# Policy Machine
## Enabling an Enterprise-wide, Data Centric Operating Environment

David Ferraiolo & Serban Gavrila
National Institute of Standards and Technology

# Towards a New Foundation

- If properly designed Access Control (AC) and in particular, the Policy Machine can be more fundamental to computing than one might expect.

  Theory: At a basic level AC and Data Services (DSs) can be built from the same elements.

- Some consequences
  - Computing goes from multiple operating environments each delivering different DSs TO a single operating environment delivering all DSs
    - Single sign-on
    - Data interoperability among DSs
    - Comprehensive policy enforcement across DSs

# Enterprise Computing

- A basic objective of enterprise computing (via a data center, a cloud, etc.) is the controlled delivery of DS capabilities to its users.
- DS capabilities enable operations to read, manipulate, manage, and/or share data.
  - E.g., Office applications, email, workflow management, enterprise calendar, time and attendance, and records management.
  - Not all DSs are applications. E.g., file and access control management are DS that are typically implemented in system software.

# Enterprise Computing (2)

- ACs limit which users can perform which operations on which objects (resource data and administrative data), in accordance with policy.
- AC is often defined as finite state machine:
  - AC data used to express access state in which users can perform operations on objects,
  - a set of environment specific operations and a set of administrative operations,
  - a set of functions for enforcing the access state on user requests to execute operations on objects based on the current configuration of the AC data.

# Today's Operating Environments (OEs)

- Capabilities of DSs are supported by a wide variety of OEs
  - E.g., operating system instances, operating system applications, web services, middleware, and database and database applications
- As OEs differ, so do the DS operation and object types that they implement.
  - E.g., Email enables capabilities to send, receive, and read messages and attachments; while workflow management enables users to specify and impose workflow instances and approve or reject work items.
- OEs do not necessarily recognize each other's data or operation types.
  - E.g., an OS that controls access to files, may view a database as just one giant file and not be aware of the DBMS object types, and email applications may distribute files to users regardless of an operating system's protection settings on those files.

# Management and Usability Challenges

- Administrators must contend with a multitude of OE domains
  - Create and manage numerous user accounts for each user
  - Coordinate AC policies and manage privileges across different OEs with different operation and object types, through different administrative interfaces.
- Ordinary users and administrators must authenticate to and establish sessions within different OEs in order to exercise legitimate capabilities.

# Policy Enforcement Challenges

- Even if properly coordinated AC policies are not always globally enforced over DSs.
- A large variety of access control policies have been specified, but only a relatively small subset of these policies can be enforced through off-the-shelf technology, and even a smaller subset can be enforced by any one mechanism.

# Policy Machine

- An AC mechanism,
- Driven by a unification of AC and DSs,
- Implemented in terms of their common and underlying data elements, relations, operations, and functions,
- Providing a common API and OE for DSs and AC.

# Practical Benefits

- Enables an enterprise-wide OE that can implement and execute capabilities of arbitrary DSs, and can specify and enforce mission tailored AC policies over those executions.
- The data of DSs naturally interoperate and users can see and consume all their authorized data (regardless of its kind) under a single authenticated session.
- Vulnerabilities and complexity of implementing AC in DSs are eliminated/reduced.

# How?

- Operations (ops) of different DSs (e.g., read, send, approve) can be implemented as combinations of (1) simple read or write ops on data and (2) admin ops on AC data that alter the access state under which users can read or write data.
- Users can obtain DS capabilities and AC policies can be enforced through configurations of the same AC data and a fixed set of functional components.

# PM Data Elements and Relations

- Basic elements
  - Users, processes, operations (r, w, admin. ops), and data objects
- Containers
  - User attributes, object attributes, and policy classes
- Relations
  - Assignments (define membership in containers)
  - Associations (define privileges)
  - Prohibitions (denies for users and processes capabilities)
  - Event pattern/Response (can dynamically alter the current access state)

# Reference Implementation

- **Data Services**: Office applications, file management, e-mail, workflow, records management, cut/copy-paste
- **Policies:** Combinations of discretionary, mandatory, and history based access controls:
  - DAC
  - RBAC
  - History and object-based Separation of Duty
  - Forms of confinement (read with restrictive write)
    - E.g., Only doctors can read medical records, MLS
    - Trojan resistant leakage
    - Sensitive data can't be leaked by email or cut/copy - paste
  - Tracking access - I know who can currently access to my data
  - Chinese wall (conflict of interest)
- Soon to be offered as open source

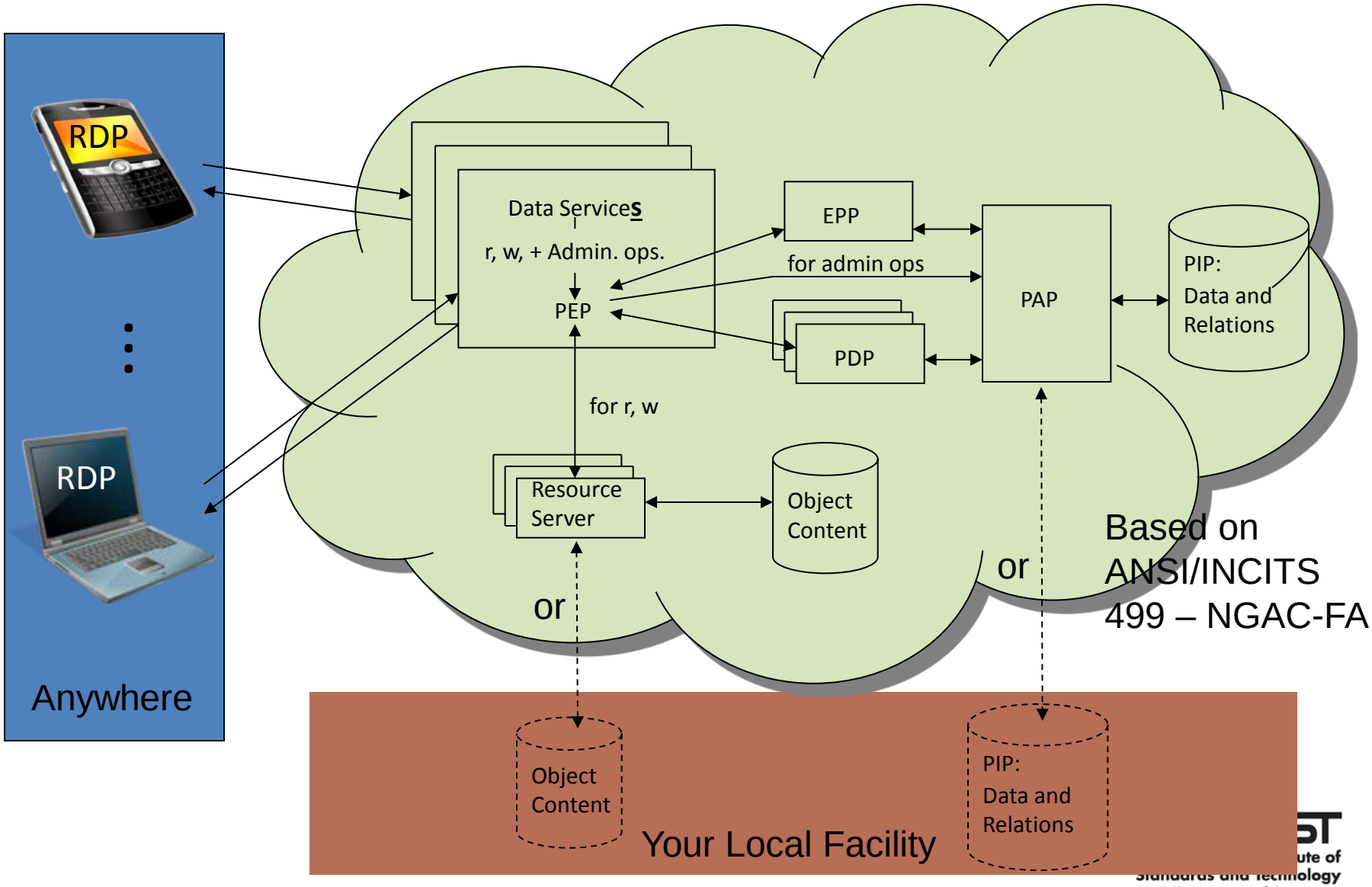
# How does the PM Work?

- User logs on to the PM,
- PM logically presents the user with all his/her accessible resources (e.g., files, inbox, work items ...),
- User requests access to resources through a process,
- PM mediates the access to resources by those processes based on capabilities derived through user and object attributes, associations, and policy classes, and process and user prohibitions ,
- Machine state may dynamically change as users and processes access resources.
- Note: Policy is created through data configuration alone

# Cloud-Like Deployment

- IaaS is an OE that implements the Policy Machine and composed of its functional components (i.e., PEPs, PDPs, … ) that run in VMs
- Users and objects are provisioned, and DSs are selected by the subscriber.
- DSs may be provided as SaaS or PaaS so long as they conform to the Policy Enforcement Point (PEP) API.
- Policies are imported from a library of predefined PM data and relation configurations or configured from scratch, by the subscriber - POLICYaaS

# PM Deployment/Architecture



# Other Benefits of Deployment

- Can be configured so that data can't be stored or leaked into local environment
  - If lap/desk top, tablet, smart phone is lost, stolen, or damaged, data is not compromised/lost
  - No need to encrypt data locally
- Device can be used for business (through RDP) and personal purposes
- All operations exist in PM Cloud so nothing needs to be installed or updated on local device
- Desktop computing aaS - Teleworking

# Some Noted Use Cases

Not a replacement, but under some circumstances a better alternative to other forms of computing:

- Ad hoc Collaborations and information sharing
- Financial - Separation of duty (SoD) and conflicts of interest (CoI)
- Health Care - Sharing and protecting Health Records
- IC – Combinations of Mandatory (prevent data leakage across communities of interest) and Discretionary Policies

# Thank You!


Contact Information

David Ferraiolo

dferraiolo@nist.gov

301-975-3046

Serban Gavrila

serban.gavrila@nist.gov

301-975-4343


For more information see:

http://csrc.nist.gov/pm/


NIST
National Institute of Standards and Technology
U.S. Department of Commerce