Nexus: An Operating System for Trustworthy Computing

Alan Shieh Dan Williams Kevin Walsh
Emin Gün Sirer Fred B. Schneider

Department of Computer Science
Cornell University

Trustworthy Computing

- New hardware for trustworthy computing is emerging
- How best to exploit this new hardware?

Project Overview

- Nexus OS
  - Builds on Trusted Platform Module (TPM)
    - Industry-standard secure coprocessor
    - Simple and pervasive
  - Provides new trustworthy computing abstractions
  - Provides assurance through a small TCB
  - Enables new trustworthy applications

TPM primitives

- Hardware root of trust
- Functionality:
  - Data integrity
  - Key storage
  - Attestation (expects hashes)
- Reasonable starting point...

TPM limitations

- Mismatch between TPM and application needs
  - Holds only a few secrets & keys
  - Attests to a system snapshot
  - Supports only hash-based authentication and authorization

Extant OS limitations

- Existing OSes are not suited for trustworthy computing
  - Linux and Windows simply too big
  - Monolithic architecture $\rightarrow$ violates principle of least privilege
  - No strong isolation between components

Nexus: A New OS

- Nexus OS bridges the gap
  - Generalizes and virtualizes the TPM
  - Enables authorization from semantic properties

Nexus: A New OS

- Supports new abstractions with comparable level of assurance relative to TPM
  - Small TCB
    - Exclude drivers and services (user-level)
    - Exclude secondary storage
  - Fine-grain components $\rightarrow$ restricted policies
  - Strong isolation of components
- ... Respectable performance too!

New abstractions

- Secure memory regions with mandatory access control and persistence
- Active attestation attests to a component's properties and environment.
  - Assigns a descriptive label to component

New Nexus OS abstraction:

Active attestation labels

- Labeling functions generate meaningful, flexible labels from:
  - Result of analysis / PCC
  - Use of reference monitors
  - Run in execution environment
- Unlike hash, captures only property of interest
- Used pervasively in the Nexus
  - E.g., IPC binding & invocation, access to secure memory regions, etc.

New Nexus OS abstraction:

Secure memory regions

- Secure memory regions are used to store sensitive application data
- Guarantees:
  - Integrity
  - Confidentiality
  - Persistence

Secure memory regions

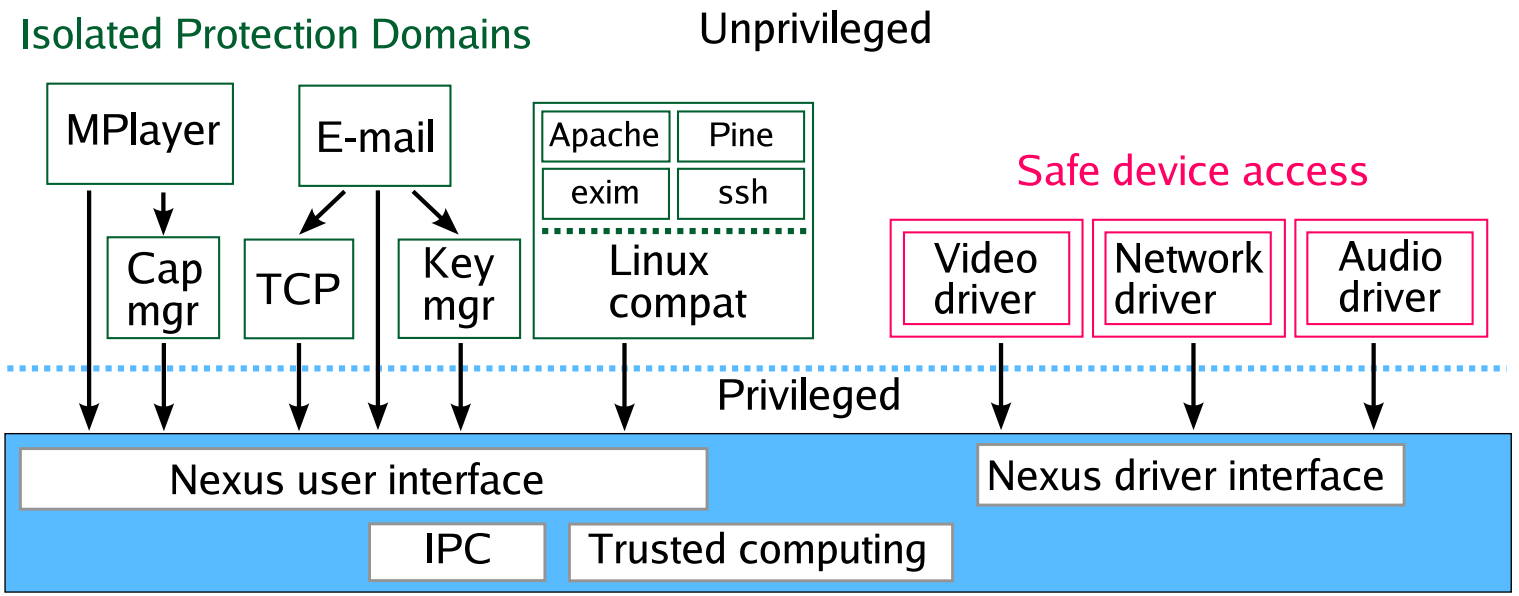
- Admit application-specific optimizations
  - Use knowledge of access patterns to compute optimal block size for hash-trees

    [Williams & Sirer, TNC2004]
- Invaluable for user-level services
  - E.g. Linear capability manager

    ... or any history-dependent policies (via security automata)

Status of the Nexus OS

Working prototype of kernel and new abstractions



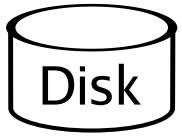
Applications

- Working applications
  - DRM-compliant media player
  - Spam-proof e-mail system
  - Tamper-evident system log
  - Attested MACEDON application
- Real applications provide insights that drive investigation into active attestation

Media player example

- Secure memory regions protect movie data and policy metadata
- Linear capabilities restrict media to a limited number of plays
- Active attestation attests to future behavior of media player
  - Media player does not write to disk
    $\longrightarrow$ This property describes a family of media players

Media player example



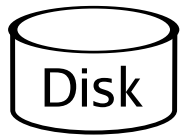
Media storage

Secure memory region encrypts movie

Linear capability manager

Secure memory region protects against tampering of capabilities

Media player example



Media storage

Secure memory region encrypts movie

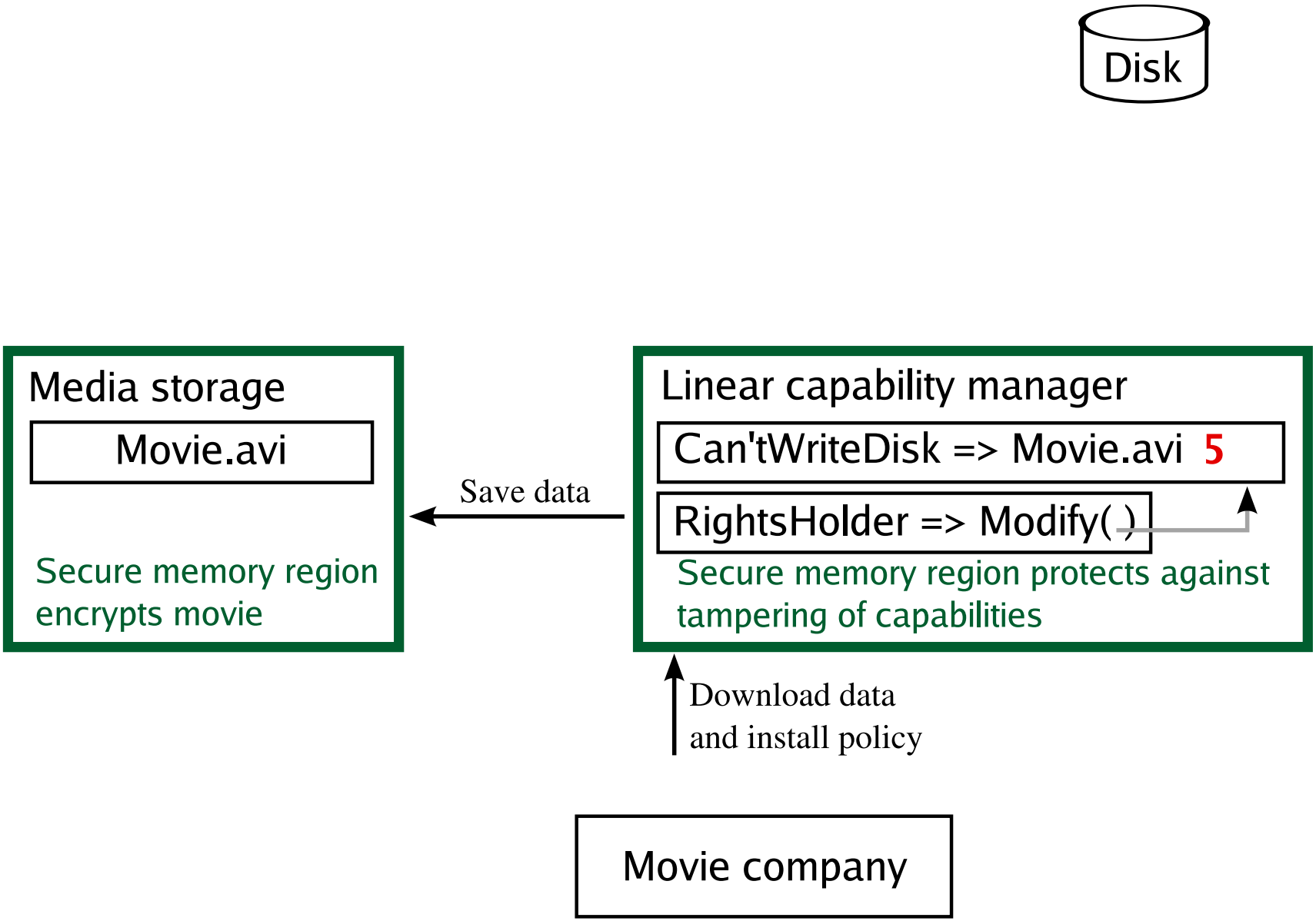
Linear capability manager

Secure memory region protects against tampering of capabilities

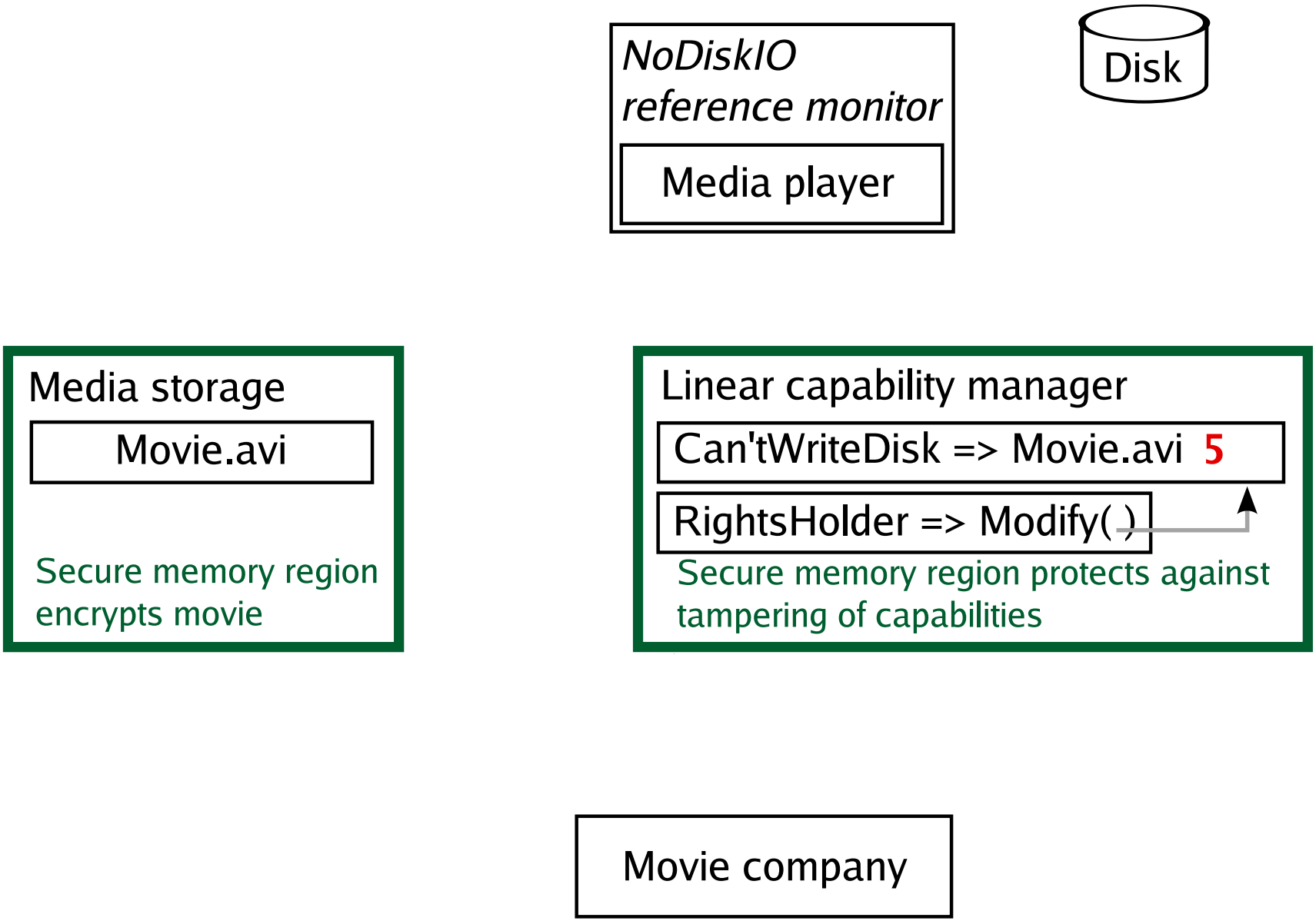
Download data and install policy

Movie company

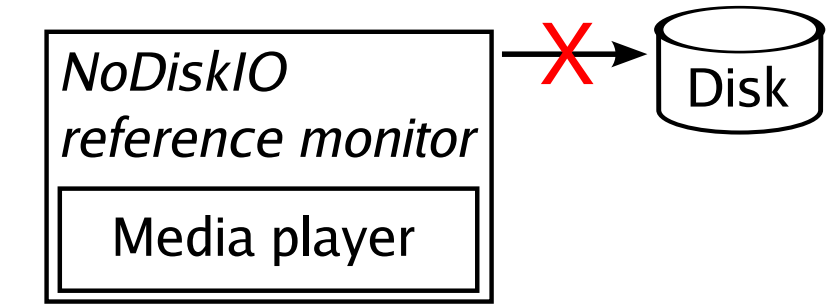
Media player example

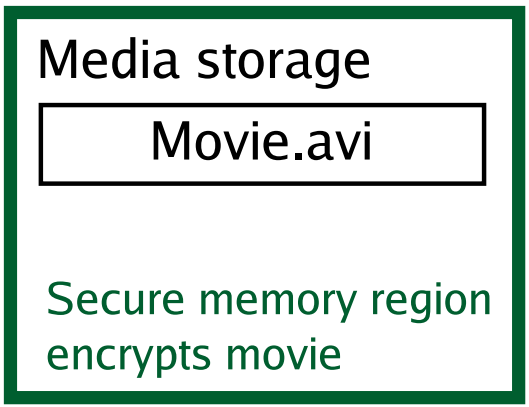


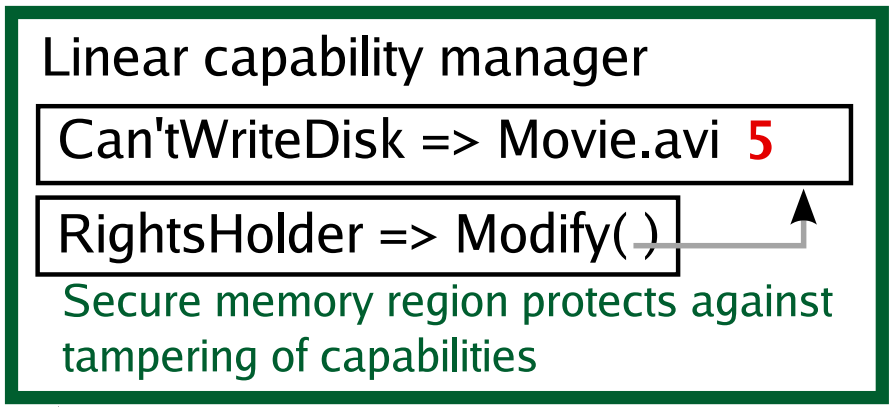
Media player example



Media player example

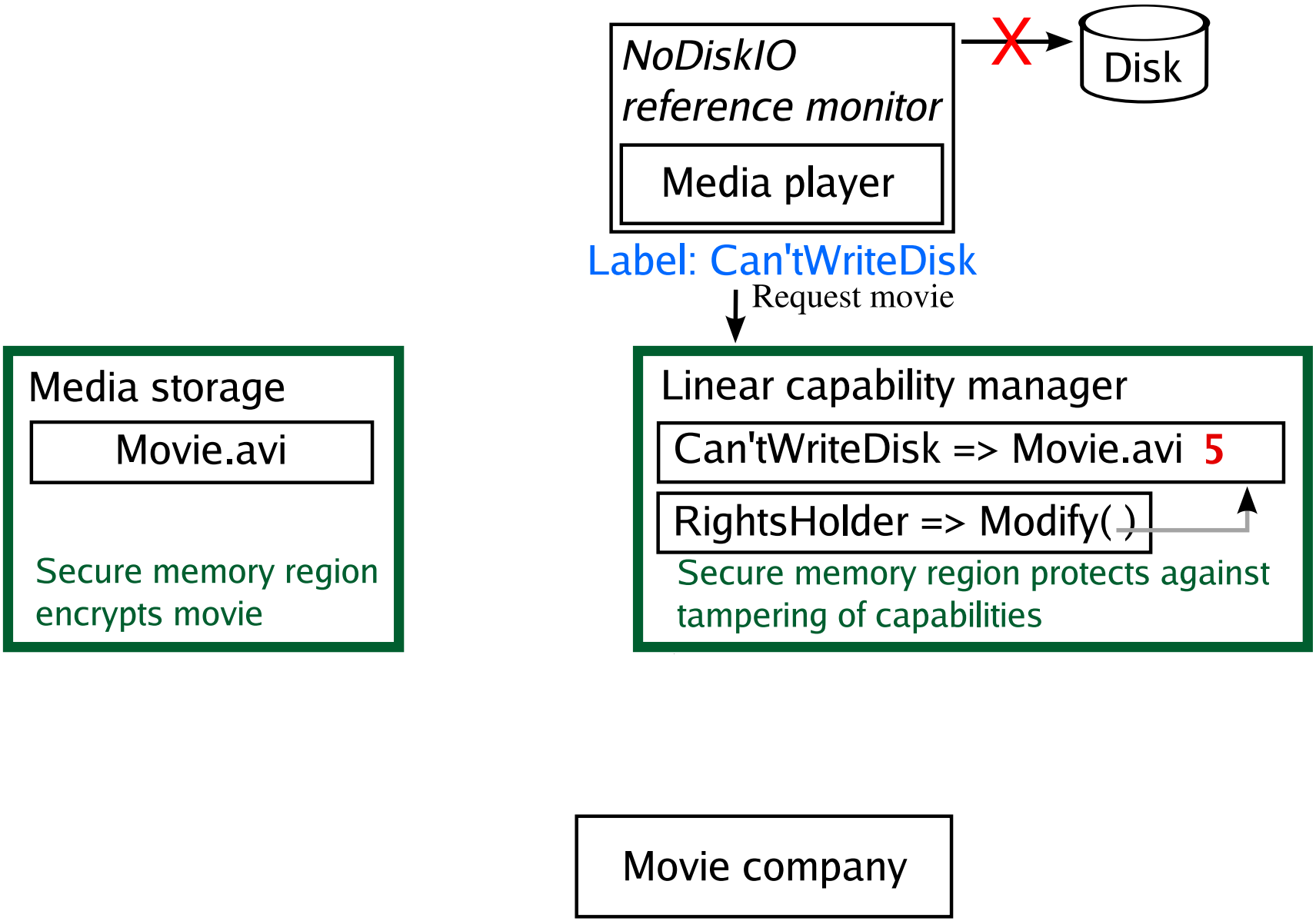




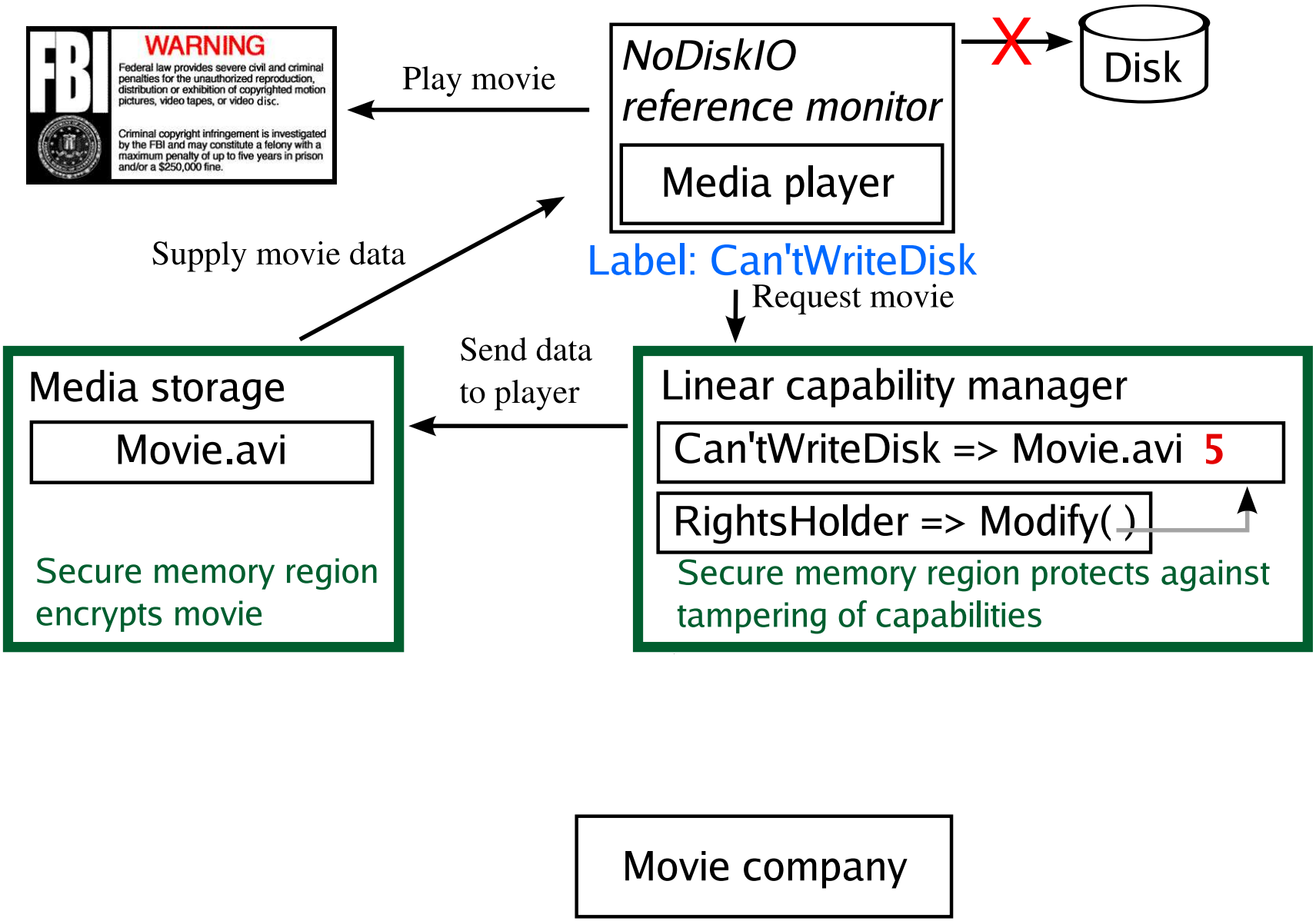


Movie company

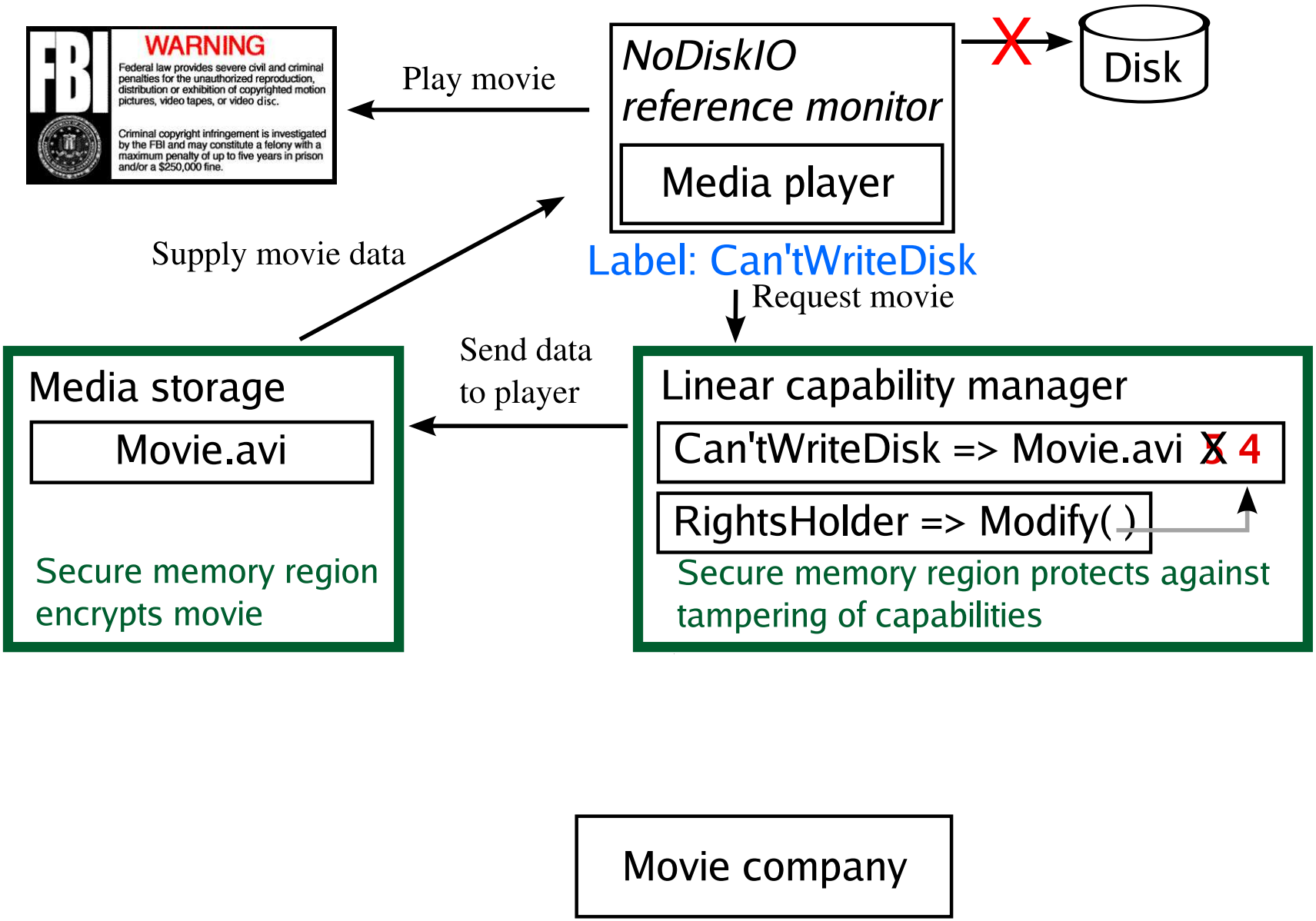
Media player example



Media player example



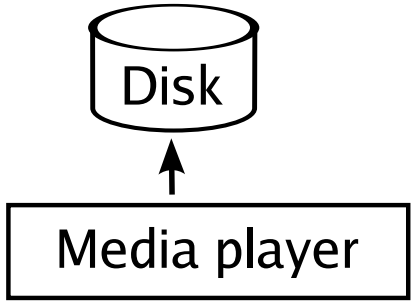
Media player example



Media player example

NoDiskIO reference monitor

Media player



Label: CanWriteDisk

Media storage

Movie.avi

Secure memory region encrypts movie

Linear capability manager

Can'tWriteDisk => Movie.avi ~~3~~ 4

RightsHolder => Modify( )

Secure memory region protects against tampering of capabilities

Movie company

Media player example

NoDiskIO reference monitor

Media player

Disk

Media player

Label: CanWriteDisk

Request movie

Media storage

Movie.avi

Secure memory region encrypts movie

Linear capability manager

Can'tWriteDisk => Movie.avi ~~3~~ 4

RightsHolder => Modify( )

Secure memory region protects against tampering of capabilities

Movie company

Media player example

NoDiskIO reference monitor

Media player

Disk

Media player

Media storage

Movie.avi

Secure memory region encrypts movie

Linear capability manager

Can'tWriteDisk => Movie.avi ~~3~~ 4

RightsHolder => Modify( )

Secure memory region protects against tampering of capabilities

Set new count

Movie company

Hacker

Media player example

NoDiskIO reference monitor

Media player

Disk

Media player

Media storage

Movie.avi

Secure memory region encrypts movie

Linear capability manager

Can'tWriteDisk => Movie.avi ~~3~~ 4

RightsHolder => Modify( )

Secure memory region protects against tampering of capabilities

Set new count

Label: RightsHolder

Movie company

Label: Unknown

Hacker

Media player example

NoDiskIO reference monitor

Media player

Disk

Media player

Media storage

Movie.avi

Secure memory region encrypts movie

Linear capability manager

Can'tWriteDisk => Movie.avi ~~3~~ 4

RightsHolder => Modify( )

Secure memory region protects against tampering of capabilities

Set new count

Label: RightsHolder

Movie company

Label: Unknown

Hacker

Spam-proof e-mail

- Only “non-spam” e-mail clients can sign message with special key
- “Non-spam” clients:
  - Client binary is approved
  - User has typed in text during this execution

Nexus lessons

- Active attestation captures application properties
- Attesting to properties enables meaningful authorization
- Third-parties can provide tools for extracting and enforcing properties

Summary

- Trustworthy computing requires new properties from OS
- The Nexus is a new OS for trusted computing
  - Capture the semantic properties of programs
  - Provide assurance about future behavior
- There are many opportunities for future research
  - New tools for capturing properties
  - New applications that require additional trust