# CS 591: Introduction to Computer Security

## Lecture 5: Integrity Models

James Hook
(Some materials from Bishop, copyright 2004)

4/15/08 21:37

# Last lecture

- Discussion?

# Last Lecture

- Bell LaPadula Confidentiality
- Lattice of security levels
  - No read up
  - No write down
- DG/UX realization of Bell LaPadula

# Ross Anderson on MLS

"… the contribution of the MLS model is not all positive. There is a tactical problem, and a strategic one.

"The tactical problem is that the existence of trusted system components … has a strong tendency to displace critical thought. …

"… MLS systems, by making the classification process easier but controlled data sharing harder, actually impair operational effectiveness."

[Comments at end of 7.6 in first edition]

# Objectives

- Integrity models in context
- Introduce integrity models
- Begin hybrid models

# Plumbing Analogy

- Potable water
  - Cold
  - Hot
- Storm water
- Gray water
- Brown water

- Shower
- Toilet
- Washing machine
- The “CSO” problem

- What comes out of the tap?

# Simple integrity

- Integrity Levels
  - Potable water
    - Cold
    - Hot
  - Storm water
  - Gray water
  - Brown water

- Multilevel devices:
  - Shower
  - Toilet
  - Washing machine

- What kind(s) of water can people easily obtain (read/execute)?
- What kind(s) of water can people produce (write)?

# Integrity is Well Motivated

- Bank balance adds up
- How much inventory do I have?
- Did I pay my employees correctly?
- Did I bill for all my sales?
- Which outstanding invoices have been paid?

# Integrity

- A system has integrity if it is trusted
- Integrity is not just a property of the information system
- A perfect information system could lose integrity in a corrupt organization

# Integrity Principles

- Separation of Duty:
  - Functional Separation:
    - If two or more steps are required to perform a critical function, at least two different people should perform the steps
  - Dual Control:
    - Two or more different staff members must act to authorize transaction (e.g. launch the nuclear missiles)

# Integrity Principles

- Separation of Function
  - Developers do not develop new programs on production systems
- Auditing
  - Record what actions took place and who performed them
  - Contributes to both recovery and accountability

# Discussion

- In a fully manual paper ballot system how is integrity maintained?
  - Examples of Separation of Duty?
    - Functional separation or dual control?
  - Examples of Separation of Function?
  - Examples of Auditing?

# Chapter 6: Integrity Policies

- Overview
- Requirements
- Biba’s models
- Clark-Wilson model

# Overview

- Biba’s model
- Clark-Wilson model

# “Low water Mark”

- Low water mark principle:
  - the integrity of an object is the lowest level of all the objects that contributed to its creation
- Biba’s first, and simplest model, was the low water mark model
  - Tends to be too simplistic
  - Everything gets contaminated

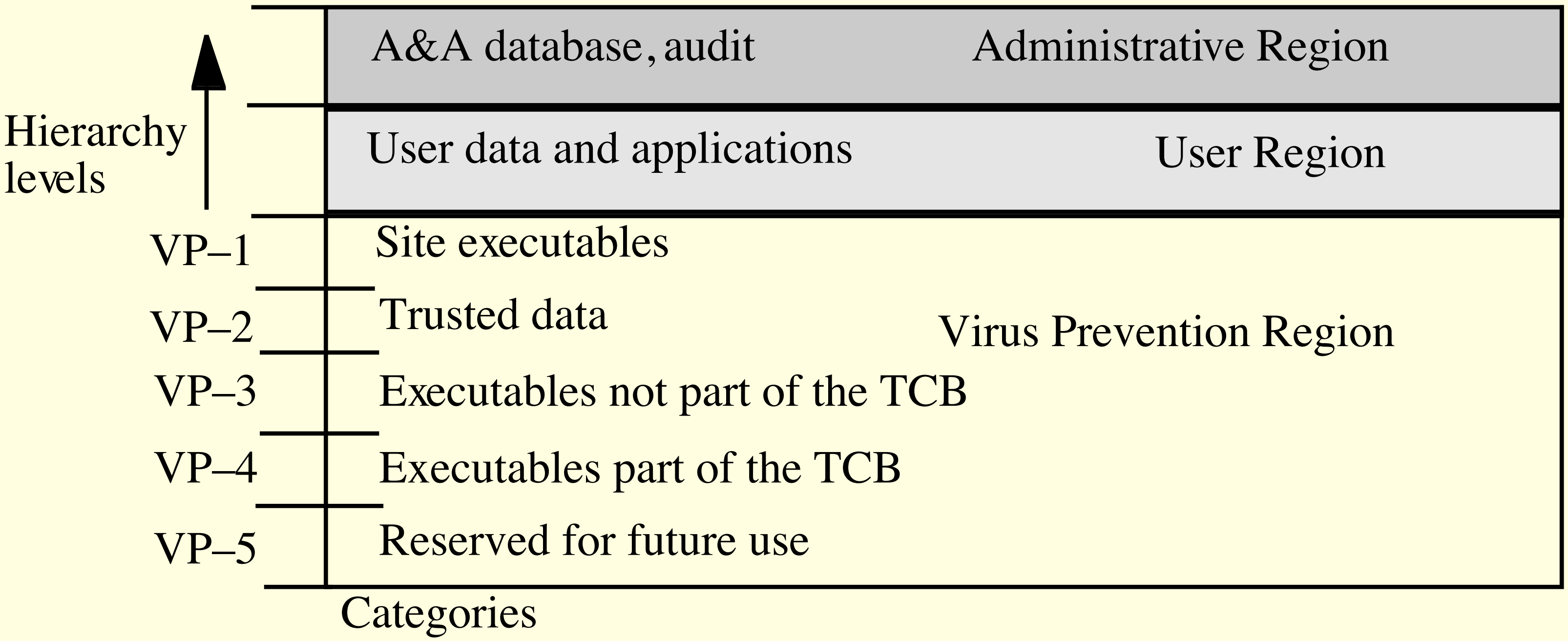
# Biba Refinements

- Ring principle (2nd Biba model)
  - Allow reads arbitrary untrusted data
  - Track execution and writes
    - Execution is seen as a subject creating a new subject at or below current integrity level
    - Can write at or below current integrity level

# Biba's Strict Integrity model

- Third Biba model
- Integrity levels in a lattice (similar to BLP)
  - Subject can read object if $i(s) \leq i(o)$
  - Subject can write object if $i(o) \leq i(s)$
  - Subject s1 can execute s2 if $i(s2) \leq i(s1)$
- Dual to BLP

# MAC Regions



A&A database, audit — Administrative Region

User data and applications — User Region

Hierarchy levels

VP–1 Site executables

VP–2 Trusted data

VP–3 Executables not part of the TCB

VP–4 Executables part of the TCB

VP–5 Reserved for future use

Virus Prevention Region

Categories

IMPL_HI is “maximum” (least upper bound) of all levels

IMPL_LO is “minimum” (greatest lower bound) of all levels

4/15/08 21:37

# Intuition for Integrity Levels

- The higher the level, the more confidence
  - That a program will execute correctly
  - That data is accurate and/or reliable
- Note relationship between integrity and trustworthiness
- Important point: *integrity levels are **not** security levels*

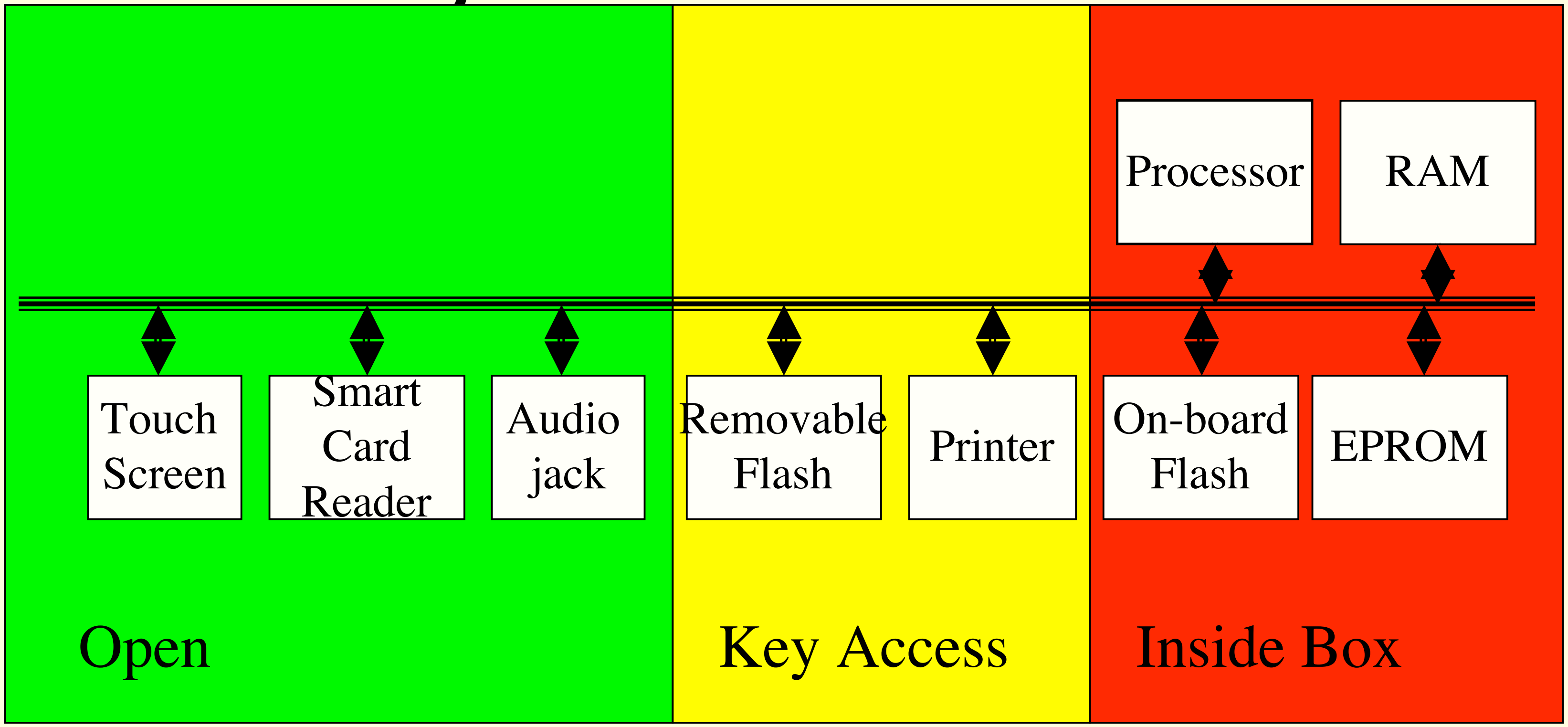
# Biba’s Model

- Similar to Bell-LaPadula model
  1. $s \in S$ can read $o \in O$ iff $i(s) \leq i(o)$
  2. $s \in S$ can write to $o \in O$ iff $i(o) \leq i(s)$
  3. $s_1 \in S$ can execute $s_2 \in S$ iff $i(s_2) \leq i(s_1)$
- Add compartments and discretionary controls to get full dual of Bell-LaPadula model
- Information flow result holds
  - Different proof, though

# LOCUS and Biba

- Goal: prevent untrusted software from altering data or other software
- Approach: make levels of trust explicit
  - *credibility rating* based on estimate of software's trustworthiness (0 untrusted, *n* highly trusted)
  - *trusted file systems* contain software with a single credibility level
  - Process has *risk level* or highest credibility level at which process can execute
  - Must use *run-untrusted* command to run software at lower credibility level

# Voting Machine with Biba



- Subjects? Objects? Integrity Levels?

4/15/08 21:37

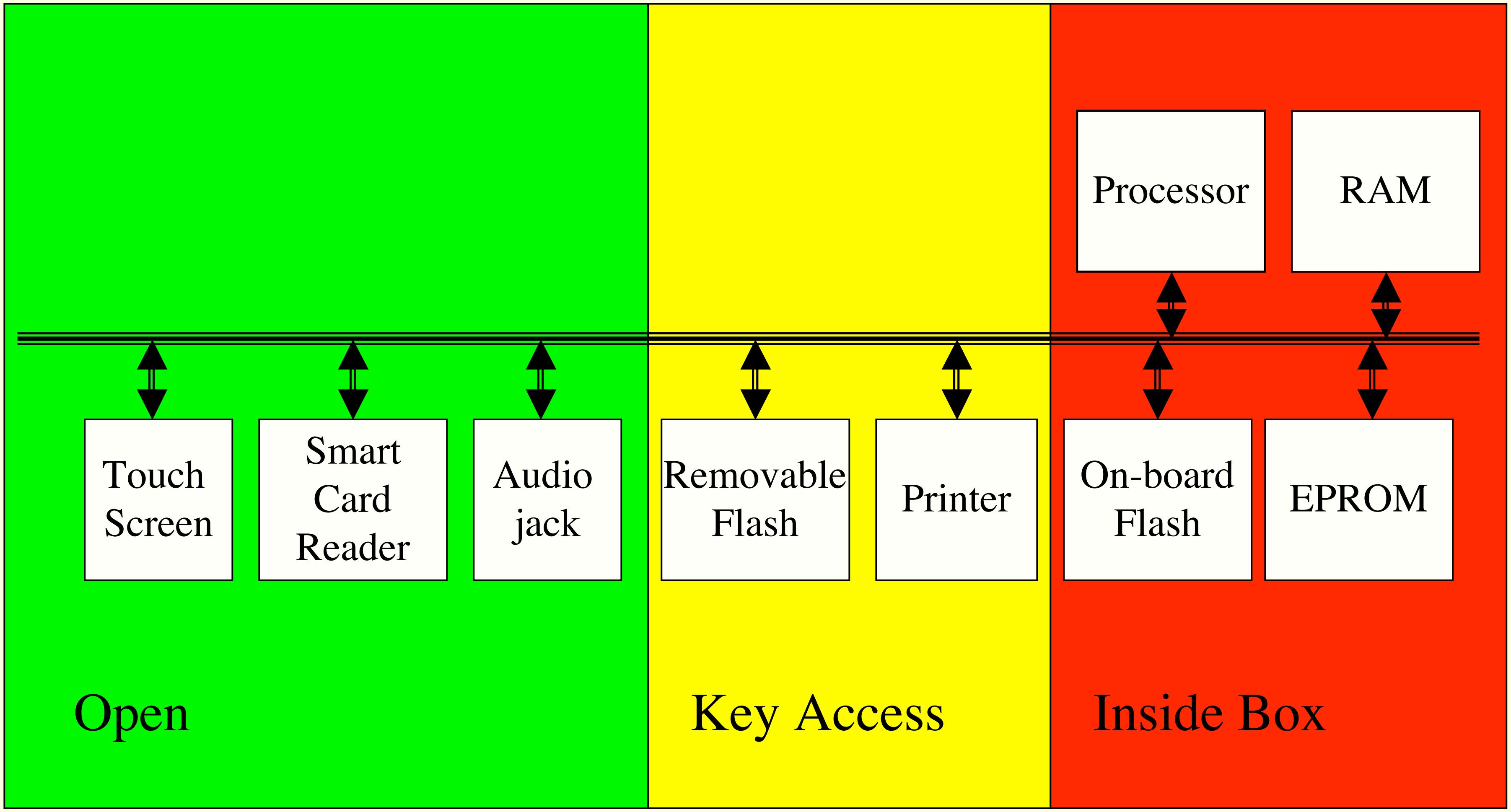
# Example

- Elaborate the Biba integrity model for this system by assigning integrity levels to all key files. Specifically assign integrity levels for creating or modifying these files.
- Several known exploits of the system rely on infection via removable media. Propose a mechanism that uses the trusted authentication mechanism and integrity model to prevent these exploits.

# Example (cont)

- Argue that the intended operations can be carried out by appropriate subjects without violating the policy.
- Argue that with these mechanisms and a faithful implementation of the integrity model that Felten's vote stealing and denial of service attacks would not be allowed.

# Voting Machine Architecture



4/15/08 21:37

# Boot Process

- Boot device specified by hardware jumpers (inside box)
  - EPROM
  - on-board flash (default)
  - ext flash
- On Boot:
  - Copy bootloader into RAM; init hardware
  - Scan Removable flash for special files
    - "fboot.nb0" => replace bootloader in on-board flash
    - "nk.bin" => replace OS in on-board flash
    - "EraseFFX.bsq" => erase file system on on-board flash
  - If no special files uncompress OS image
  - Jump to entry point of OS

# Boot (continued)

- On OS start up:
  - run Filesys.exe
    - unpacks registry
    - runs programs in HKEY_LOCAL_MACHINE\Init
      - shell.exe (debug shell)
      - device.exe (Device manager)
      - gwes.exe (graphics and event)
      - taskman.exe (Task Manager)
  - Device.exe mounts file systems
    - \ (root):  RAM only
    - \FFX:  mount point for on-board flash
    - \Storage Card:  mount point for removable flash

# Boot (continued)

- Customized taskman.exe
  - Check removable flash
    - explorer.glb => launch windows explorer
    - *.ins => run proprietary scripts
      - (script language has buffer overflow vulnerabilities)
      - used to configure election data
    - default => launch “BallotStation”
      - \FFX\Bin\BallotStation.exe

# BallotStation

- Four modes: pre-download, pre-election testing, election, post-election
- Mode recorded in election results file
  - \Storage Card\CurrentElection\election.brs

# Stealing Votes

- Malicious processes runs in parallel with BallotStation
- Polls election results file every 15 seconds
  - If election mode and new results
    - temporarily suspend Ballot Station
    - steal votes
    - resume Ballot Station

# Viral propagation

- Malicious bootloader
  - Infects host by replacing existing bootloader in on-board flash
  - subsequent bootloader updates print appropriate messages but do nothing
- fboot.nb0
  - package contains malicious boot loader
  - and vote stealing software

# Discussion

- Having developed this design, it is now time to critique it!
  - Are you satisfied with the protection against external threats?
  - Are you satisfied with the protection against insider threats?

# Clark-Wilson Integrity Model

- Integrity defined by a set of constraints
  - Data in a *consistent* or valid state when it satisfies these
- Example: Bank
  - *D* today's deposits, *W* withdrawals, *YB* yesterday's balance, *TB* today's balance
  - Integrity constraint: $D + YB - W$
- *Well-formed transaction* move system from one consistent state to another
- Issue: who examines, certifies transactions done correctly?

# Entities

- CDIs: constrained data items
  - Data subject to integrity controls
- UDIs: unconstrained data items
  - Data not subject to integrity controls
- IVPs: integrity verification procedures
  - Procedures that test the CDIs conform to the integrity constraints
- TPs: transaction procedures
  - Procedures that take the system from one valid state to another

# Certification Rules 1 and 2

CR1 When any IVP is run, it must ensure all CDIs are in a valid state

CR2 For some associated set of CDIs, a TP must transform those CDIs in a valid state into a (possibly different) valid state

- Defines relation *certified* that associates a set of CDIs with a particular TP
- Example: TP balance, CDIs accounts, in bank example

# Enforcement Rules 1 and 2

ER1 The system must maintain the certified relations and must ensure that only TPs certified to run on a CDI manipulate that CDI.

ER2 The system must associate a user with each TP and set of CDIs. The TP may access those CDIs on behalf of the associated user. The TP cannot access that CDI on behalf of a user not associated with that TP and CDI.

- System must maintain, enforce certified relation
- System must also restrict access based on user ID (*allowed* relation)

# Users and Rules

CR3 The allowed relations must meet the requirements imposed by the principle of separation of duty.

ER3 The system must authenticate each user attempting to execute a TP

- Type of authentication undefined, and depends on the instantiation
- Authentication *not* required before use of the system, but *is* required before manipulation of CDIs (requires using TPs)

# Logging

CR4 All TPs must append enough information to reconstruct the operation to an append-only CDI.

- This CDI is the log
- Auditor needs to be able to determine what happened during reviews of transactions

# Handling Untrusted Input

CR5 Any TP that takes as input a UDI may perform only valid transformations, or no transformations, for all possible values of the UDI. The transformation either rejects the UDI or transforms it into a CDI.

- In bank, numbers entered at keyboard are UDIs, so cannot be input to TPs. TPs must validate numbers (to make them a CDI) before using them; if validation fails, TP rejects UDI

# Separation of Duty In Model

ER4 Only the certifier of a TP may change the list of entities associated with that TP. No certifier of a TP, or of an entity associated with that TP, may ever have execute permission with respect to that entity.

- Enforces separation of duty with respect to certified and allowed relations

# Discussion

- How can we apply CW to Voting Machine?
  - Constrained Data Items:
  - Integrity Constraints:
  - Unconstrained Data Items:
  - Transaction Procedures:
  - Integrity Verification Procedures:

# Constrained Data Items:

- Boot loader
- Operating System and Trusted Applications
- Voting Application
- Ballot Definition
- Vote Tally
- Completed Ballot

# Integrity constraints:

- New images of the boot loader, OS, Trusted Applications, and Voting Applications must include a certificate of origin signed by a trusted party. The certificate must include a message digest of the image.
- The OS, Trusted Applications, and Voting Applications must pass an integrity check based on their certificate of origin before being executed.
- The Ballot Definition must be signed digitally by an election official distinct from the official operating the voting machine.

# Transaction processes (TPs):

- Update Boot Loader
- Update OS and Trusted Applications
- Update Voting Application
- Define Ballot
- Start Election
- End Election
- Vote

# Comparison to Biba

- Biba
  - No notion of certification rules; trusted subjects ensure actions obey rules
  - Untrusted data examined before being made trusted
- Clark-Wilson
  - Explicit requirements that *actions* must meet
  - Trusted entity must certify *method* to upgrade untrusted data (and not certify the data itself)

# Key Points

- Integrity policies deal with trust
  - As trust is hard to quantify, these policies are hard to evaluate completely
  - Look for assumptions and trusted users to find possible weak points in their implementation
- Biba based on multilevel integrity
- Clark-Wilson focuses on separation of duty and transactions

# Hybrid Policies

- Policy models in specific domains
- Combine notions of confidentiality and integrity
- Two case studies:
  - Chinese Wall, Brewer and Nash
  - British Medical Association (BMA) model, Ross Anderson, 1996

# Chinese Wall

- Domain:
  - Financial institutions
- Problem:
  - Want to enable sharing of sensitive information between traded companies and investment banks
  - Don’t want the investment banks to become a conduit of information
  - British securities law dictates that strict conflict of interest rules be applied preventing one specialist to work with two clients in the same sector

# Example

- Oil Companies
  - Shell
  - Texaco
  - Mobil
- Soft Drink
  - Pepsi
  - Coke

- Analysts
  - Amy
  - Bob
- Problem
  - Amy is working on Shell and Pepsi
    - Amy cannot work on T, M or C
  - Bob starts working on Coke
    - Can Bob help Amy on Shell?

# Novel Aspect

- Model is temporal --- it changes with time
- Before Bob starts working on Coke he can work on anything
- Once he commits to Coke he is directly blocked from working on Pepsi

# Concepts

- Objects:  information related to a client company
- Company Dataset (CD):  objects related to a single company
- Conflict of Interest (COI) class:  datasets of the companies in competition
- Sanitized:  Non confidential information about a company

# Rules

- Simple Security:
  - S can read O if one of:
    - S has accessed O’ and CD(O) = CD(O’)
    - For all previously accessed O’, COI (O’) ≠ COI (O)
    - O is sanitized
- *-Property
  - S can write O iff
    - S can read O by above
    - For all unsanitized O’ readable by S, CD (O’) = CD (O)

# Comments

- *-Property is very strict (too strict?)
- How does this relate to BLP?

# BMA Model

- Presentation follows Anderson Section 8.2.3
- BMA model influenced HIPAA

# BMA Model

- Typical attack
  - Hello, this is Dr. B of the cardiology department at …. Your patient S has just been admitted here in a coma, and he has a funny-looking ventricular arrhythmia. Can you tell me if there's anything relevant in his record?
- At time of study (1997) 15% of queries were bogus

# Past Approaches

- Adapt Military policy
  - Secret: AIDS, STDS,
  - Confidential: “normal” patient records
  - Restricted: admin and prescription data
- Problem:
  - What about a prescription for AZT?

# BMA Model Goals

- "… enforce principle of patient consent, prevent too many people getting access to too large databases of identifiable records. … not anything new … codify best practices"

# BMA Principles

- Access Control
  - each identifiable clinical record shall be marked with an access control list naming the people or groups of people who may read it and append data to it. The system shall prevent anyone not on the ACL from accessing the record in any way
- Record Opening
  - a clinician may open a record with herself and the patient on the ACL. Where a patient has been referred, she may open a record with herself, the patient, and the referring clinician(s) on the ACL

# BMA Principles (cont)

- Control:
  - One of the clinicians on the ACL must be marked as being responsible. Only she may alter the ACL, and she may only add other health care professionals to it
- Consent and notification:
  - the responsible clinician must notify the patient of the names on his record’s ACL when it is opened, of all subsequent additions, and whenever responsibility is transferred. His consent must also be obtained, except in emergency or in the case of statutory exemptions

# BMA Principles (cont)

- Persistence:
  - No one shall have the ability to delete clinical information until the appropriate time period has expired
- Attribution:
  - all accesses to clinical records shall be marked on the creord with the subject’s name, as well as the date and time.  An audit trail must also be kept of all deletions

# BMA

- Information flow:
  - Information derived from record A may be appended to record B if and only if B’s ACL is contained in A’s
- Aggregation control:
  - There shall be effective measures to prevent the aggregation of personal health information. In particular, patients must receive special notification if any person whom it is proposed to add to their access control list already has access to personal health information on a large number of people

# BMA

- Trusted Computing Base
  - computer systems that handle personal health information shall have a subsystem that enforces the above principles in an effective way.  Its effectiveness shall be subject to evaluation by independent experts.

# Contrasts

- BMA is decentralized
- Chinese Wall is centralized
- Both hybrid models reflect concerns not naturally provided by BLP alone

# Up Next

- Readings on Telephone Fraud detection
  - Gary M. Weiss (2005). Data Mining in Telecommunications. http://storm.cis.fordham.edu/~gweiss/papers/kluwer04-telecom.pdf
  - Corinna Cortes, Daryl Pregibon and Chris Volinsky, "Communities of Interest", http://homepage.mac.com/corinnacortes/papers/portugal.ps
  - NY Times article on NSA spying, Dec 2005, http://www.commondreams.org/headlines05/1216-01.htm
  - USA Today article on NSA phone records, May 2006, http://www.usatoday.com/news/washington/2006-05-10-nsa_x.htm
- Bishop Chapter 13
  Anderson 17 and 21