VIRTUAL APPLIANCE DELIVERY OVER WIDE AREA NETWORKS
Yoshihisa Abe and Mahadev Satyanarayanan (Carnegie Mellon University)

BACKGROUND

- Virtual appliances
  - Pre-configured, ready-to-go virtual machine (VM) images
  - Free users from installing and configuring individual VMs, software etc.
- Contains entire software and configurations
- OS, applications, dependencies (libraries)
- All necessary elements for certain purpose packaged in VM
- Often used for cloud services
- Virtual appliances for end users have additional value
  - Try new software easily
  - No need to manage it on their own
  - Use software temporarily
  - Rent otherwise expensive software
  - Use software environment only when needed

APPROACH

- Take advantage of properties specific to virtual appliances
  - Fixed delivered images
  - Starting point of VM execution is well-known
  - Specific user workloads
  - Each virtual appliance is constructed for use of specific application(s)
- Use VM execution traces to estimate working set for each virtual appliance
  - Prioritize important parts of working set upon transfer so VM can start early
- Deal with missed VM state with prefetching by read-ahead

PRELIMINARY EXPERIMENTS

- Memory and disk footprint for varied workloads
  - Represents minimum VM state amount required

- Video playback performance
  - Initial wait time traded off for good FPS preservation
  - VM state transferred without compression

<table>
<thead>
<tr>
<th></th>
<th>Memory (unit: MB)</th>
<th>Disk (unit: MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accessed</td>
<td>Compressed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Separately)</td>
</tr>
<tr>
<td>Accessed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Video Playback</td>
<td>76.6</td>
<td>34.6</td>
</tr>
<tr>
<td>Video Editing</td>
<td>97.1</td>
<td>39.9</td>
</tr>
<tr>
<td>LibreOffice (Writer)</td>
<td>101.7</td>
<td>32.0</td>
</tr>
</tbody>
</table>

MOTIVATION

- Fast virtual appliance delivery to end users over wide-area networks (WANs)
  - Assumes provision of all VM state necessary for user tasks
  - No VM state cached on client or access through remote storage etc.
- Work in VM state transfer has mainly targeted good network environments (= LANs)
  - Except Internet Suspend/Resume (CMU), Collective (Stanford) etc.
- Key challenges
  - Deliver virtual appliances quickly
  - Minimize user wait
  - Preserve good VM performance
  - Minimize disruption of VM execution once started
- Efficient VM state transfer to clients
  - VM execution with partial state
    - Whole VM transfer, resulting in tens of GB, is usually prohibitive over WANs
    - Start VM execution as soon as necessary state is available

SYSTEM ARCHITECTURE

- Client implemented as modified qemu-kvm
- Memory and disk images passed through FUSE
- Lightweight TCP server handling state prefetch/fault-in requests

WORK IN PROGRESS

- Refining working set estimation
  - Balancing trade-off between initial wait and VM performance
  - More wait while transferring larger working set leads to better expected performance
- Using VM execution traces gathered through collection facility
  - Automated VM launch with instrumentation, on remote host
  - Target application is used in various ways while traces are collected
- Performance evaluation with virtual appliance images
  - System-level metrics: VM stall, state hit/miss rates etc.
  - Application-level metrics: FPS, timings of interactive actions etc.