Ken Murphy
CS 495
Capstone Project Proposal
Project title:

Scheduled Hypervisor Allocation in a Cluster of Computers

Description:

This project aims to create a system where a user can spawn an arbitrary number of cluster nodes for the purpose of researching x86 virtualization. Each node will run on bare metal, giving the user full access over the machine. This will alleviate any issues that may arise from virtualizing a virtualization system and provide the user with a greater understanding of the system as a whole. Current systems that exist limit the user to spawning virtual machines, while this system will allow the user to spawn whole virtualization systems.

It is proposed that when a user spawns a cluster there is a time limit involved. This will allow multiple users to access the system in a time shared manner. When the time limit is reached all of their nodes will be archived 100% for further system spawns. This will allow a user to experiment without worry of losing all their progress once the time limit is reached. When a project (or group of nodes) are reserved, they will PXE boot and save all resources to the local disk. This will alleviate concerns of a single user causing a system slowdown for everyone. Once the time limit is expired the machine will gracefully shut down via IPMI and reboot with a backup utility that will also be loaded over PXE. This utility will create a differential backup of their system relative to either the base image or their last known state. This type of backup will reduce storage space requirements dramatically.

To begin with the system will have ten slave nodes and a single master node. The master node will be in charge of running all services such as DHCP, PXE, web, etc. The slave nodes will be the systems allocated for user spawns. For more information on the hardware setups see Paul Brennan's part of this project proposal.

In order to schedule allocations there will be a web interface available. This interface will be the front end to all system management scripts. A user's log-in will be checked against the CS department's
LDAP server in order to grant or deny access. If access is granted they will be able to see the available
timeslots and machines for their project. From there they can schedule when the machines should be up
and ready, how many they need, what virtualization environment is required, and for how long. A small
amount of project information may also be required for record keeping purposes. The user should also
be able to select other users who should have access to their project. This will allow groups of people
working on a single project to work together more efficiently.

The web interface will use a SQL based database for information storage. Backend scripts may
be used which read from the database in order to keep the web interface as 'dumb' as possible. This will
also make the front end interface feel fast and responsive which is a major requirement as just the 'feel'
of an interface can make it a pleasure or pain to work with. To start with the interface will be a
relatively simple and plain website. Over time the look of it may be improved, perhaps by a web design
oriented student.
## Project timeline:

<table>
<thead>
<tr>
<th>Date</th>
<th>Milestone</th>
</tr>
</thead>
<tbody>
<tr>
<td>September 13(^{th})</td>
<td>Hardware should be prepped and running. Should be testing various software solutions for services.</td>
</tr>
<tr>
<td>September 20(^{th})</td>
<td>PXE services should be in the testing phase, as well as the IPMI control system.</td>
</tr>
<tr>
<td>October 4(^{th})</td>
<td>A backup system should be implemented. This means that the machines should be able to pull down an image, run it, reboot, and get backed up. This cycle should be able to run again without losing any configuration changes made on each node.</td>
</tr>
<tr>
<td>October 18(^{th})</td>
<td>Backend scripting for node initialization, backups, and node management should be in the testing phase. The bulk of the functionality should be working, only minor tweaking should be necessary at this point.</td>
</tr>
<tr>
<td>November 1(^{st})</td>
<td>Web interface should be up and in the testing phase. All functionality should available and working. At this point it will go under a security assessment.</td>
</tr>
<tr>
<td>November 22nd</td>
<td>Images available to PXE boot should be undergoing development and testing. Once satisfied with the available images public system testing can begin.</td>
</tr>
</tbody>
</table>