Security Challenges at the Foundation

Breakout Panel 4:
Moderator: Sean Smith
Scribe: Warren A. Hunt, Jr
August 31, 2004
# Topics and Participants

<table>
<thead>
<tr>
<th>Topics</th>
<th>Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem-space meta-issues</td>
<td>Richard Brooks</td>
</tr>
<tr>
<td>Problems</td>
<td>Kate Cherry</td>
</tr>
<tr>
<td>Quad charts?</td>
<td>Conan Dailey</td>
</tr>
<tr>
<td>DARPA-hard problems</td>
<td>George Dinolt</td>
</tr>
<tr>
<td></td>
<td>Ed Greenwood</td>
</tr>
<tr>
<td></td>
<td>Warren A. Hunt, Jr.</td>
</tr>
<tr>
<td></td>
<td>Dwight Cass</td>
</tr>
<tr>
<td></td>
<td>Sean Smith</td>
</tr>
<tr>
<td></td>
<td>John Vonderheide</td>
</tr>
<tr>
<td></td>
<td>Michael</td>
</tr>
<tr>
<td></td>
<td>Nic Watson</td>
</tr>
</tbody>
</table>
Problem Space

• Why is HW different?
  – Tool chain
  – Lifetime
  – Reliable? (due to economics?)
• What type of hardware?
  – CPU? Or FPGA
  – Routers?
  – Graphics, Keyboard, Storage
• Whose space?
  – DoD/"intel"
  – COTS/Intel
• Evolution or revolution?
  – Don’t look at client-side SSL problem
  – Look at it---and try to find why it failed

• What do we look at?
  – Solutions: what do we understand now, and can we do more of it?
  – Problems: what is it that’s so hard now?
• Relation to what industry cares about?
  – Infusion
  – Economic drivers
  – What effect can we have on industry?
Problems Discussed

1. Side-channel/covert channel
2. DRM: can we build upon what we know industry’s interested in?
3. Virtualization
4. Tagged architectures/persistent labels
5. Encrypted functions/multiparty/ORAM/coercion-resistant
6. Hw support for better models?
7. Reliability vs security
8. Audit trails during building
9. Reference monitors
10. Scatter/Gather
11. ISAs that reduce TOC-TOU risks
12. How do we apportion sec props to hw components, so they compose?
13. Does hw attestation support tell you the right thing? (What is the right thing?) (Sean’s Apache story) (Warren’s tree?)
14. (Hw support for better patch testing?)
15. Can hw make the “client-side SSL” problem easier?
16. Can we build hw that helps with formal models and evaluation? (related to ISA issue above)
And other…
1. Side/Covert Channel

- **What are you trying to solve?**
  - What can we add to make covert channels easier to defend against?
  - Resource consumption
  - Timing channels
  - (QoS risks in routers…)
  - Efficient defense against side-channel
- **What limitations need to be overcome?**
  - Conflicting interaction between goals. Static vs performance
- **Why Now?**
  - PDA: physical partition doesn’t work
  - Side-channel
- **How will you know if you succeed?**
  - Defining a reusable specification of this problem.
2. Exploit DRM

- What are you trying to solve?
  - Can we leverage TCPA/TCG/etc?
  - If not, what are the gaps

- What is the hardest problem that must be solved?
  - Can we specify it formally and prove it rigorously?
  - (ties in to Ruby/Terry’s stuff)

- How is it currently addressed?
  - Appears to be ad-hoc

- What limitations need to be overcome?
  - Analysis capability
  - Understanding of usage model

- Why Now?
  - It’s happening anyway

- How will you know if you succeed?
  - Use by industry?
3. Virtualization

- What are you trying to solve?
  - Our ability to specify what is virtualizable
  - Hw to support virtualization and isolation
  - Vertical partitions...horizontal partitions
- What is the hardest problem that must be solved?
  - Understanding what it means
  - How far up does it go?
  - What is sufficient for today’s problems
- How is it currently addressed?
  - Vmware? Kernel-mode drivers
- What limitations need to be overcome?
  - Hw doesn’t quite do it
  - Gap in understanding of concept
- Why Now?
  - Extra headroom
  - Legacy investment
- How will you know if you succeed?
  - Ia32, ia32e
4. Tagged Architecture

• What are you trying to solve?
  – Easier, efficient management of MLS (and DRM?)

• What is the hardest problem that must be solved?
  – Legacy infrastructure doesn’t do it

• How is it currently addressed?
  – It isn’t

• What limitations need to be overcome?
  – Multilevel Secure system

• Why Now?
• How will you know if you succeed?
  – R6000, 432…

• (Oops. IBMAs400 is a tagged architecture. So why is there a perception that “tagged archs failed”? And is there time for a renaissance?)
What are you trying to solve?
- Encrypted functions, etc, provide impractical but useful security tools

What is the hardest problem that must be solved?
- Can hw design usefully exploit these tools

How is it currently addressed?
- ORAM, hash trees

What limitations need to be overcome?
- Computational model needs to be tweaked

Why Now?
- Extra headroom
- Some progress

How will you know if you succeed?
- Real prototypes, for real problems
Other Problems

- 6. Hw support for better models?
- 7. Reliability vs security
- 8. Audit trails during building
- 9. Reference monitors
- 10. Scatter/Gather
- 11. ISAs that reduce TOC-TOU risks
- 12. How do we apportion sec props to hw components, so they compose?
- 13. Does hw attestation support tell you the right thing? (What is the right thing?) (Sean’s Apache story) (Warren’s tree?)
- 14. (Hw support for better patch testing?)
- 15. Can hw make the “client-side SSL” problem easier?
  - Trusted display. Other citations:
    - Ye/Smith/Anthony
    - Herzberg
  - Keypad
  - Use of crypto
- 16. Can we build hw that helps with formal models and evaluation? (related to ISA issue above)
  - (LISP machine renaissance)
• Virtualization
  – Formal Specification
  – Layered Abstraction
  – Mechanized Verification

• Routers
  – Reactive, continuously operating
  – Verified MLS operation

• Digital Right Management
  – What abstraction is being provided?
  – Rigorously specifying
  – Mechanized mechanism to ensure that designs meet their specifications.

• Metrics
  – Percentage of systems that can be modeled and analyzed
  – Functional properties verified
  – Timing properties verified

• Buyer/Seller Contracts
  – How? Formal Specification of intent and formal specification of the implementation
  – Mechanized means by which a third party can validate the contract.

• Evaluation Support
  – Checking and verification of embedded specifications
  – Mechanized evaluator support