

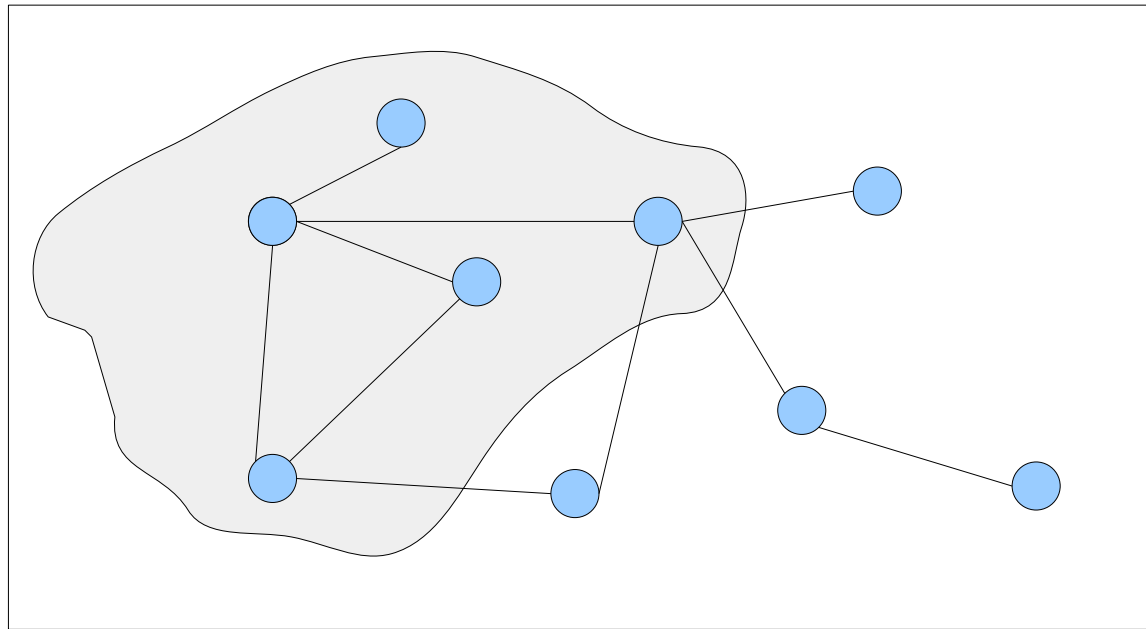


# Criptografía y Seguridad

Políticas de  
seguridad

# Política de seguridad

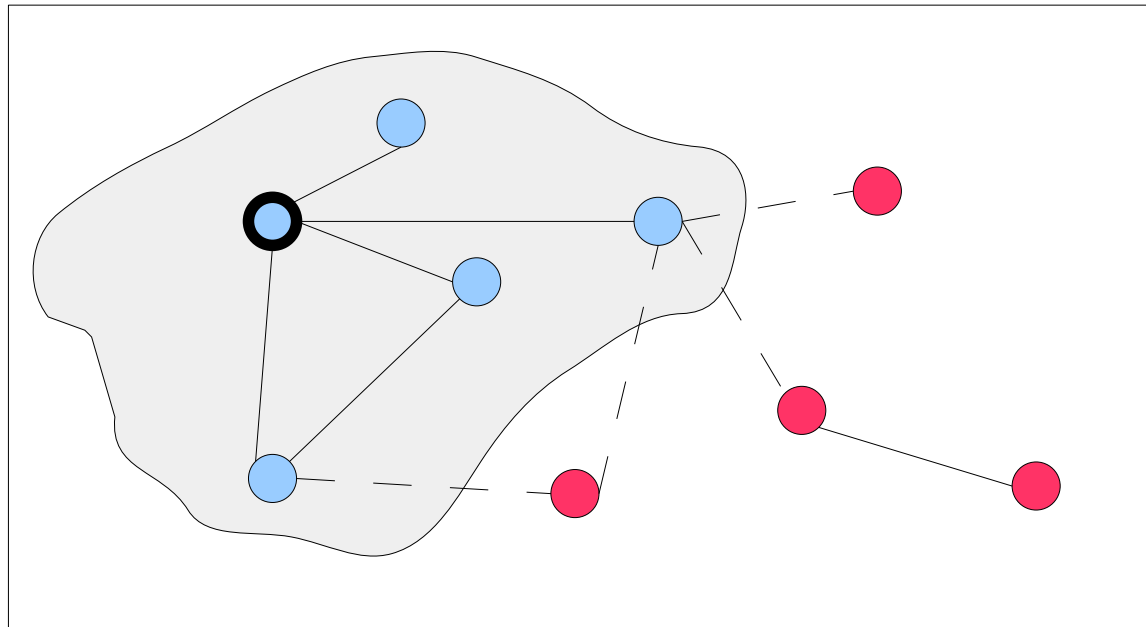
- Es un enunciado que parte los estados de un sistema en autorizados (o seguros), y no autorizados.



- Si el sistema entra en un estado no autorizado ocurrió una violación de seguridad

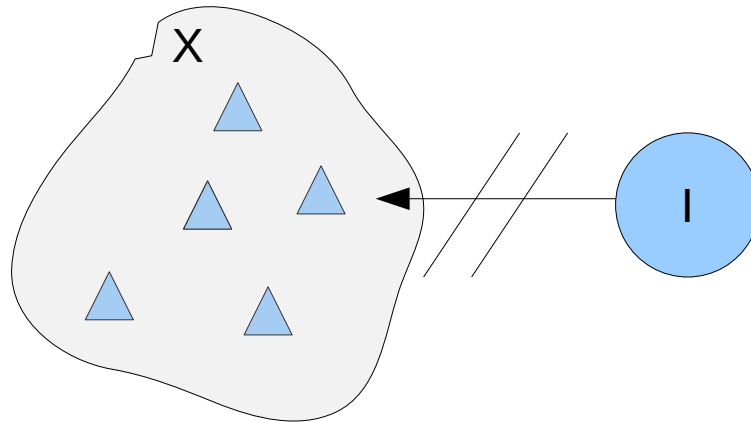
# Sistema seguro

- Es un sistema que comienza en un estado autorizado y no puede entrar en un estado no autorizado



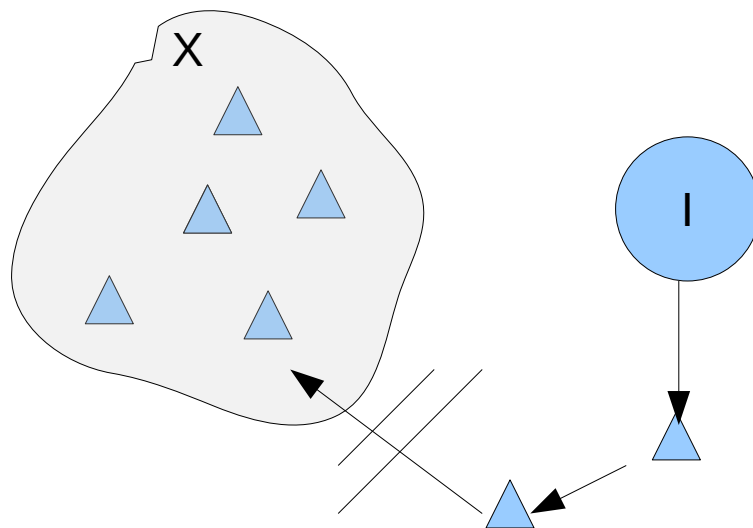
# Confidencialidad

- Sean
  - $X$  = conjunto de entidades
  - $I$  = Información
- Propiedad de confidencialidad
  - $I$  es confidencial para  $X$  si ningún miembro de  $X$  puede obtener información de  $I$



# Confidencialidad (2)

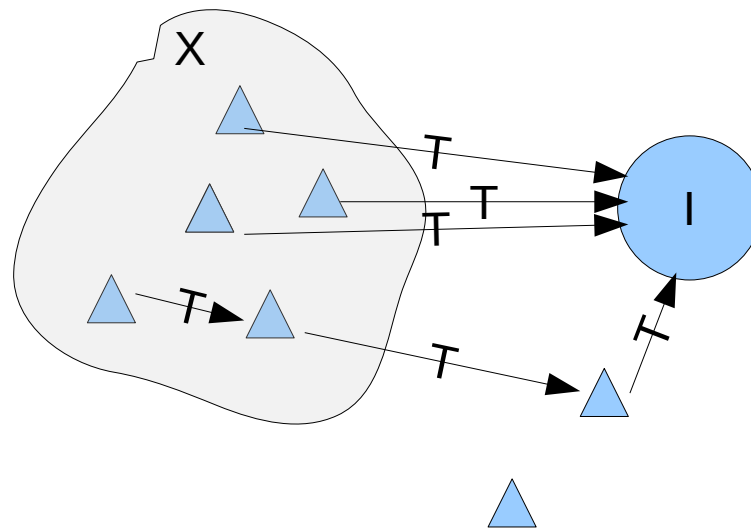
- Sean
  - $X$  = conjunto de entidades
  - $I$  = Información
- Propiedad de confidencialidad
  - $I$  es confidencial para  $X$  si ningún miembro de  $X$  puede obtener información de  $I$



¡Ni siquiera por vías Indirectas!

# Integridad

- Sean
  - $X$  = conjunto de entidades
  - $I$  = Información o recurso
- Propiedad de integridad
  - $I$  es íntegro para  $X$  si todo miembro de  $X$  confía en  $I$

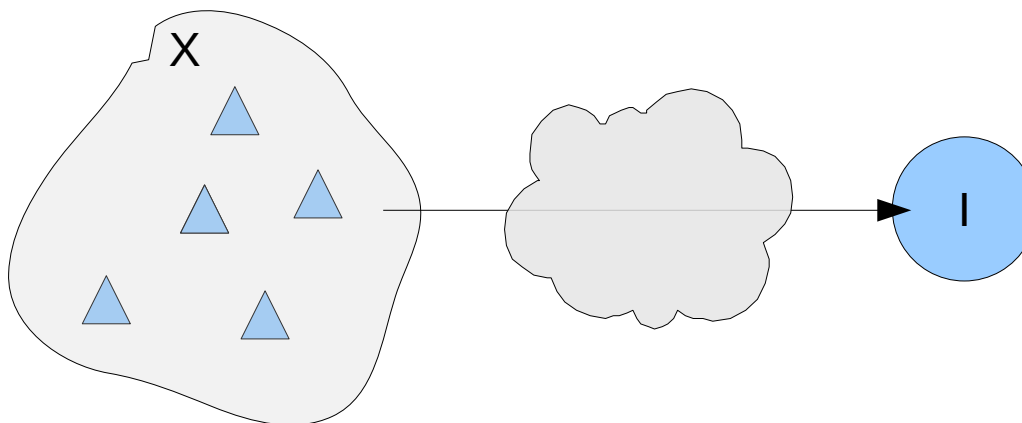


# Tipos de integridad

- Integridad de datos: Confianza en transporte y almacenamiento
- Integridad de origen: Confianza en el origen del dato o la identidad que representa
- Garantía: confianza en que el recurso o programa funciona como debería

# Disponibilidad

- Sean
  - $X$  = conjunto de entidades
  - $I$  = Recurso
- Propiedad de disponibilidad
  - $I$  está disponible para  $X$  si todo miembro de  $X$  puede acceder a  $I$  cuando lo requiera





# Paradigmas de Control de Acceso

- Las políticas se centran en controlar el acceso a objetos:
- Acceso Discrecional (DAC)
  - Reglas arbitrarias (adhoc)
  - Mecanismos puntuales
  - Opcional: Acceso controlado por creadores
    - Quien crea la información controla el acceso a la misma
- Acceso Mandatorio (MAC)
  - Reglas prefijadas
  - Mecanismos del sistema
  - No pueden ser alterados

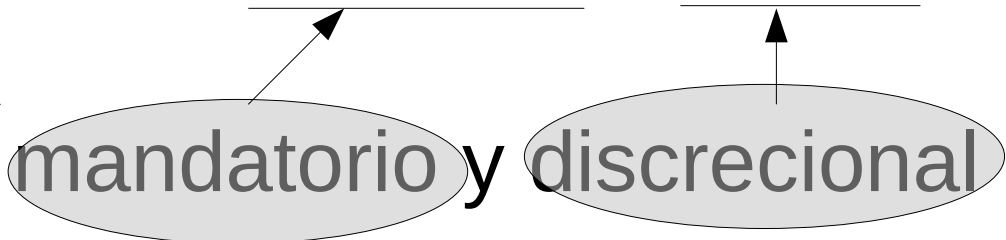
# Modelos

- Describen familias de políticas
- Proveen un marco teórico común
  - Permiten reutilizar demostraciones
  - Simplifican el desarrollo de políticas

# Modelo Bell-LaPadula

- Es un modelo de política militar
  - Se centra en garantizar confidencialidad
- Conceptualización del sistema:
  - Dividir el sistema en Sujetos y Objetos
  - Transición: (Sujeto, Objeto, Acción)
  - Las acciones se clasifican en Lectura y Escritura
- El modelo (versión simplificada):
  - Una lista ordenada de Niveles
  - Cada objeto y sujeto tienen un nivel asignado
  - ¿Cómo se restringe el acceso?

# Modelo Bell-LaPadula - Lectura

- La información fluye hacia arriba, no hacia abajo
  - Se permite leer información de menor nivel
  - Se prohíbe leer información de mayor nivel
- Condición de seguridad simple (reducida)
  - S puede leer O si y solo si  $L(O) \leq L(S)$  y S tiene permiso para leer O
- Se combina acceso  mandatorio y discrecional

**Importante:** Los accesos discrecionales solo pueden restringir a los mandatorios, no contradecirlos

# Modelo Bell-LaPadula - Escritura

- La información fluye hacia arriba, no hacia abajo
  - Se permite escribir información de mayor nivel
  - Se prohíbe escribir información de menor nivel
- Condición de cierre (\* property - reducida)
  - S puede escribir O si y solo si  $L(S) \leq L(O)$  y S tiene permiso para escribir O

# Modelo Bell-LaPadula - DAC

- El acceso discrecional se define con una matriz de acceso:

	O1	...	On
S1	R, W		R
...			
Sn	W		

Importante: La matriz de acceso RESTRINGE los accesos dados por el modelo mandatorio

# Modelo Bell-LaPadula

- El modelo original define 4 niveles
  - Top Secret (TS)
  - Secret (S)
  - Confidential (C)
  - Public (P)
- Pero funciona para cualquier cantidad de niveles

# Modelo Bell-LaPadula - Ejemplo

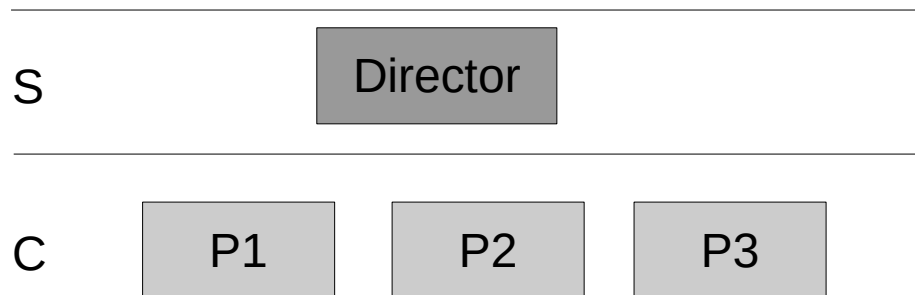
- Ejemplo
  - Sujetos: Diseñador, Gerente, Director
  - Objetos: Producto X, Balances
- Etiquetado:
  - $L(\text{Diseñador}) = \text{Confidential}$
  - $L(\text{Gerente}) = \text{Secret}$
  - $L(\text{Director}) = \text{Top Secret}$
  - $L(\text{Producto X}) = \text{Confidential}$
  - $L(\text{Balances}) = \text{Secret}$

¿Qué acciones están permitidas?



# Modelo Bell-LaPadula

- ¿Que ocurre si no existe un orden completo?
- Ejemplo:
  - Varios proyectos aislados
  - No se ven entre sí
  - Un director ve todos los proyectos



No se puede  
GARANTIZAR la aislacion  
entre Px y Py!

# Modelo Bell-LaPadula

- Modelo completo
  - Además de niveles se definen categorías
  - Describen el tipo de información
  - Las categorías no están ordenadas
  - A cada objeto y sujeto se le asigna un compartimento
  - Compartimento = (Nivel, {categorías})

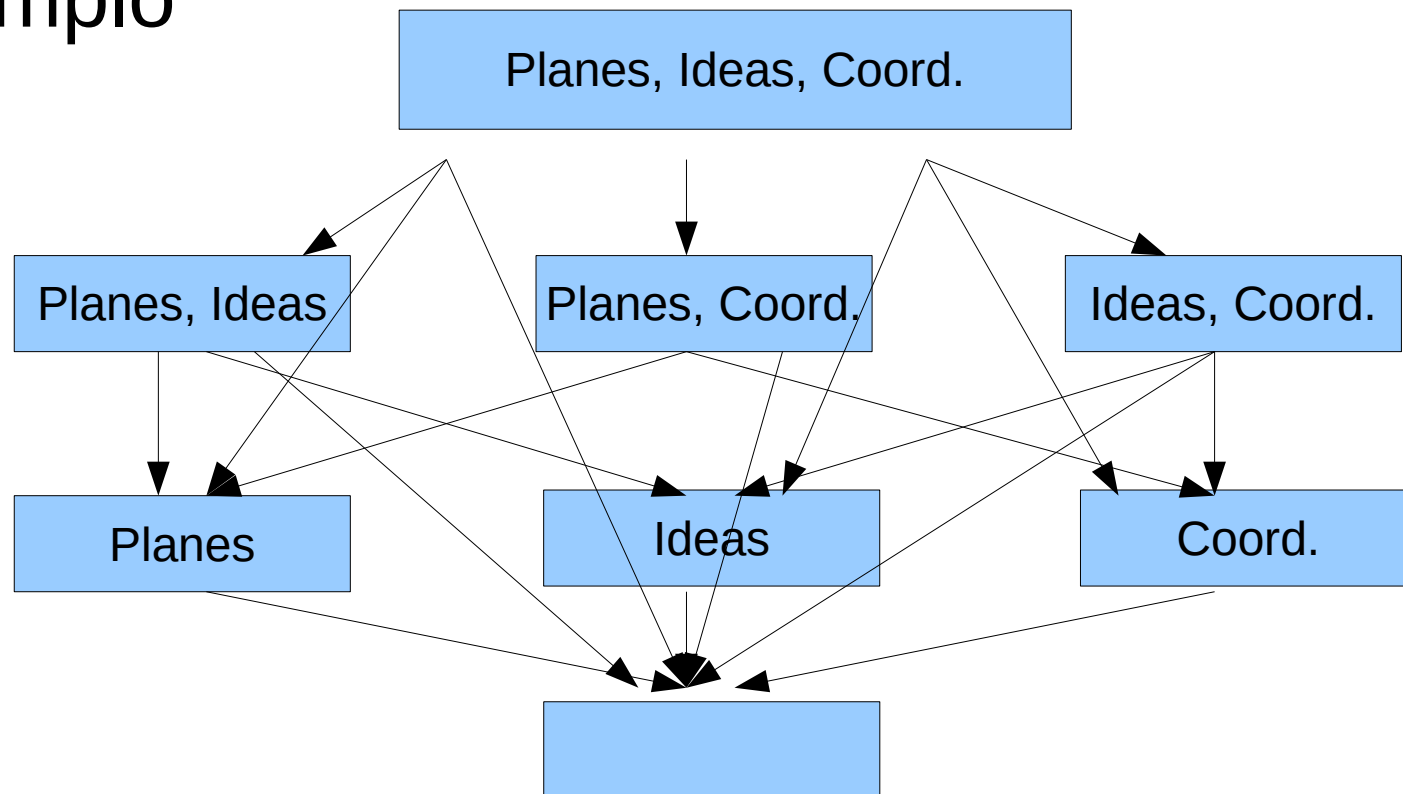
# Modelo Bell-LaPadula

- Dominancia:

*Sea ,  $l \in L, C \subseteq CAT$*

$$(l, C) \text{ dom } (l', C') \Leftrightarrow l' \leq l \wedge C' \subseteq C$$

- Ejemplo



# Modelo Bell-LaPadula

- Compartimentos

- Conjunto parcialmente ordenado
- Dado A y B puede ocurrir que:
  - $A \text{ dom } B$
  - $B \text{ dom } A$
  - Ni  $A \text{ dom } B$  ni  $B \text{ dom } A$
- Ejemplos:
  - $(TS, \{X\}) \text{ dom } (TS, \{\})$
  - $(TS, \{X\}) \text{ dom } (S, \{X\})$
  - $(S, \{X\}) \text{ no dom } (P, \{Y\})$
  - $(P, \{Y\}) \text{ no dom } (S, \{X\})$

# Modelo Bell-LaPadula - Lectura

- La información fluye hacia arriba, no hacia abajo
  - Se permite leer información de menor nivel
  - Se prohíbe leer información de mayor nivel
- Condición de seguridad simple
  - S puede leer O si y solo si  $L(S) \text{ dom } L(O)$  y S tiene permiso para leer O
- Se combina acceso mandatorio y discrecional

# Modelo Bell-LaPadula - Escritura

- La información fluye hacia arriba, no hacia abajo
  - Se permite escribir información de mayor nivel
  - Se prohíbe escribir información de menor nivel
- Condición de cierre
  - S puede escribir O si y solo si  $L(O) \text{ dom } L(S)$  y S tiene permiso para escribir O

# Modelo Bell-LaPadula

- Teorema básico de la seguridad
  - Si un sistema comienza en un estado seguro y sus transiciones satisfacen la condición simple de seguridad y la condición de cierre, todos los estados del sistema son seguros
- Garantiza que no existe flujo de información en el sentido de la condición simple
- ¿Para que existe la condición de cierre?
  - Para garantizar que no existan caminos indirectos que violen la condición simple

# El problema de la comunicación

- A le envía un mensaje a B ( $B \text{ dom } A$ )
  - Como  $B \text{ dom } A$ , la regla de cierre permite enviar el mensaje (escritura)
- B le contesta
  - Como  $B \text{ dom } A$ , la regla de cierre prohíbe enviar el mensaje
  - Problemas!!!
- El modelo Bell-LaPadula establece la posibilidad de disminuir el nivel de acceso temporalmente
  - $\text{MaxLevel} \text{ y } \text{CurLevel} / \text{MaxLevel} \text{ dom } \text{CurLevel}$
  - La disminución de acceso debe ser solicitada explícitamente



# Principio de tranquilidad

- Usuarios y objetos no cambian sus niveles luego de ser creados
- ¿Que pasaría se los niveles cambiasen?
- Subir el nivel de un objeto
  - La información fue leída por usuarios de menor nivel
  - Viola principio de seguridad simple
- Bajar el nivel de un objeto
  - Problema de desclasificación
  - Viola principio de cierre

# Políticas de integridad

- Se concentran en preservar la integridad
- Mayor uso en ambientes comerciales
- Requerimientos muy diferentes a las políticas de confidencialidad
  - Prevenir modificación de datos por entidades no autorizadas
  - Prevenir modificaciones no autorizadas de datos por entidades autorizadas
  - Asegurar que los datos representan la información que se supone deben representar

# Modelos de integridad de Biba

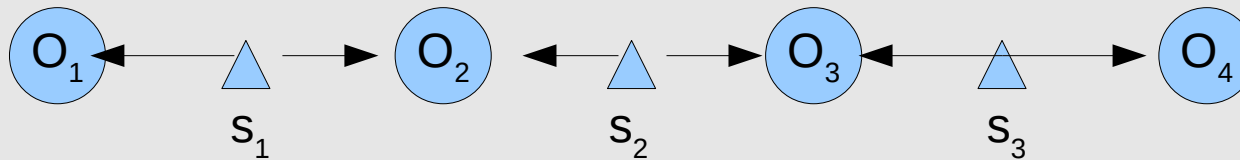
- Base para los tres modelos:
  - Conjunto de sujetos  $S$ , objetos  $O$ , Niveles de Integridad  $I$
  - Relación  $< : I \times I$ , dominancia del 1ro sobre el 2do
  - $i: S \cup O \rightarrow I$ , *nivel de integridad de una entidad*
  - $\underline{r}: S \times O$ , *pares  $s \in S, o \in O$  donde se puede leer  $o$*
  - $\underline{w}: S \times O$ , *ídem para escritura*
  - $x: S \times O$ , *ídem para ejecución*

# Niveles de integridad

- A mayor nivel, mayor confianza de que
  - Un programa se ejecutará correctamente o detectará errores en sus entradas
  - Un dato es preciso y/o fiable
- Clasificación normal
  - No confiable                      Untrusted
  - Ligeramente confiable              Slightly trusted
  - Confiable                      Trusted
  - Altamente confiable              Highly trusted
  - Intachable                      Unimpeachable

# Camino de transferencia de información

- Camino que puede seguir la información para llegar desde un objeto a otro
- Secuencia de objetos  $o_1, \dots, o_{n+1}$  y sujetos  $s_1, \dots, s_n$  tal que  $s_i \underline{r} o_i$  y  $s_i \underline{w} o_{i+1}$  para todo  $i, 1 \leq i \leq n$ .



¿Como fluye información de  $O_1$  a  $O_4$ ?

$S_1$  lee  $O_1$  y escribe  $O_2$

$S_2$  lee  $O_2$  y escribe  $O_3$

$S_3$  lee  $O_3$  y escribe  $O_4$

# Low-Water-Mark Policy (1er modelo)

- Idea: Si un sujeto usa información poco confiable, se vuelve poco confiable
- Reglas:
  1.  $s \in S$  puede escribir  $o \in O$  si y solo si  $i(o) \leq i(s)$ .
  2. Si  $s \in S$  lee  $o \in O$ , entonces  $i'(s) = \min(i(s), i(o))$  es el nuevo nivel de integridad de  $s$
  3.  $s_1 \in S$  puede ejecutar  $s_2 \in S$  si y solo si  $i(s_2) \leq i(s_1)$
- Previene contra:
  - Modificaciones directas que bajarían el nivel de integridad
  - Modificaciones indirectas con información de menor nivel de integridad

# Restricción en el flujo de información

- Si hay un camino de transferencia entre  $o_1$  y  $o_n$  la aplicación de la política requiere  $i(o_j) \leq i(o_1)$  para todo  $1 < j \leq n$
- Demostración (idea):
  - Asumir que existe un camino de transferencia y las operaciones se realizan ordenadamente ( $s_1$  lee  $o_1$ ,  $s_1$  escribe  $o_2$ ,  $s_2$  lee  $o_1$  ...)
  - Por inducción  $i(s_1) = \min(i(s_1), i(o_1))$ . Luego de  $k$  lecturas  $i(s_k) = \min(i(o_1), i(o_2), \dots, i(o_k))$
  - La última escritura requiere  $i(o_n) \leq i(s_n) \leq i(o_1)$

# Low-Water-Mark - Problemas

- Los niveles de integridad de los sujetos decaen con el uso del sistema
  - Eventualmente nadie puede acceder o generar objetos de niveles altos de integridad
- Alternativas: modificar los niveles de integridad de los objetos en lugar de los sujetos
  - Problema similar: los objetos se degradan hasta llegar a los niveles mas bajos de integridad



# Ring Policy (2do modelo)

- Considera solamente el problema de la modificación directa de objetos
- Reglas:
  1.  $s \in S$  puede escribir  $o \in O$  si y solo si  $i(o) \leq i(s)$ .
  2. **Cualquier sujeto puede leer cualquier objeto**
  3.  $s_1 \in S$  puede ejecutar  $s_2 \in S$  si y solo si  $i(s_2) \leq i(s_1)$
- Los niveles de integridad son estaticos
- Previene contra la modificación directa
- Permite utilizar información de menor nivel de confianza para generar información de mayor nivel

# Strict Integrity (3er modelo)

- Similar al modelo Bell-LaPadula
- Reglas:
  1.  $s \in S$  puede leer  $o \in O$  sii  $i(s) \leq i(o)$
  2.  $s \in S$  puede escribir  $o \in O$  sii  $i(o) \leq i(s)$
  3.  $s_1 \in S$  puede ejecutar  $s_2 \in S$  sii  $i(s_2) \leq i(s_1)$
- Se pueden agregar categorías y controles discrecionales para obtener el dual de Bell-LaPadula
- Mantiene la misma restricción en el flujo de información

# Modelo de la pared China

- Modelo híbrido
  - Toma en cuenta confidencialidad e integridad
- Se concentra en el problema de conflictos de interés
  - De amplio uso en ámbitos bursátiles y judiciales
  - Algunos países exigen medidas que prevengan problemas de conflicto de interés
- Ejemplos:
  - Impedir que un trader represente a dos clientes que compiten en el mercado
  - Impedir que un abogado represente a dos empresas competidoras

# Concepto

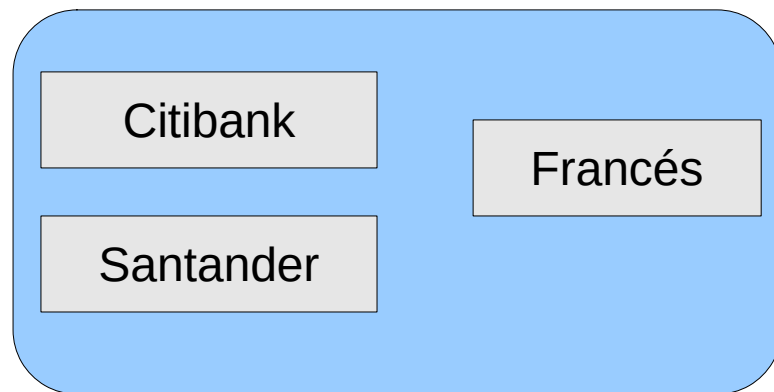
- Agrupar las entidades en clases de conflicto de interés
- Controlar el acceso de sujetos a a cada clase
- Controlar la escritura a todas las clases para impedir que se mueva información en contra de la política
- Permitir que datos desclasificados sean vistos por todos

# Definiciones

- Objetos. Items de información
- Company dataset (CD): conjunto de objetos relacionados con la misma empresa o grupo
- Conflict of interest Class (COI): contiene CDs de empresas con conflicto de intereses
  - Se asume que cada objeto pertenece a exactamente un CD y a un COI

# Ejemplo

COI de entidades financieras



COI de medios de prensa



# Idea

- Un usuario nuevo puede leer cualquier objeto
  - Una vez que leyo un objeto no puede acceder a objetos de otras empresas que entren en conflicto.
- Por ejemplo, un asesor puede leer objetos pertenecientes al CD La Nación
  - Pero una vez hecho esto no puede leer objetos del CD Clarin
  - Aunque nada impide que lea objetos del CD Santander

# Elemento temporal

- Si S lee cualquier CD de un COI, no puede volver a leer otra CD del mismo COI, *nunca*.
  - El acceso depende de la historia de S
  - Se impide que use información que obtuvo antes para tomar decisiones que afecten a intereses en competencia
- El elemento temporal es un nuevo requerimiento que no es capturado en el modelo Bell-LaPadula



# CW–Condición Simple de Seguridad

- s puede leer o si alguna de las condiciones se cumple:
  - Existe un  $o'$  leído previamente tal que  $CD(o) = CD(o')$ 
    - Se accedió previamente a un dato de la empresa
  - Para todo  $o'$  leído previamente,  $COI(o) \neq COI(o')$ 
    - No se accedió a algún dato en una categoría de conflicto de interés
  - o es un objeto público (desclasificado)
    - Información que dejó de ser confidencial (por ejemplo: balance anual del período anterior)
    - Información sanitizada (donde se eliminan partes confidenciales)

# Escritura

Control mas complicado para prevenir flujos indirectos

Considerar:

- $s_1$  accede a objetos de Clarín y Santander
- $s_2$  accede a objetos de La Nación y Santander
- Ninguno esta en una situación de conflicto de interés.

Pero si  $s_1$  escribiera un objeto en DS(Santander), podría estar volcando información de Clarín, y de esa forma hacer que  $s_2$  la tenga disponible.

# CW- Propiedad de cierre

- S puede escribir o si se cumplen las siguientes condiciones:
  - S puede leer o según la condición simple de seguridad
  - Para todo objeto no público  $o'$ , si  $s$  puede leer  $o'$  entonces  $CD(o') = CD(o)$
- Significa que para escribir un objeto es necesario que todo objeto accesible para la entidad pertenezca al mismo grupo
  - Por ejemplo: el dato es escrito por un miembro de la empresa

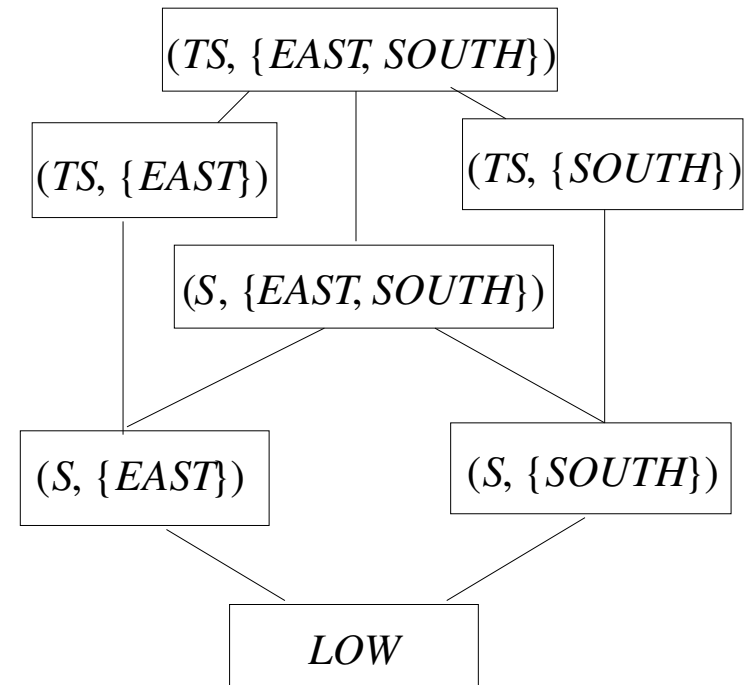
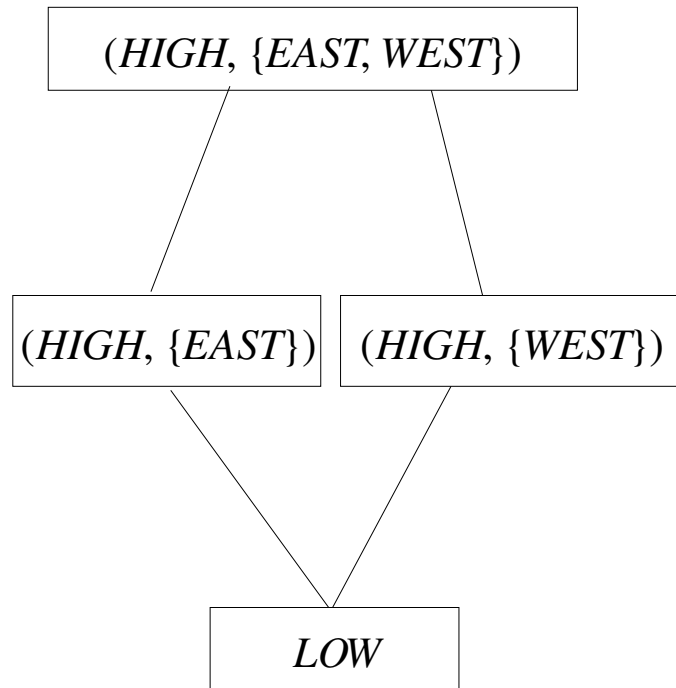
# Composición de políticas

- El problema:
  - Conectar dos sistemas seguros
- Las preguntas
  - ¿La composición de los dos sistemas será segura?
  - ¿Se puede crear una política única consistente con ambas?

# Composición de políticas

- Supongamos dos sistemas que siguen un modelo del tipo Bell-LaPadula
- ¿Cual es el modelo compuesto del sistema conjunto?
- Problemas
  - Los grupos no tienen orden total
  - Es necesario establecer una correspondencia entre niveles de seguridad

# Ejemplo



# Analisis del ejemplo

- Determinar orden de los niveles
  - Por ejemplo,  $S < HIGH < TS$
- Determinar equivalencia de categorías
  - Por ejemplo: east representa lo mismo
- El modelo complementario tendría:
  - 4 niveles ( $LOW < S < HIGH < TS$ )
  - 3 categorías (SOUTH, EAST, WEST)
  - Notar que es una nueva política

# Composición

- Si los modelos de políticas son iguales
  - Si se puede cambiar la política de los componentes (reemplazarla por el modelo compuesto) la composición es trivial
  - Si no se puede, hay que demostrar que la composición cubre los requerimientos de las políticas de los componentes. Muy difícil



# Composición

- Si los modelos de políticas son diferentes
  - ¿Que significa seguro en este contexto?
  - ¿Que política domina la composición?
- No hay una única solución
- Posibles principios guía:
  - Cualquier acceso permitido por la política de seguridad de un componente debe estar permitido por la política emergente (autonomía)
  - Cualquier acceso prohibido por la política de seguridad de un componente debe ser prohibido por la política emergente (seguridad)

# Consecuencias

- La política compuesta satisface la seguridad de las políticas de los componentes
- ¿Si algún caso no esta explícitamente permitido o prohibido por alguna política?
  - Permitirlo (modelo original de Gong & Quiam)
  - Prohibirlo (principio de denegación por defecto)

# Ejemplo

- Sistema X: Bob no puede leer archivos de Alice
- Sistema Y: Eve y Lilith pueden leer los archivos del otro
- Composición:
  - Por el sistema Y: bob podría leer archivos de Alice.
  - Pero el sistema X prohíbe explícitamente esto
  - Metodología:
    - Crear el conjunto de accesos posibles (expansión de relaciones transitivas)
    - Quitar las relaciones no permitidas
    - Determinar el número de relaciones que deben ser quitadas es un problema NP

# Lectura recomendada

Capítulo 4  
Capítulo 5: 5.1-5.4  
Capítulo 6: 6.1-6.2  
Capítulo 7: 7.1  
Capítulo 8: 8.1

Computer Security Art and Science  
Matt Bishop