Summary
Architectural Support for Copy and Tamper Resistant Software

Kiran Yellajyosula
February 14, 2005

1 To be completed before class

What are the problems solved by this paper? (50 words)
Implementing copy protection on software is becoming a necessity due to software piracy for many companies. A hacker tries to modify the data or program so as to reveal critical information. Current architectures are vulnerable to security attacks like spoofing, splicing and replay due to the presence of an untrusted memory model.

What are the approaches attempted by this paper? (50 words)
The authors propose a compartment based execution model called XOM, where the instructions or data of a program are not visible to other programs. Every processor is equipped with a private key built in the hardware and software is compiled for the processor by using a symmetric key which is then encoded using the public key of the processor. They extend the architecture with 6 new instructions to support secure storage at different levels of memory. Additional hardware buffers are placed at the I-cache to identify cache misses and decode instructions before placing them in the cache.

What are the main conclusions of this paper? (50 words)
The paper explains XOM based execution and how it counters the security attacks performed against spoofing and splicing. It explains the extra required hardware and instructions to store data safely at different cache levels and in the memory. The authors discuss about executing part of the applications in XOM mode and the overhead involved with executing the whole program in XOM mode. The authors do not present or explain any results obtained using SimOS simulation environment.

2 To be completed after class

Did this paper address an important issue? Explain. (100 words)

Are the proposed approaches valid? Describe its strength and weakness. (100 words)

Do the results support the conclusions? Explain. (100 words)
Describe the potential future works? (100 words)