Palladium

Michael Aday
Senior Program Manager
Windows Trusted Platform Technologies
Microsoft Corporation

Agenda

- Introduction and Motivation
- Architecture
- Where’s the Value
- Policy
- Summary
- Q&A

Introduction and Motivation

What is Palladium?

- Palladium (Pd) is a set of new security-oriented capabilities in Windows
  - ➢ Enabled by new hardware
  - ➢ New Software: Trusted Security Kernel (Nexus) and Nexus Computing Agents
- Goal is to "protect software from software"
  - ➢ Defend against malicious software running in Ring 0
- Enable and safeguard decentralized Trusted Computing Base ("TCB") on Open Systems

Trusted Open Systems

- Our OSs are designed for:
  - Features
  - Performance
  - Openness
    - Applications
    - Drivers
    - Core OS components
  - Ease of use, and
  - Security
    - Contrast this with the design of a smartcard OS

Terminology

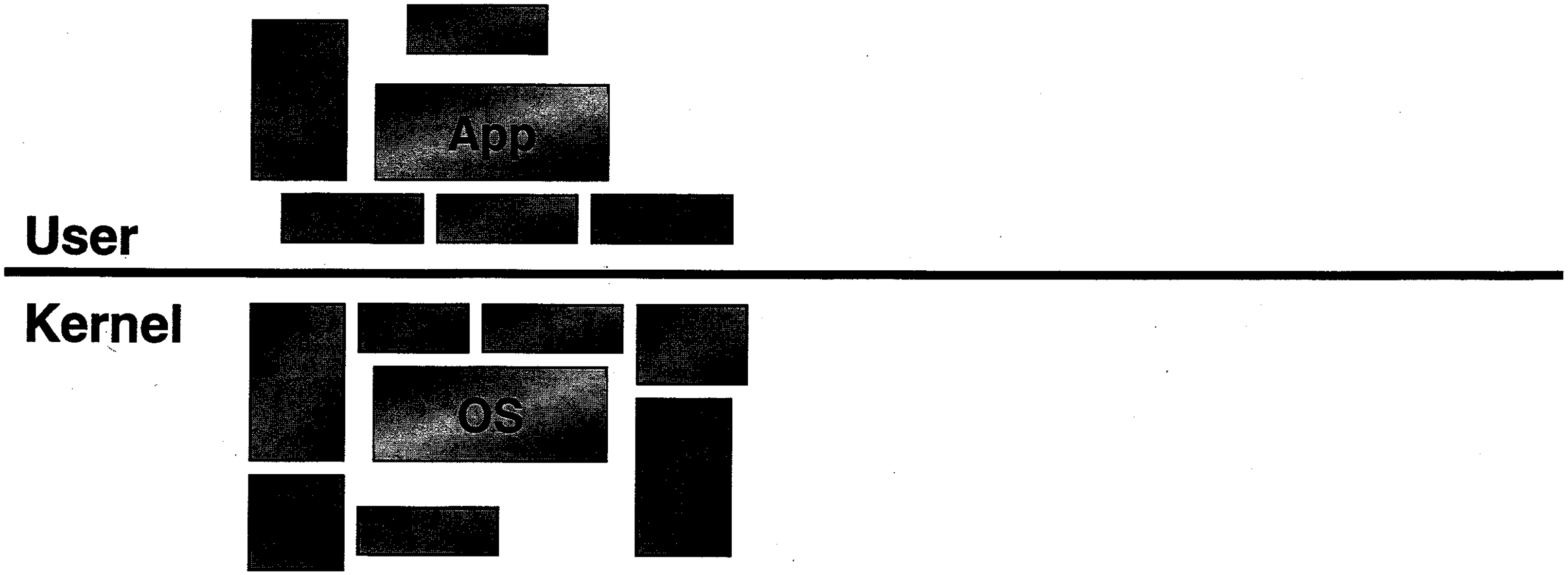
- “Palladium” (a.k.a. Pd)
  - Codename for a set of Windows features built on new HW
- Nexus
  - secure kernel in Pd
- NCA
  - Nexus Computing Agent or Nexus Controlled Agent
- Sealed Storage
  - Method the nexus uses to encrypt and store data
- Authenticated Boot
  - Method used to securely load nexus
- Trusted I/O
  - Secure input and output systems managed by the nexus
- SSC (a.k.a. TPM, SCP, SSP)
  - Security Support Component - Security chip on the motherboard

Mechanism

Construct Security Perimeter Dynamically

- Mechanism couples
  - Software isolation (Curtained Memory --- establish TCB)
  - Software authentication (Attestation --- extend TCB)
  - Secrets for software (Sealed Storage --- persist TCB state)
  - Secure I/O (Include trusted user)
- Credential based security assertions, permissions and authentication
  - A la Lampson, Rivest, Abadi, etc.

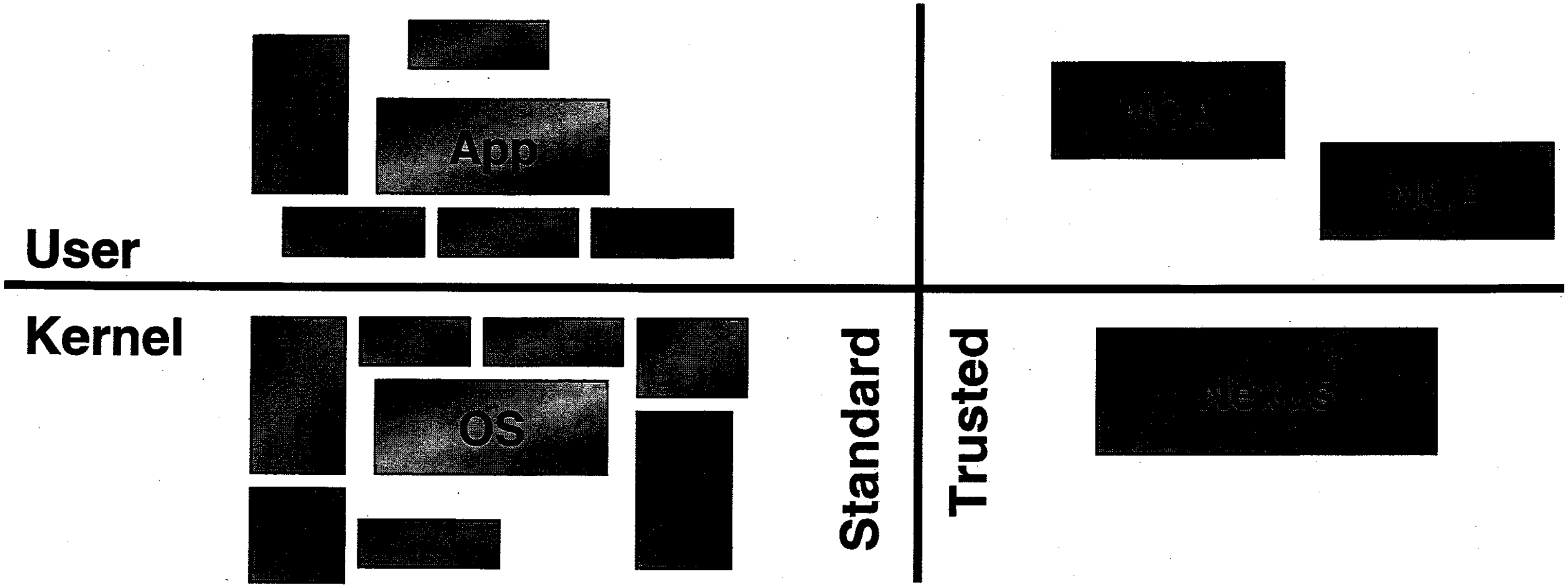
Palladium At 50,000 Feet: 1



- How do you preserve the flexibility and extensibility that contributes so much to the entire PC ecosystem, while still providing end users with a safe place to do important work?
- In particular, how can you keep anything secret, when pluggable kernel components control the machine?

Palladium At 50,000 Feet: 2

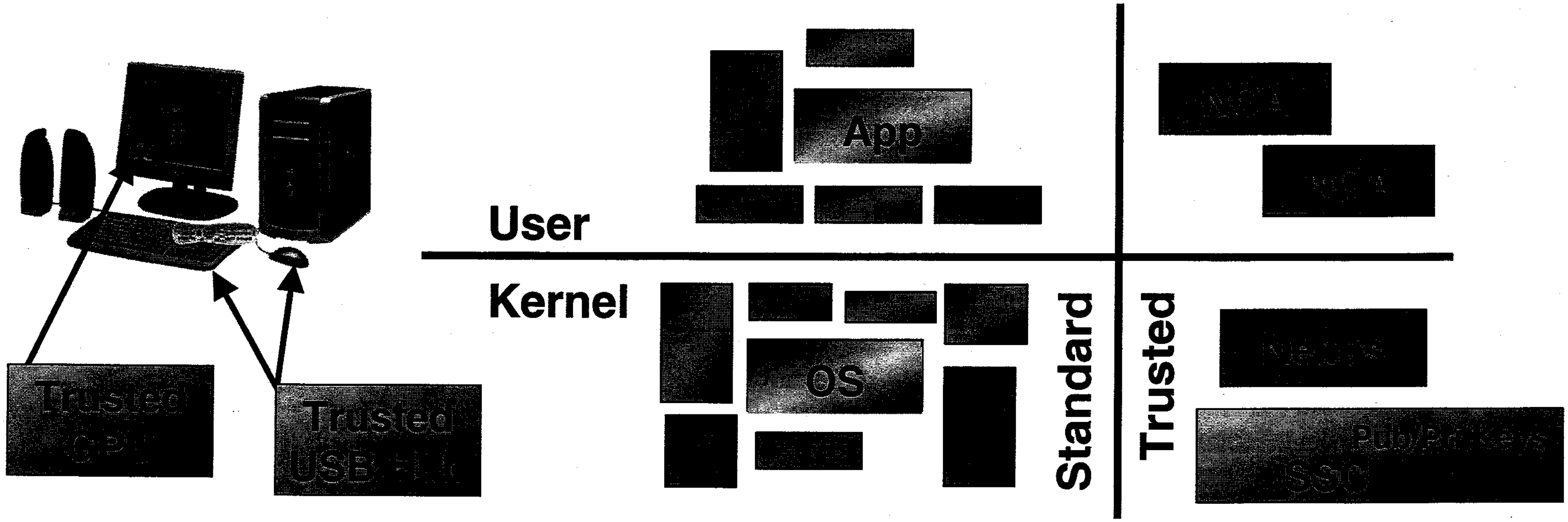
- The solution: subdivide the execution environment by adding a new mode flag to the CPU.



- The CPU is either in “standard” mode or “trusted” mode.
- Pages of physical memory can be marked as “trusted.” Trusted pages can only be accessed when the CPU is in trusted mode.

Palladium At 50,000 Feet: 3

- Agents also need to let the user enter secrets and to display secrets to the user.



- Input is secured by a trusted USB ‘hub’ for KB and mouse that carries on a protected conversation with the nexus.
- Output is secured by a trusted GPU that carries on a crypto-protected conversation with the nexus.
- This gives us “fingertip-to-eyeball” security.

Overarching Principles

- Palladium will be built to the highest standard of security practice.
- A Palladium PC must be able to boot and run any OS and any software from any vendor
- The Palladium Trusted Computing Base (TCB) from Microsoft will be made available for review.
- A Palladium PC must continue to run legacy applications and device drivers.
- Palladium will be designed as an opt-in system.
- Anyone who can write applications for the PC can write applications that take advantage of Palladium.

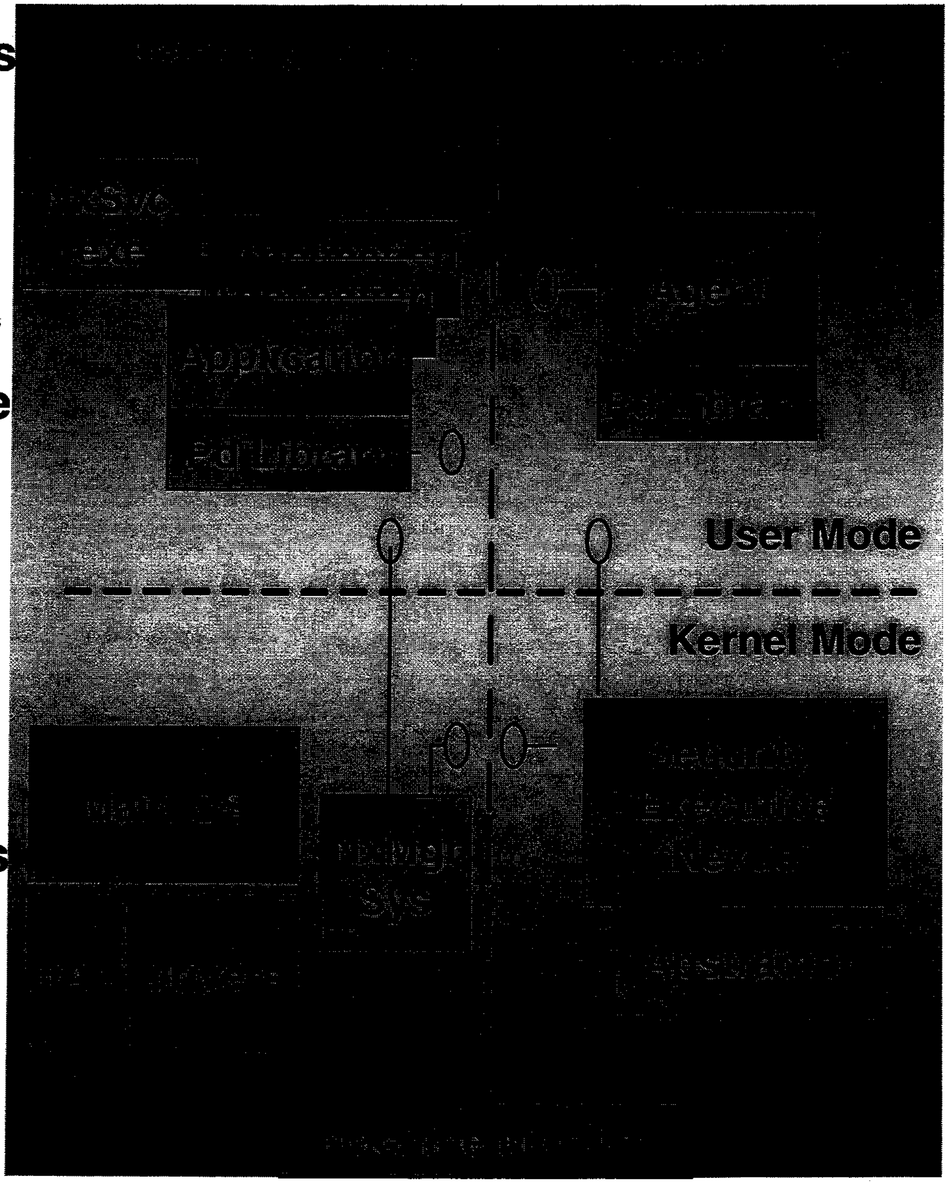
Overarching Principles *(continued)*

- Palladium won't stop piracy.
- Palladium systems will provide the means to protect user privacy better than any operating system does today.
- User information is not a requirement for Palladium to work.
- Palladium may not withstand determined attackers with physical access to an individual machine, but will be highly BORE (Break Once, Run Everywhere) resistant.
- Palladium enables 360° of policy enforcement.

Architecture

How Palladium Works

- Leverages CPU enhancements (new modes) to “wall off” a protected area of memory
- Small Security Kernel (“nexus”) abstracts hardware and provides programmability
- Software components that use secrets run behind the wall (“Nexus Computing Agents” or NCAs)
- Secrets bound to software identity and platform
- Secure user interaction through secure video, keyboard and mouse channels



Nexus in the OS
What's Familiar

- Private address space
- Contain EXE's
  - (may or may not support DLLs)
- Ownership
- Normal process-control block
- Access rights
- Thread creation, etc…

Nexus in the OS

What's Different

- Process separation is stronger
  - Main OS/apps unconditionally excluded
  - Debugging, memory inspection by the Nexus/agents is strictly controlled
- The code that can be loaded into a NCA is restricted by NCA policy
- NCAs have privileged access to one or more cryptographic keys (based on code identity)
  - Basis for authentication and authorization
  - Also decentralized

Palladium Security Model

- Agents have *less* privileges than applications (in general)
- Just because you're protected when running, doesn't mean that you're protected on the disk
- Code identity is a key concept in Pd

SSC
Security Support Component

- Think “smart-card soldered to the motherboard”
- Cheap, fixed-function device
- Contains
  - At least an AES key and an RSA key pair
    - AES key & RSA private key never leave the chip
  - Registers: e.g. the “PCR” (platform configuration register) that contains the digest of the running Nexus
- Must be close to the chipset (e.g. not a real smartcard) because it must be involved in Nexus initialization
- SSC can be TCPA TPM 1.2

Hardware Changes

- CPU changes
- MMU changes
- Southbridge (LPC bus interface) changes
- Security Support Component (SSC)
  - New chip on the motherboard (LPC bus)
- Trusted USB hub
  - May be on motherboard, in keyboard, or anywhere in between
- Trusted GPU

Hardware Services for Nexus

- Hardware provides nexus with:
  - Strong process isolation
  - Per nexus keys for persistent secret protection
  - Secure path to and from the user
  - Attestation
- Attestation breaks new ground
  - Facts about “things” (SW, users, machines, services) can be proved to (and believed by) remote entities.
- Nexus returns the favor for its NCAs
  - Nexus to NCA services can be a bit richer

Where’s the Value?

Applications

- System Management
  - ➢ Secure Boot
  - ➢ Administration
    - ▪ Installation, upgrade and update management
    - ▪ Login, key/password management, crypto engine
    - ▪ Monitoring machine health including virus checking
- High assurance applications
  - ➢ Banking, secure transactions
  - ➢ Private IM
- Shared Resources
  - ➢ Kiosks
  - ➢ Home Machine using corporate apps

Applications
(continued)

- Collaborative Apps
  - Multiplayer Games
  - Negotiations
  - Bidding
- Decentralize Access Control
  - Web Services
  - Cross Domain Authentication and Authorization
- DRM
  - Enterprise
  - Privacy/Consumer
    - Identity and usage information, health and financial records
  - Mass market content
    - Books, movies, audio, video

Attestation

- Attestation lets a remote client know what SW is running
  - OS / Nexus
  - Application
  - Client policy (virus checker, admin access, etc.)
- Attestation is an authentication technology
  - But more than “simple signing”
- Enables authentication of a software configuration (nexus, application process)

Secure User Input and Output

- Is the banking application being driven by a user or a virus?
- Is a Trojan modifying the dialog that contains the transaction I'm authorizing?
- Is a rogue application viewing the video frame buffer while I type a password?
- User / Application Relationship
  - Protected path between user and application

Pd Misconceptions

- Palladium will censor or disable content without user permission
  - As designed, no such mandatory policy can be in Pd
- Palladium will lock out vendors Microsoft doesn't approve of
  - No required Microsoft signatures to use Pd
- Palladium is not controlled by user
  - All Pd programs can be run only if authorized by user
- Palladium is "super" virus spreader
  - Palladium applications do not run at elevated privilege
- Palladium NCA is not debuggable
  - Yes it is. Tag in manifest to turn on debugging.

Palladium Security Model

- Underlying access control system
  - MAC/DAC
- Based on credentials
  - Code credentials
  - User credentials
- Layered model of security
  - Seal/Unseal can be understood as special instances of a code based ACL policy
- Mandatory access control policy
  - Likely candidates: MLS and Domain Type Enforcement

Policy Issues

- Some of the technical issues we have to solve to make Palladium successful also have policy components to them. For example:
- How do we in practice build an "attestable" TCB?
  - "Attestable" == open, auditable, comprehensible and provable to a remote party
- Since the Pd RSA key pair is unique to the platform, what steps should we take to defend against traffic analysis of user behavior?

Privacy of Machine Identities

- The issue: Palladium uses at least two sets of unique hardware keys (one AES key, one RSA key pair):
  - Essentially equivalent to unique machine identifiers
  - But this is the only way we can keep your stuff safe!
- Sealed Storage:
  - Uses a unique AES key, but the algorithms are:
    - Opt-in (user designates what software can access functions)
    - Randomizing (can't decide whether two ciphertexts were created on same machine)
- Attestation:
  - Uses a unique RSA key, but is designed to authenticate the platform
    - Opt-in (user designates what software can access functions)
    - We strictly control HW authentication key disclosure
- The hardware has privacy safeguards built into it
  - Access to the RSA public key components is restricted
  - In the current design, only one export of RSA public key is allowed per power cycle

More Information

- **Subscribe to Newsletter:**
  Send email to PdInfo@microsoft.com

Questions?

Microsoft Government

This presentation is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.