

# LXC

## and why system containers matter too

**Stéphane Graber <stgraber@ubuntu.com>**

Software engineer for Canonical Ltd.  
Upstream maintainer of LXC.

LinuxCon Europe 2014, Düsseldorf

# What are containers?

- Application containers
- System containers
- Embedded containers

# What about virtual machines?

- Performance
- Resources
- Manageability

# LXC 1.0

- First production ready release
- Supported (bugfix & security) for 5 years
- Stable C API (liblxc1) and bindings
  - ◆ Python3 (in-tree, full support)
  - ◆ Lua (in-tree, full support)
  - ◆ Python2 (out of tree, community support)
  - ◆ Go (out of tree, community support)
  - ◆ Ruby (out of tree, community support)
  - ◆ Haskell (out of tree, community support)

# Unprivileged containers

- Supported starting with 1.0
- Requires a recent distribution and kernel
- Allows running containers without any root privileges.
- Resolves most security concerns about giving container access to untrusted users.

# Container images and clones

- Containers images available for download
- Local cache for downloaded images.
- Get your own image workflow locally by customizing existing containers images and cloning new containers from them.

# Script your way around LXC

- All the LXC tools are only API clients
- You can easily write your own tools using just the bits of LXC you want and using your preferred language.

# Allow your guests to run containers

- LXC allows container nesting for both privileged containers and unprivileged containers (requires cgmanager).
- Resources can be restricted as usual through cgroups.



# Lets run some crazy things

- GUI applications in unprivileged containers
- Running your own Internet in containers
- Developer workflow using containers

**Stéphane Graber**

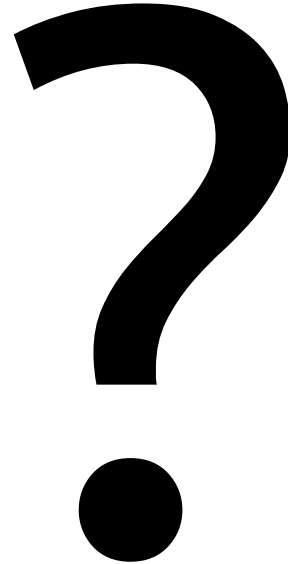
[stgraber@ubuntu.com](mailto:stgraber@ubuntu.com)

<https://www.stgraber.org>

**LXC**

<https://www.linuxcontainers.org>

<https://github.com/lxc>



**Questions anyone?**