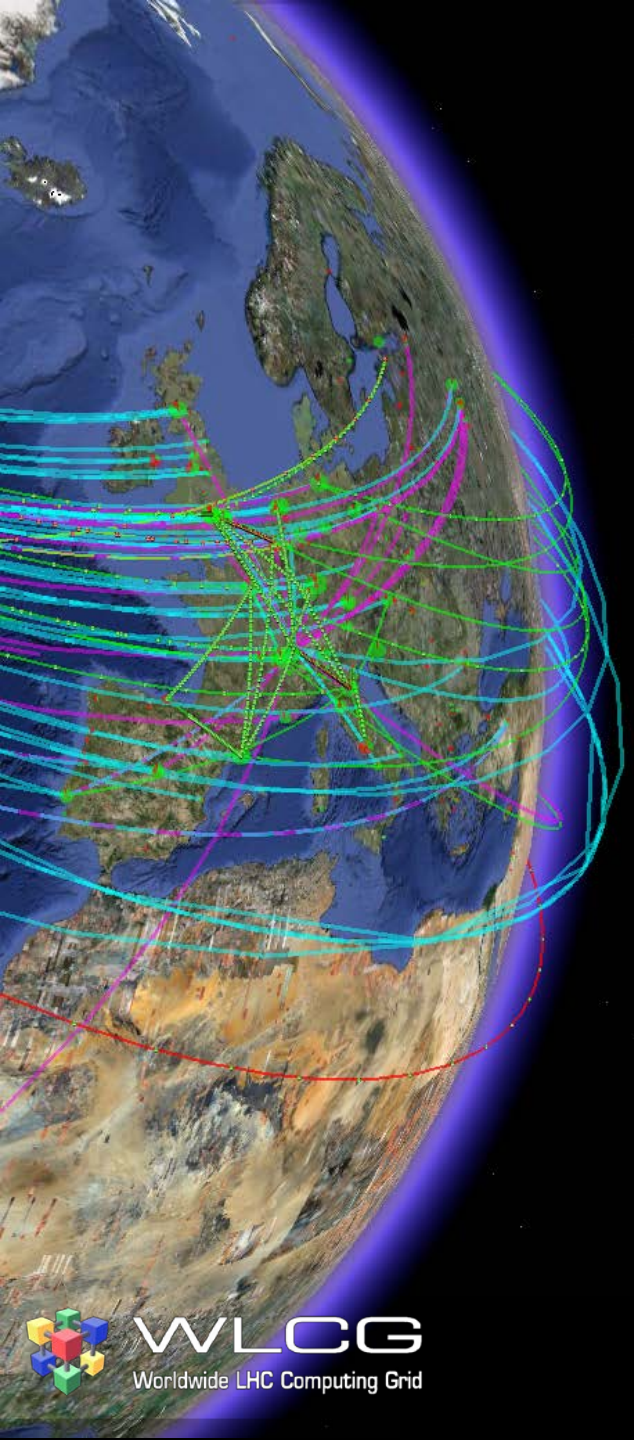
- *Young/recent* technology
  - New classes of bugs in kernel, missing support
  - Ecosystem changing fast (esp. docker)
- Most kernel bugs can still be exploited
  - Local privilege escalation kernel bugs still critical
  - Emergency updates still required…
- No migration possible (not like VMs)
  - Draining still required for reboots

Replacing GLExec using Singularity?

- Better isolation (container VS UID switch)
- Singularity SUID could disappear with RHEL 7.4
  - One sysctl configuration might be needed
  - Would rely on kernel security updates
- Simpler configuration (single RPM)
- No central callout/service required:
  - Simpler configuration & less failures
  - No traceability on end-user!

Retaining user traceability

- Re-build local traceability:
  - Pilot job could “tell” site which user is running
  - Feature present in HTCondor CE:
    - Missing audit logs: only current situation is available
- Use central VO services
  - If central service contain enough data & reliable
  - Incident response would need to be adapted
- Combine both?
  - Simple/small sites need less local features
  - Large sites can still react independently of VOs



Thanks for your attention!

Any questions ?

WLCG
Worldwide LHC Computing Grid