Cloud Terminal: Secure Access to Sensitive Applications from Untrusted Systems

Lorenzo Martignoni,* Pongsin Poosankam,* † Matei Zaharia,* Jun Han, † Stephen McCamant,* Dawn Song,* Vern Paxson,* Adrian Perrig, † Scott Shenker,* and Ion Stoica*

*UC Berkeley and †CMU
Challenge

Goal: protect sensitive information in applications
- Confidentiality and integrity

Problem: client-size software stack
- Complexity → bugs
- User-administered → out of date, mis-configured
- Malware can be present at any level
Vision

- Sample application: online banking
- Quickly switch your PC to a secure operation mode
- Application provides a normal GUI
- But, information security does not depend on primary OS or its software
  - Even if commodity OS is compromised by malware
## Existing Approaches

<table>
<thead>
<tr>
<th>Property</th>
<th>Red / Green VMs</th>
<th>Per-app VMs</th>
<th>Browser OS (Chrome)</th>
<th>VDI &amp; Thin Clients</th>
<th>Flicker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installable w/existing OS</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Attestation</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Fine-grained isolation</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>No trust in host OS</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>User interface</td>
<td>any</td>
<td>any</td>
<td>browser</td>
<td>any</td>
<td>✗</td>
</tr>
<tr>
<td>Mgmt. effort</td>
<td>med.</td>
<td>high</td>
<td>low</td>
<td>low</td>
<td>low</td>
</tr>
<tr>
<td>TCB size (LOC)</td>
<td>&gt;1M</td>
<td>&gt;1M</td>
<td>&gt;1M</td>
<td>&gt;1M</td>
<td>250 + app logic</td>
</tr>
</tbody>
</table>
## Existing Approaches

<table>
<thead>
<tr>
<th>Property</th>
<th>Red / Green VMs</th>
<th>Per-app VMs</th>
<th>Browser OS (Chrome)</th>
<th>VDI &amp; Thin Clients</th>
<th>Flicker</th>
<th>Cloud Terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installable w/existing OS</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Attestation</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Fine-grained isolation</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>No trust in host OS</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>User interface</td>
<td>any</td>
<td>any</td>
<td>browser</td>
<td>any</td>
<td>✗</td>
<td>any</td>
</tr>
<tr>
<td>Mgmt. effort</td>
<td>med.</td>
<td>high</td>
<td>low</td>
<td>low</td>
<td>low</td>
<td>low</td>
</tr>
<tr>
<td>TCB size (LOC)</td>
<td>&gt;1M</td>
<td>&gt;1M</td>
<td>&gt;1M</td>
<td>&gt;1M</td>
<td>250 + app logic</td>
<td>22K</td>
</tr>
</tbody>
</table>
Secure Thin Terminal (STT)
Microvisor

- Minimal hypervisor, does not support multiple general VMs
- Uses hardware virtualization (Intel VT)
- Intercepts PS/2 keyboard and mouse
- Redirects frame buffer when Cloud Terminal is active
Startup and attestation

- Microvisor starts from a running untrusted OS
- Intel TXT with dynamic root of trust
  - Code derived from Flicker
  - CPU atomically hashes the microvisor, stores hash in TPM
  - Generate key pair kept by microvisor (but lost on reboot)
Cloud Terminal client

- Lightweight implementation of RFB (VNC) protocol
- When active, takes complete control of mouse, keyboard, and display
- Transport security based on SSL
- Reverse password to demonstrate authenticity
Untrusted user-space helper

- Runs as an unprivileged process inside commodity OS
  - Active when the Cloud Terminal is
  - Communicates with microvisor via hypercalls
- Relays encrypted data
  - Across network to CRE
  - To disk for persistence
- Cannot access or modify plaintext data
STT installation

**Case 1: pre-installed**
- Corporate-provided laptop
- Out-of-the box consumer device

**Case 2: install on existing machine**
- Verification service performs remote attestation
- User confirms a random nonce via an out-of-band (telephone) channel
Cloud Rendering Engine (CRE)
CRE approach

- In provider-administered data center
- Each user application runs in a VM with a standard VNC server
- Dispatcher relays connections to application VMs
- VMs run standard (e.g. Linux) applications
  - In a stripped-down environment
CRE scalability

- Share identical memory pages copy-on-write (up to 61% savings)
- Share base disk image
- Remove software not needed for desired application
- Disable periodic timer interrupts
CRE security

- Each VM has its own virtual network and firewall white-list
- Resource usage is capped
- Limited user environment
  - i.e., kiosk mode
- VM has no more authority than its user
Session walk-through

(1) secure attention key
(2) reverse password
(3) select application
(4) transport security setup
(5) attestation using TPM
(6) start remote UI session
(7) user auth. within remote UI
(8) access application

User

Secure Storage
reverse password
master key
app keys

STT

CRE
User experience
Evaluation: client TCB

<table>
<thead>
<tr>
<th>Component</th>
<th>Lines of Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microvisor</td>
<td>7.7K</td>
</tr>
<tr>
<td>Terminal client</td>
<td>3.0K</td>
</tr>
<tr>
<td>Crypto (PolarSSL)</td>
<td>5.5K</td>
</tr>
<tr>
<td>Attestation (Flicker)</td>
<td>5.7K</td>
</tr>
<tr>
<td>Total</td>
<td>21.9K</td>
</tr>
</tbody>
</table>
Evaluation: applications

- Document editing: AbiWord
  - MS Word `.doc` compatible
- Document viewing: Evince (PDF)
- Online banking: Firefox + Wells Fargo
- Secure email: Firefox + Gmail
### Evaluation: performance

16 core, 64GB server, 670 mi from client

Simultaneous clients replay recorded usage

<table>
<thead>
<tr>
<th>App. Activity</th>
<th>Baseline (ms)</th>
<th>STT (ms) with # of clients =</th>
<th>Network Usage (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>150</td>
<td>200</td>
</tr>
<tr>
<td><strong>Edit Launch</strong></td>
<td>2,844</td>
<td>2,208</td>
<td>2,441</td>
</tr>
<tr>
<td>Type a key</td>
<td>30</td>
<td>53</td>
<td>50</td>
</tr>
<tr>
<td>Move mouse</td>
<td>32</td>
<td>49</td>
<td>59</td>
</tr>
<tr>
<td><strong>PDF Launch</strong></td>
<td>1,699</td>
<td>2,093</td>
<td>2,147</td>
</tr>
<tr>
<td>Scroll</td>
<td>114</td>
<td>1,270</td>
<td>1,380</td>
</tr>
<tr>
<td><strong>Bank Launch</strong></td>
<td>6,911</td>
<td>2,319</td>
<td>2,563</td>
</tr>
<tr>
<td>New page</td>
<td>1,183</td>
<td>2,610</td>
<td>2,661</td>
</tr>
<tr>
<td><strong>Gmail Launch</strong></td>
<td>6,936</td>
<td>2,254</td>
<td></td>
</tr>
<tr>
<td>Display msg.</td>
<td>992</td>
<td>2,254</td>
<td></td>
</tr>
</tbody>
</table>
Qualitative usability

Display is 800x600, 8 bit color
- Suitable for a single application
- Could be improved with compression

Typing latency feels usable
- Similar to SSH

Scrolling feels sluggish
- Add optimization of block moves
Cost analysis

- A suitable server costs $1010/month
- Between 1.2 and 2.5 cents per user-hour
- Online banking: 5 cents per user per month
- Corporate application: $3 per employee per month (8 hours per day)
Summary

- Cloud Terminal: new architecture for secure remote applications
- Achieves sweet spot between security, trusted code size, and generality
  - Near minimal client size for remote interaction
- Runs inexpensively using standard hardware

http://bitblaze.cs.berkeley.edu/