InstantLab 2.0 - A Platform for Operating System Experiments on Public Cloud Infrastructure

Prof. Dr. Andreas Polze / Christian Neuhaus
with Rehab Alnemr, Lysann Kessler, Frank Schlegel

Cloud Futures, Berkeley, May 7th 2012
Many years experience in Windows-based OS classes

InstantLab 2.0 employs Windows Research Kernel (WRK)

- Stripped down Windows Server 2003 sources
  - Only kernel itself, no drivers, GUI, user-mode components
  - Missing components: HAL, power management, plug-and-play
- Released in 2006
- Freely available to academic institutions
- Encouraged by license:
  - Modification
  - Publication (of excerpts)
Structuring Experiments: The UMK Approach

- **U-phase**
  - Concentrate on OS concepts
  - Introduce OS interfaces
  - Systems programming

- **M-phase**
  - Observe concepts at run-time
  - Introduce monitoring tools
  - System measurements

- **K-phase**
  - Discuss kernel implementation
  - Introduce kernel source code (WRK/UNIX)
  - Kernel programming

---

The burden of running OS experiments

- Teaching operating systems requires chances for hands-on experience and demonstrations on live systems
- Providing these experiments is hard:
  - Changes of the underlying hardware and software make it hard to reproduce results
  - Considerable set-up work is required

Solution

InstantLab 2.0
InstantLab 2.0: Challenges & Solutions

• **Problem #1:**
  • Changes of hardware and software make it hard to reproduce experiment results (after more than 5 years of WRK in class)

  • **Solution:**
  • Run experiments in virtualized environments
  • Set of „canned“ experiments available via MSDN AA

---

• **Problem #2:**
  • Users (lecturers/students) need to set up and maintain experiment environment

  • **Solution:**
  • Experiment provisioning on cloud infrastructure
  • But: Which provider? Public/private?
Everything in the cloud, flexible choice of providers

- U/M experiments are easy
- K experiments are best run in private cloud
- User/Experiment management is big challenge
K experiment with two interconnected VMs – running kernel debugger and system under test.
InstantLab 2.0: Making it public

- So far: InstantLab accessible by invitation only
  - Our students (undergrad OS class)
  - A few connected schools

- The current trend: making teaching material available online

- InstantLab 2.0:
  Make experiment resources available to the public
  - Using public cloud infrastructure (easy)
  - On a self-service platform (tough)

InstantLab 2.0: Everything Self-Service

- InstantLab 1.0: Limited group of known users
- InstantLab 2.0:
  - Potentially thousands of users
  - Users we don’t know nothing about!
  - Problem:
    - How to manage and administer all these people?
    - How to decide who gets to use which resources

- Our Solution: A self managed version of InstantLab
  - Access control to resources based on trust relationships
  - Fully-automatic provisioning of experiments
How to distribute resources
(sponsors may be want to address certain target groups)

- Experiments on public cloud infrastructure consume resources and cost real money!
- Resources for a public teaching programme are limited
- Access control to experiment resources should:
  - ... foster earnest and competent users
  - ... limit misuse and wasting of resources.

Reputation and Trust

- How to rate a user as „beginner“, „advanced“ or „expert“?

- Behavior of user on the platform constitutes a user’s reputation:
  - Evaluation of completed experiments
  - Community interactions (e.g. contribution to support forums)
  - Online lessons and quizzes
  - Referrals and recommendations

- Reputation, interpreted by one’s own weighting and calculation scheme constitutes trust
Trust-based Access Control – the idea

Trust:
- is real-valued: e.g. \( t = 0.73 \)
- multi-dimensional:
  - correct computation, reliability, benevolence
- can be applied between machine entities and humans

Building and Maintaining Trust Levels

The trust value for every InstantLab users:
- ... is set to an initial value.
- ... is updated by transactions with this user.
Making Access Control Decisions

- Access to experiment resources requires a certain level of trust

<table>
<thead>
<tr>
<th>Experiment Repository</th>
<th>Kind of Experiment</th>
<th>Req. Trust</th>
</tr>
</thead>
<tbody>
<tr>
<td>user mode experiments</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>metering experiments</td>
<td>0.8</td>
<td></td>
</tr>
<tr>
<td>kernel experiments</td>
<td>0.95</td>
<td></td>
</tr>
</tbody>
</table>

\[ t \geq t_{\text{min}} \]

\( t \geq t_{\text{min}} \)

user mode: yes
metering: yes
kernel mode: no

Resource allocation – Trust revisited

- Reputation influences Community
- Community gives privileges
- Trust activates
- Public increases
- Private increases
- Public
- Private
Our private cloud – OpenNebula

- Private + Public Cloud
  - C12G Labs, Microsoft, CERN
  - 4CaaSt, BonFIRE, CERN, CESGA, D-Grid Resource Center Ruhr, Deltacloud, RESERVOR, SARA, StratusLab

Authentication

- OpenID
- OAuth
- Windows Live ID
Connecting with the Cloud

Future SOC Lab @ HPI – our private cloud

- Vision was to establish an open research platform for tomorrow’s IT landscape, start: June 2010
- Industry partners
  - Fujitsu
  - Hewlett-Packard
  - SAP
  - EMC
  - VMware
  - NetApp
  - Intel SCC

Application areas:
- Large Databases
- Consolidation, Virtualization
- High-Performance Computing

Testbed:
- MultiCore MultiThreading
- Hardware, huge memories, NehalemEX-based, GPU computing

- HP ProLiant DL980 G6: 64 Cores, 1-2TB
- Fujitsu Primergy RX600S: 32 Cores, 1TB

Steering committee from industry and academia
Conclusions – InstantLab 2.0

- Maintaining OS experiments for teaching is cumbersome
  - Virtualization may be the answer in provider side
  - Consumer still has to maintain experiment environment
- Putting experiments on the cloud lifts burden on consumer side

- Not all clouds are equal
  - In particular, none of the public clouds allows requesting co-located VMs or even worse – co-located physical machines
  - Need generic architecture for public/private clouds

- Everything has to be self-managed – otherwise it won’t scale
  - Notion of Trust and Reputation is crucial
  - Adopt techniques established for social networks