Лекция 5

Модели мандатного (полномочного) доступа

или модели контроля информационных потоков.

Основные положения моделей мандатного доступа

Основаны:

- на субъектно-объектной модели КС
- на правилах организации секретного делопроизводства

принятых в государственных (военных) учреждениях многих стран

Multi Level Security – MLS.

В моделях мандатного доступа устанавливается жесткое управление доступом с целью **контроля не столько операций, а потоков** между сущностям с разным уровнем безопасности.

Для управления (разграничения) доступом к объектам **одного уровня конфиденциальности используют дискреционный принцип**, т.е. дополнительно вводят матрицу доступа.

Основные положения моделей мандатного доступа

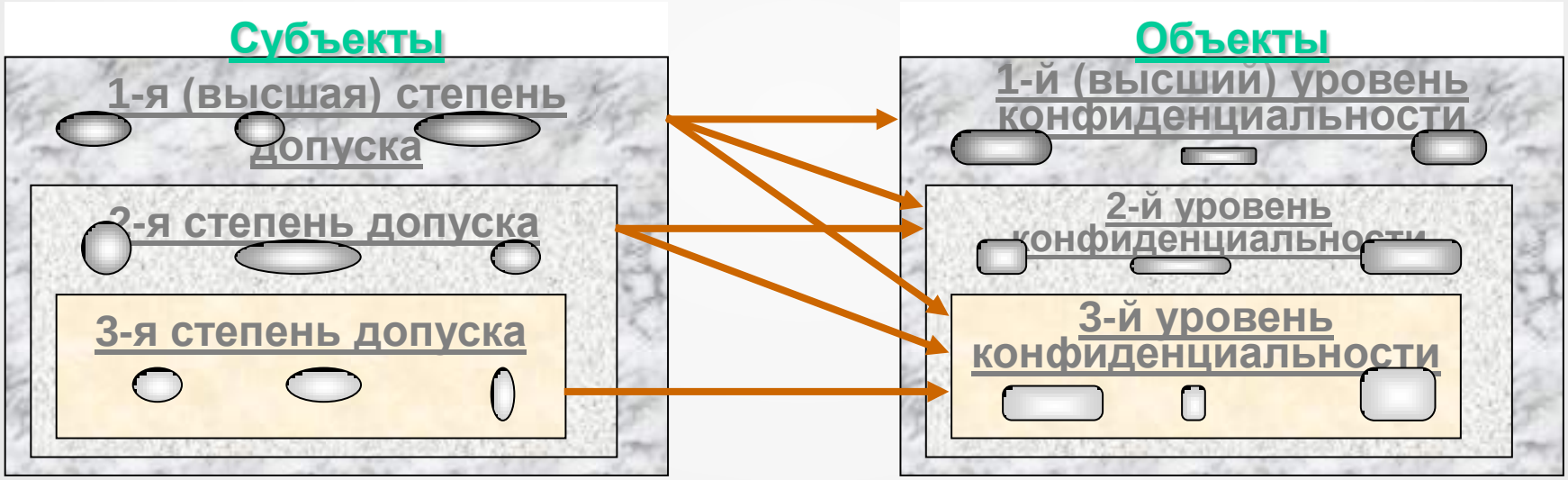
Вводится система "*уровней безопасности*" – решетка с оператором доминирования

Устанавливается функция (процедура) присваивания субъектам и объектам уровней безопасности

Управление и контроль доступом субъектов к объектам производится на основе двух правил.

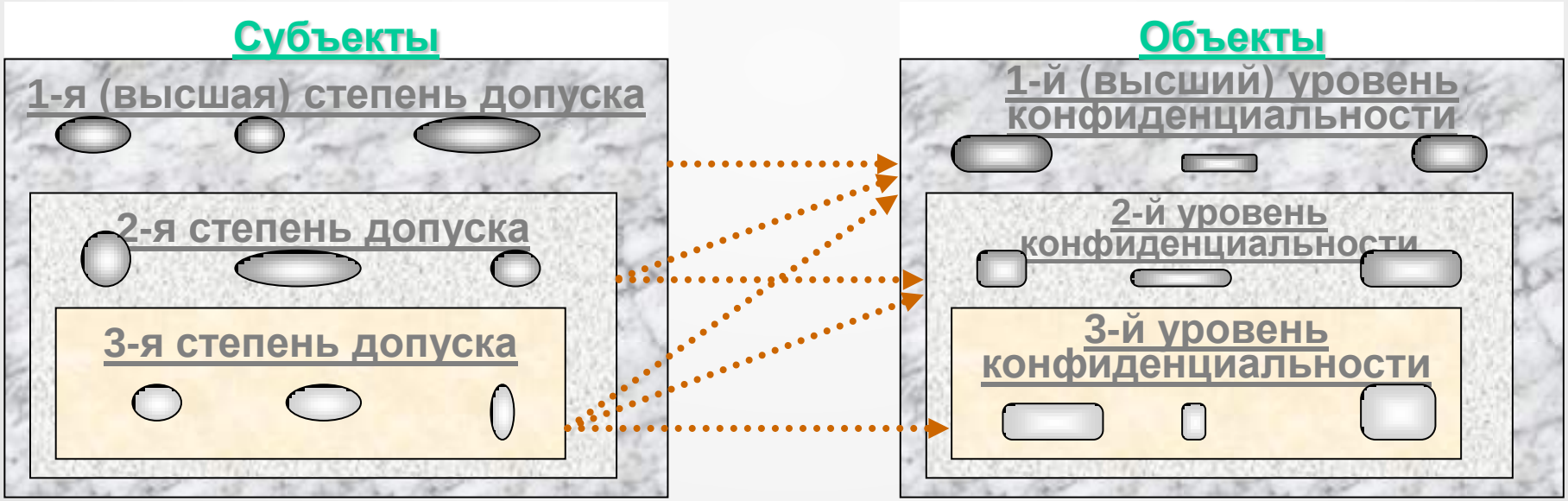
Основные положения моделей мандатного доступа

Запрет чтения вверх (*no read up - NRU*) - субъект не может читать объект с уровнем безопасности, большим своего уровня безопасности

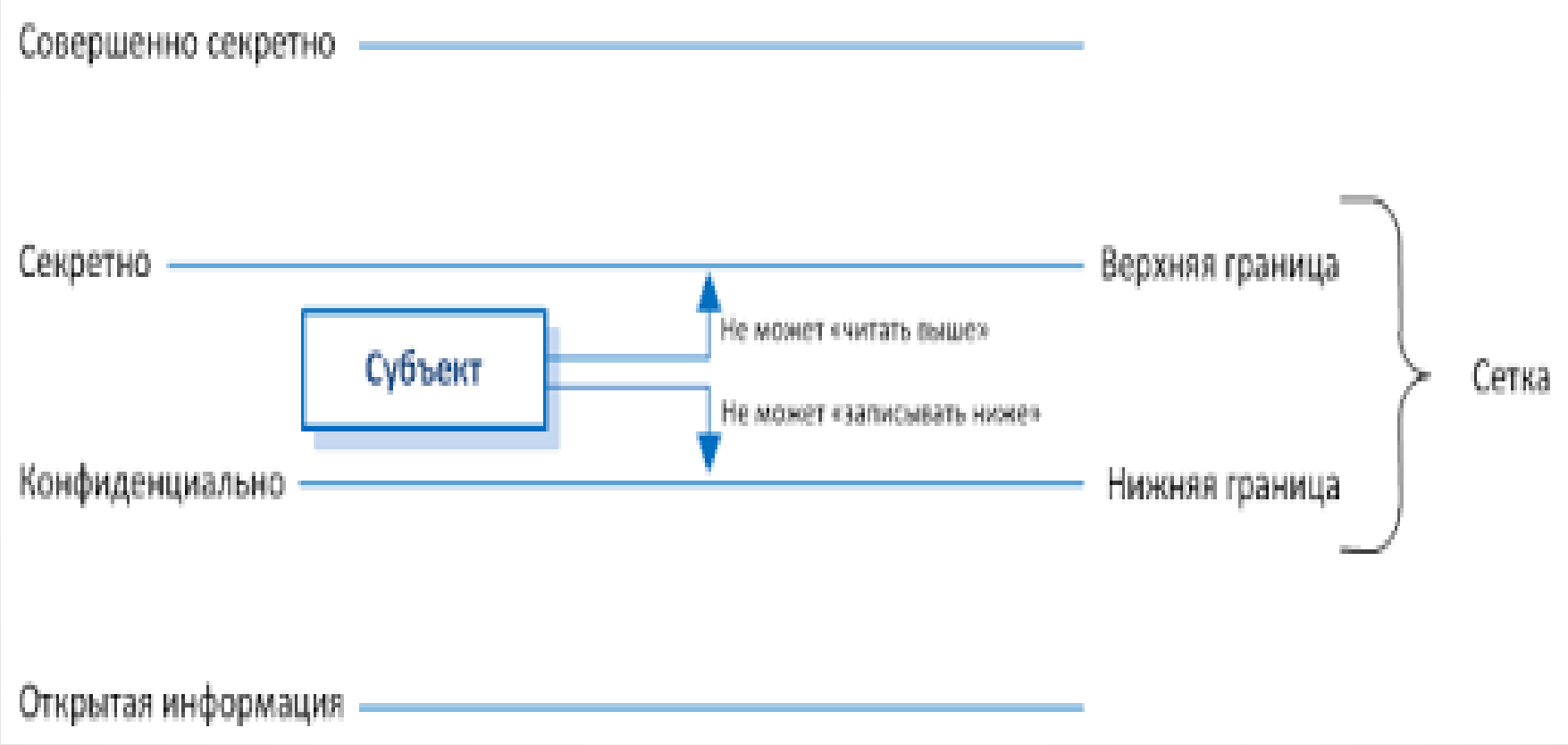


Основные положения моделей мандатного доступа

Запрет записи вниз (*no write down - NWD*) - субъект не может писать информацию в объект, уровень безопасности которого ниже уровня безопасности самого субъекта



Основные положения моделей мандатного доступа



Модель АДЕПТ–50

Модель Адепт-50 — одна из первых моделей безопасности, которая рассматривает только 4 группы объектов безопасности: пользователи, задания, терминалы и файлы. Каждый объект безопасности описывается вектором (A, C, F, M), включающим следующие параметры безопасности.

Компетенция A — элемент из набора упорядоченных универсальных положений о безопасности, включающих априорно заданные возможные в ИС характеристики объекта безопасности, например, категория конфиденциальности объекта: несекретно, конфиденциально, секретно, совершенно секретно.

Категория C — рубрикатор (тематическая классификация). Рубрики не зависят от уровня компетенции. Пример набора рубрик: финансовый, политический, банковский.

Полномочия F — перечень пользователей, имеющих право на доступ к данному объекту.

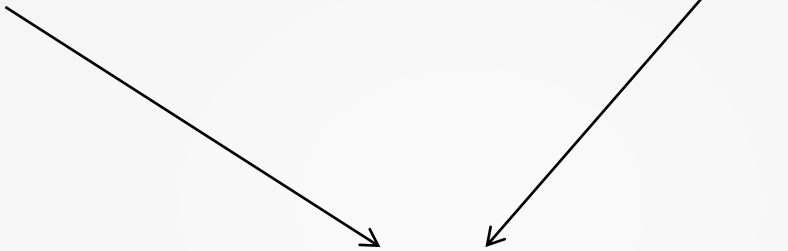
Режим M — набор видов доступа, разрешенных для определенного объекта или осуществляемых объектом. Пример: читать данные, записывать данные, исполнить программу (исполнение программ понимается как порождение активной компоненты из некоторого объекта — как правило, исполняемого файла).

Четырехмерный вектор, полученный на основе прав задания (субъекта), а не прав пользователя, используется в модели для управления доступом. Такой подход обеспечивает контроль права на доступ над множеством программ и данных, файлов, пользователей и терминалов. Например, наивысшим полномочием доступа к файлу для пользователя «секретно», выполняющего задание с «конфиденциального» терминала будет «конфиденциально».

Модель решётки ценностей

несекретно,
конфиденциально,
секретно,
совершенно секретно.

финансовый, политический, банковский



(секретно, {финансовый, банковский})

Т.е. создается новое множество упорядоченных пар
Lx**X**
из частично упорядоченных множеств:
(**L**, ≤) и (**X**, ⊆):

If a ≤ b из L and A ⊆ B из X then (a,A) $\underline{\leq}$ (b,B)

Решетка уровней безопасности Λ_L

Λ_L - алгебра (L, ≤, •, ⊗), где

L – базовое множество уровней безопасности

≤ – оператор доминирования, определяющий частичное нестрогое отношение порядка на множестве L.

Отношение, задаваемое ≤ , рефлексивно, антисимметрично и транзитивно:

$$\forall\, l \in L: l \le l\ ;$$
$$\forall\, l_1, l_2 \in L: (l_1 \le l_2 \ \wedge l_2 \le l_1) \Rightarrow l_1 = l_2\ ;$$
$$\forall\, l_1, l_2, l_3 \in L: (l_1 \le l_2 \ \wedge l_2 \le l_3) \Rightarrow l_1 \le l_3\ ;$$

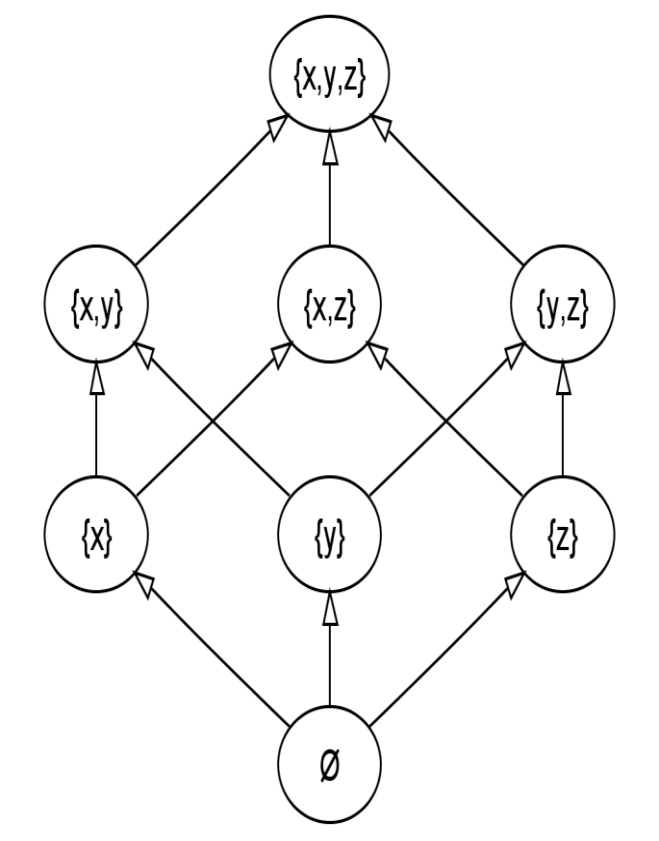
• – оператор, определяющий для любой пары $l_1, l_2 \in L$ наименьшую верхнюю границу -

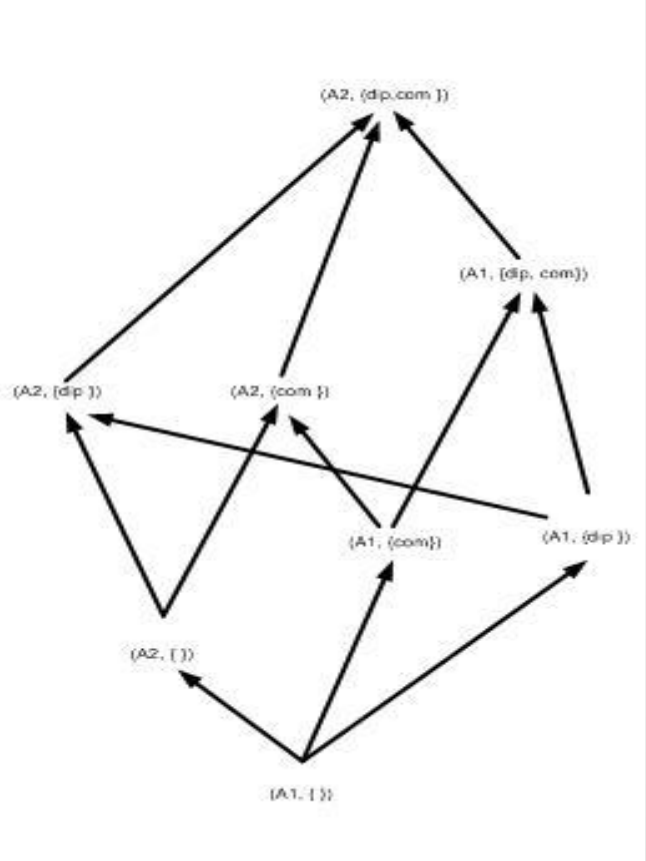
$$l_1 \bullet l_2 = l \ \Leftrightarrow l_1, l_2 \le l \wedge \forall\, l' \in L: (l' \le l) \Rightarrow (l' \le l_1 \vee l' \le l_2)$$

⊗ – оператор, определяющий для любой пары $l_1, l_2 \in L$ наибольшую нижнюю границу -

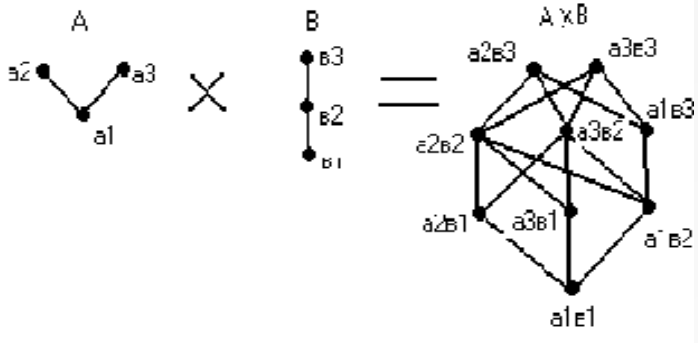
$$l_1 \otimes l_2 = l \ \Leftrightarrow l \le l_1, l_2 \wedge \forall\, l' \in L: (l' \le l_1 \wedge l' \le l_2) \Rightarrow (l' \le l)$$

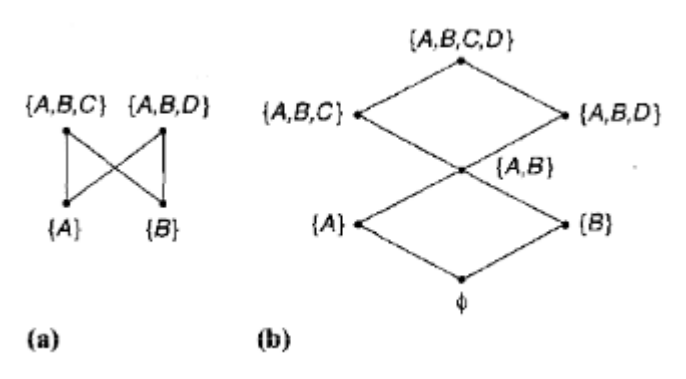
Решетка уровней безопасности Λ_L

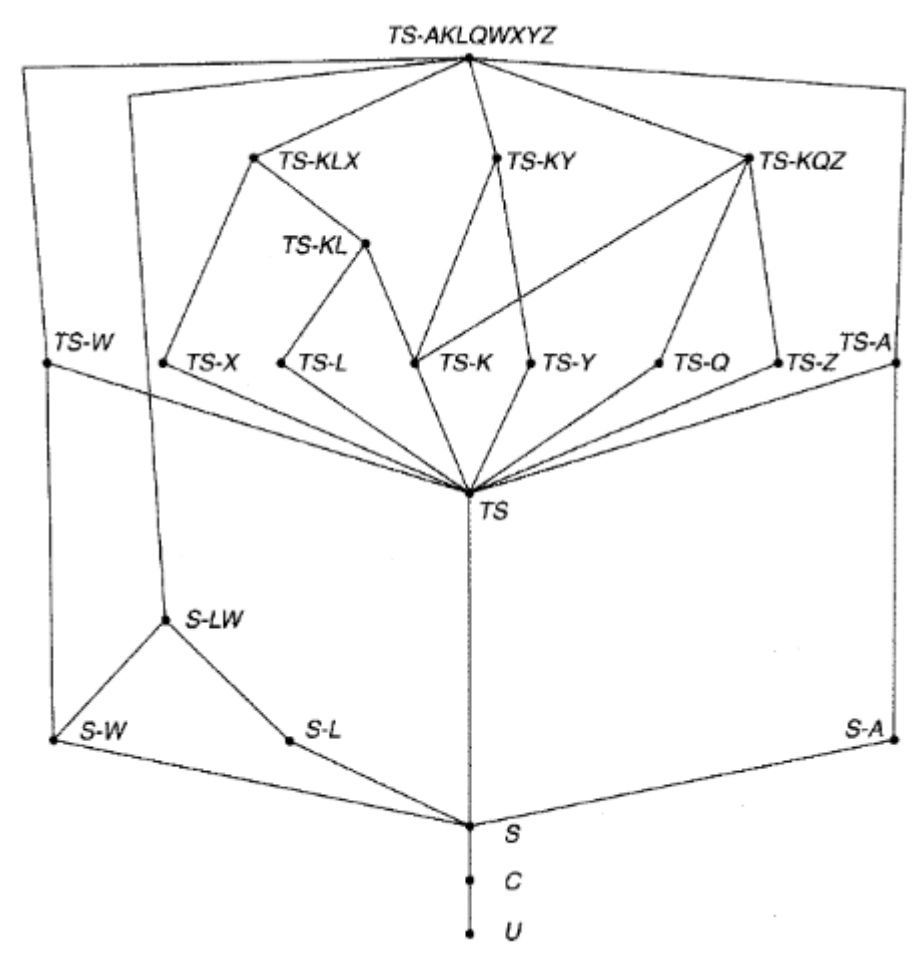




Решетка уровней безопасности Λ_L







Функция уровня безопасности $f_L: X \to L$

f_L - однозначное отображение множества сущностей $X = S \cup O$ во множество уровней безопасности L решетки Λ_L .

Обратное отображение $f_L^{-1}: L \to X$ задает разделение всех сущностей на классы безопасности X_i, такие что:

$X_1 \cup X_2 \cup \ldots \cup X_N = X$,

где N - мощность базового множества уровней безопасности L;

$X_i \cap X_j \equiv \varnothing$, где $i \neq j$;

$\forall x' \in X_i \Rightarrow f_L(x') = l_i$, где $l_i \in L$

Решетка и функция уровней и требования безопасности

1. Предположим, что информация может передаваться от сущностей класса X_i ***к сущностям класса*** X_j ***и наоборот, т. е. от сущностей класса*** X_j ***к сущностям класса*** X_i. *Тогда для выполнения критерия безопасности* сущности классов X_i ***и*** X_j *должны образовывать один общий* класс X_{ij}. *Для того чтобы не создавать избыточных классов, необходимо,* чтобы отношение, задаваемое оператором доминирования ≤, было антисимметричным.

2. Предположим, что информация может передаваться от сущностей класса X_i ***к сущностям класса*** X_j ***, и, кроме того, от сущностей класса*** X_j ***к сущностям класса*** X_k. *Если каждая такая передача безопасна в* отдельности, то, очевидно, безопасна и передача информации от сущностей класса X_i ***к сущностям класса*** X_k. *Таким образом, отношение,* задаваемое оператором ≤, должно быть транзитивным.

3. Внутри класса сущности имеют одинаковый уровень безопасности. Следовательно, передача информации между сущностями одного класса безопасна. Отсюда следует сравнимость по оператору ≤ сущностей одного класса между собой и самих с собой, т. е. рефлективность отношения ≤.

Решетка и функция уровней и требования безопасности

4. Предположим, что имеется два различных класса сущностей X_i **и** X_j. Тогда, из соображений безопасности очевидно, что существует только один класс **X'**, потоки от сущностей которого безопасны по отношению к сущностям класса X_i или класса X_j , совпадающий с классом X_i или с классом X_j , и не имеется никакого другого класса ***X"*** , менее безопасного чем ***X'*** , и с такими же возможностями по потокам к сущностям классов X_i **и** X_j **.** Это означает, что ***X'*** должен быть наименьшей верхней границей по уровням безопасности классов X_i **и** X_j **.**

5. Аналогично по условиям предыдущего пункта должен существовать ближайший снизу к классам X_i **и** X_j класс ***X',*** такой, что потоки от сущностей классов X_i **и** X_j к сущностям класса ***X'*** безопасны, и совпадающий с классом X_i или с классом X_j **,** при этом не имеется никакого другого класса X" , более безопасного, чем ***X'*** , и с такими же возможностями по потокам от сущностей классов X_i **и** X_j**.** Это означает, что ***X'*** должен быть наибольшей нижней границей по уровням безопасности классов X_i **и** X_j**.**

Модель Белла-ЛаПадулы

Классическая модель Белла — Лападулы была описана в 1975 году сотрудниками компании MITRE Corporation Дэвидом Беллом и Леонардом Лападулой, к созданию модели их подтолкнула система безопасности для работы с секретными документами Правительства США.

Суть системы заключалась в следующем: каждому субъекту (лицу, работающему с документами) и объекту (документам) присваивается метка конфиденциальности, начиная от самой высокой («особой важности»), заканчивая самой низкой («несекретный» или «общедоступный»). Причем субъект, которому разрешён доступ только к объектам с более низкой меткой конфиденциальности, не может получить доступ к объекту с более высокой меткой конфиденциальности. Также субъекту запрещается запись информации в объекты с более низким уровнем безопасности.

Модель Белла-ЛаПадулы

Известны варианты описания модели:

1. Классическая модель Белла — ЛаПадулы
2. Модель Белла — ЛаПадулы, подход RW
3. Модель Белла — ЛаПадулы, простое описание

.....

.....

.....

Модель Белла-ЛаПадулы

Система защиты - совокупность

- множества субъектов S
- множества объектов O
- множества прав доступа R (рассматриваем всего два элемента - *read* и *write*)
- матрицы доступа $A[s,o]$
- решетки уровней безопасности L субъектов и объектов (допуска и грифы секретности)
- функции уровней безопасности f_L, отображающей элементы множеств S и O в L
- множества состояний системы V, которое определяется множеством упорядоченных пар (f_L,A)
- начального состояния v_0
- набора запросов Q субъектов к объектам, выполнение которых переводит систему в новое состояние
- функции переходов $\mathcal{F}_T : (V \times Q) \rightarrow V$, которая переводит систему из одного состояния в другое при выполнении запросов

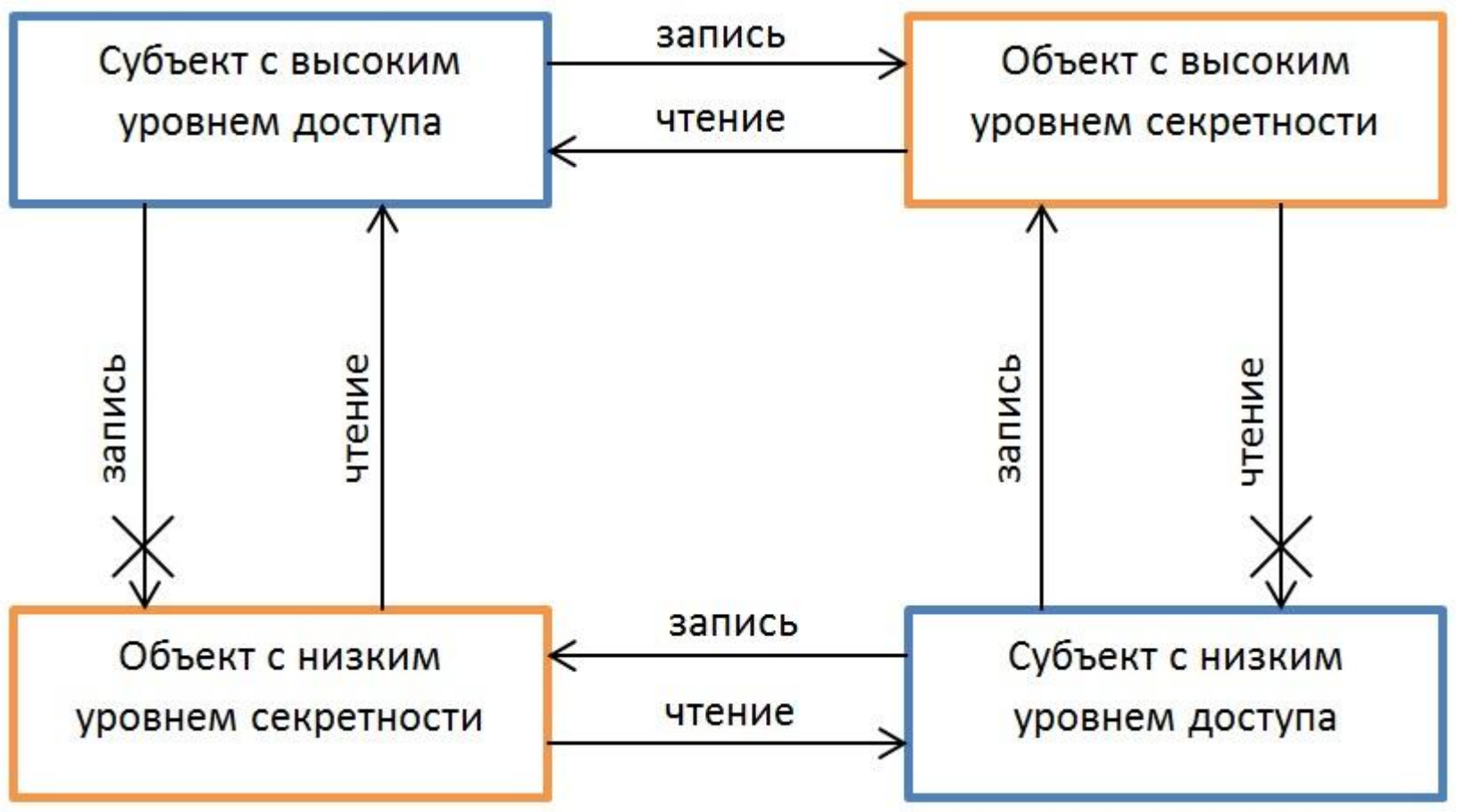
Модель Белла-ЛаПадулы

Простое правило безопасности (simple security rule – ss-правило) говорит о том, что субъект не может читать данные, находящиеся на более высоком уровне безопасности, чем его допуск.

Правило *-свойства (*-property rule, star property rule) говорит о том, что субъект не может записывать данные на меньший уровень безопасности, чем его допуск.

Простое правило безопасности это правило «не читать сверху» (no read up - NRU), а правило *-свойства – правило «не записывать вниз» (no write down - NWD)

Модель Белла-ЛаПадулы



Модель Белла-ЛаПадулы

Строгое правило *-свойства (strong *-property rule) говорит о том, что субъект может выполнять функции чтения и записи только на том же уровне безопасности, что и его допуск – не выше и не ниже.

Таким образом, чтобы субъект имел возможность чтения и/или записи объекта, его уровень допуска должен совпадать с классификацией объекта.

Модель Белла-ЛаПадулы

1. Состояние называется безопасным по чтению (ss-безопасным или *просто безопасным*) тогда и только тогда, когда для каждого субъекта, осуществляющего в этом состоянии доступ чтения к объекту, уровень безопасности этого субъекта доминирует над уровнем безопасности этого объекта:

$$\forall s \in S, \forall o \in O, read \in A[s,o] \rightarrow f_L(s) \geq f_L(o)$$

2. Состояние называется безопасным по записи (или **-безопасным*) тогда и только тогда, когда для каждого субъекта, осуществляющего в этом состоянии доступ записи к объекту, уровень безопасности объекта доминирует над уровнем безопасности этого субъекта:

$$\forall s \in S, \forall o \in O, write \in A[s,o] \rightarrow f_L(o) \geq f_L(s)$$

3. Состояние безопасно тогда и только тогда, когда оно безопасно *и по чтению, и по записи*

Модель Белла-ЛаПадулы

Система $\Sigma(v_0, Q, \mathcal{F}_T)$ безопасна тогда и только тогда, когда ее начальное состояние v_0 безопасно и все состояния, достижимые из v_0 в результате конечной последовательности запросов из Q безопасны.

Замечание: В качестве элементов Q могут быть запросы:

1. Изменение множества текущих доступов b (получить или отменить доступ).
2. Изменение функции f_L (изменить уровень конфиденциальности)
3. Изменение множества разрешений на доступ A (добавить разрешение в A или удалить разрешение из A)

Модель Белла-ЛаПадулы

Теорема ОТБ (БТБ). Система $\Sigma(v_0, Q, \mathcal{F}_T)$ безопасна тогда и только тогда, когда:

1. Состояние v_0 безопасно

2. Функция переходов $\mathcal{F}_T$ такова, что любое состояние v, достижимое из v_0 при выполнении конечной последовательности запросов из множества Q, также безопасно т. е. если при $\mathcal{F}_T(v,q)=v^*$, *где* $v=(f_L, A)$ *и* $v^*=(f_L^*, A^*)$, переходы системы из состояния в состояние подчиняются следующим ограничениям для

$\forall s \in S$ и для $\forall o \in O$:

- **если read $\in A^*[s,o]$ и read $\notin A[s,o]$, то $f_L^*(s) \geq f_L^*(o)$**
- **если read $\in A[s,o]$ и $f_L^*(s) < f_L^*(o)$, *то* read $\notin A^*[s,o]$**
- **если write $\in A^*[s,o]$ и write $\notin A[s,o]$, то $f_L^*(s) \leq f_L^*(o)$**
- **если write $\in A[s,o]$ и $f_L^*(o) < f_L^*(s)$, то write $\notin A^*[s,o]$**

Модель Белла-ЛаПадулы

Доказательство.

Необходимость.
Предположим, что система безопасна.
Тогда состояние v_0 безопасно по определению. Предположим также, что существует некоторое состояние v, достижимое из состояния v_0 путем исполнения конечной последовательности запросов из Q, при которых $\mathcal{F}_T(v,q)=v^*$.
И пусть при этом одно из первых двух ограничений (по чтению) не выполняется:

- если read $\in A^*[s,o]$ и read $\notin A[s,o]$, то $f_L^*(s) \geq f_L^*(o)$
- если read $\in A[s,o]$ и $f_L^*(s) < f_L^*(o)$, *то* read $\notin A^*[s,o]$

-тогда, хотя состояние v* и является достижимым, но небезопасно по определению. Если в состоянии v* не выполняется одно из двух последних (по записи) ограничений :

- если write $\in A^*[s,o]$ и write $\notin A[s,o]$, то $f_L^*(s) \leq f_L^*(o)$
- если write $\in A[s,o]$ и $f_L^*(o) < f_L^*(s)$, то write $\notin A^*[s,o]$

-то в этом случае состояние v* также будет небезопасным по определению.

Модель Белла-ЛаПадулы

Доказательство.

Достаточность.
Предположим, что система небезопасна. Тогда
должно быть небезопасным либо состояние v_0, либо состояние v^*,
достижимое из v_0 путем исполнения конечной последовательности
запросов из Q .
Исходное состояние v_0 безопасно по условию теоремы. Тогда, так как v_0
безопасно, небезопасно какое-либо состояние v^*, переход в которое
осуществляется из безопасного состояния v: $\mathcal{F}_T(v,q)=v^*$. Однако это
противоречит четырем ограничениям на переходы $\mathcal{F}_T$ по условиям
теоремы. Следовательно, такой переход невозможен, и данное
утверждение неверно.
Таким образом, условия теоремы достаточны для безопасности системы
$\Sigma(v_0,Q,\mathcal{F}_T)$.
Теорема доказана. ■

Модель Белла-ЛаПадулы

Основной смысл теоремы ОТБ.

«Если начальное состояние системы безопасно, и все переходы системы из состояния в состояние не нарушают ограничений, сформулированных политикой безопасности, то любое состояние системы безопасно».

Классическая модель Белла-ЛаПадулы 2

Система защиты - совокупность

- множества субъектов *S*
- множества объектов *O*
- множества прав доступа *R* (***read, write, append и execute***)
- Множество возможных множеств текущих доступов **B={b⊆SxOxR}**
- матрицы разрешенных доступов *A*[*s*,*o*]⊆*R*
- решетки уровней секретности *L* субъектов и объектов (U<C<S<TS)
- функции уровней безопасности f_L, (f_s , f_o , f_c)для множеств *S* и *O* в *L*, *где:*

f_o - уровень доступа объекта

f_s - уровень доступа субъекта

f_c - текущий уровень доступа субъекта $f_c \le f_s$

Классическая модель Белла-ЛаПадулы

- множества состояний системы V, которое определяется множеством (B, f_L, A)
- начального состояния v_0
- набора запросов Q субъектов к объектам
- множество ответов по запросам $D=\{yes,no,error\}$
- множество действий системы $W=\{w=(q,d,v^*,v)\}$
- множество значений времени $N_0=\{0,1,\ldots.\}$
- множество $\boldsymbol{X}$ функций $x: N_0 \rightarrow Q$, задающих все возможные последовательности запросов к системе
- множество $\boldsymbol{Y}$ функций $y: N_0 \rightarrow D$, задающих все возможные последовательности ответов на запросы к системе
- множество $\boldsymbol{Z}$ функций $z: N_0 \rightarrow V$, задающих все возможные последовательности состояний системы

Классическая модель Белла-ЛаПадулы

$\sum(Q,D,W,z_o) \subseteq \boldsymbol{X} \times \boldsymbol{Y} \times \boldsymbol{Z}$ называется системой, если выполняется $\boldsymbol{(x,y,z)} \in \sum(Q,D,W,z_o)$ тогда и только тогда, когда $\boldsymbol{(x_t,y_t,z_{t+1},z_t)} \in W$ для каждого, $\boldsymbol{t} \in \boldsymbol{N_o}$ где $\boldsymbol{z_o}$ -начальное состояние системы.

При этом каждый набор $\boldsymbol{(x,y,z)} \in \sum(Q,D,W,z_o)$ называется реализацией системы, а $\boldsymbol{(x_t,y_t,z_{t+1},z_t)} \in W$ -действием системы $\forall\ \boldsymbol{t} \in \boldsymbol{N_o}$

Безопасность системы определяется с помощью трех свойств:
1) ss-свойства простой безопасности (simple security);
2) *- свойства "звезда";
3) ds-свойства дискретной безопасности (discretionary security).

Классическая модель Белла-ЛаПадулы

Доступ (s,o,r) обладает **ss-свойством** относительно $f_L = (f_s, f_o, f_c)$, где

f_s - функция уровней доступа субъектов,

f_o - функция уровней конфиденциальности объектов,

f_c - функция текущих уровней доступа субъектов,

когда выполняется одно из условий:

r ∈ {execute;append};

r ∈ {read;write} и $f_s(s) \geq f_o(o)$.

Доступ *(s,o,r)* обладает *-свойством относительно $f_L = (f_s, f_o, f_c)$, когда выполняется одно из условий:

r=execute;

r=append и $f_o(o) \geq f_c(s)$;

r=read и $f_c(s) \geq f_o(o)$.;

r=write и $f_s(s) = f_o(o)$.

Состояние системы *(b,f,A)* обладает **ss-свойством (*-свойством)**, когда в нём все доступы обладают ss-свойством (*-свойством) относительно f_L.

Состояние системы *(b,f,a)* обладает **ds-свойством**, когда в нём для каждого доступа *(s,o,r)* выполняется условие $r \in a[s;o]$.

Классическая модель Белла-ЛаПадулы

Состояние системы $(b,f,A) \in V$ обладает *-свойством, относительно подмножества $S' \subseteq S$, если каждый элемент $(s,o,r) \in b$, где $s \in S'$ обладает *-свойством относительно f_L. При этом $S \backslash S'$ называются множеством доверенных субъектов, т.е. субъектов, имеющих право нарушать политику безопасности.

Состояние системы (b,f,A) *называется безопасным*, если обладает *-свойством относительно S', ss-свойством и ds-свойством.

Реализация системы $(x,y,z) \in \sum(Q,D,W,z_o)$ обладает ss-свойством (*-свойством, ds-свойством), если в последовательности $(z_0,z_1,\ldots)$ каждое состояние обладает ss-свойством (*-свойством, ds-свойством).

Система $\sum(Q,D,W,z_o)$ обладает ss-свойством (*-свойством, ds-свойством), если каждая ее реализация обладает ss-свойством (‘*-свойством, ds-свойством).

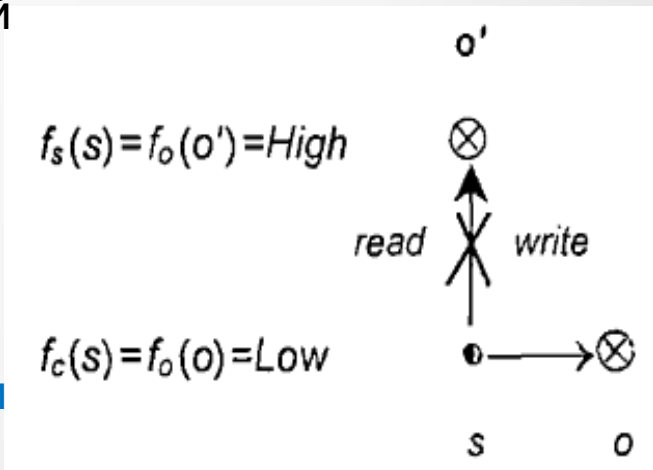
Система $\sum(Q,D,W,z_o)$ называется безопасной, если она обладает ss-свойством, *-свойством, ds-свойством одновременно.

Классическая модель Белла-ЛаПадулы

Во-первых, из обладания доступом *-свойством относительно f_L *следует* обладание этим доступом ss-свойством относительно f_L.

Во-вторых, из обладания системой ss-свойством следует, что в модели БЛ выполняется запрет на чтение вверх, принятый в мандатной (полномочной) политике безопасности. Кроме того, ss-свойство не допускает модификацию с использованием доступа *write, если* $f_s(s) < f_0(o)$. *Таким образом, функция* $f_s(s)$ определяет для субъекта s верхний уровень секретности объектов, к которым он потенциально может получить доступ *read или write.*

В-третьих. Если субъект s может понизить свой текущий доступ до $f_c(s) = f_0(o)$, *то он может получить доступ write к объекту* o, но не доступ *read к объекту o', с уровнем* $f_0(o') > f_c(s)$.
Хотя при этом, возможно, выполняется $f_s(s) = f_0(o)$, *и в каких-то других состояниях системы* субъект s может получить доступ *read к объекту o' Таким образом, * -* свойство исключает появление в системе канала утечки информации сверху вниз и соответствует требованиям мандатной (полномочной) политики безопасности.



Классическая модель Белла-ЛаПадулы

Теорема A1. Система $\sum(Q,D,W,z_o)$ обладает ss-свойством для любого начального состояния z_o, обладающего ss-свойством, тогда и только тогда, когда $\forall(q,d,(b^*,A^*,f^*),(b,A,f)) \in W$ удовлетворяет условиям:
Условие 1. $\forall(s,o,r)\in b^*\backslash b$ обладает ss-свойством относительно f^*
Условие 2. Если $(s,o,r)\in b$ и не обладает ss-свойством относительно f^*, то $(s,o,r)\notin b^*$.

Теорема A2. Система $\sum(Q,D,W,z_o)$ обладает *-свойством относительно подмножества $S'\subseteq S$ для любого начального состояния z_o, обладающего *-свойством, тогда и только тогда, когда $\forall(q,d,(b^*,A^*,f^*),(b,A,f))\in W$ удовлетворяет условиям:
Условие 1. $\forall(s,o,r)\in b^*\backslash b$ обладает *-свойством относительно f^*
Условие 2. Если $(s,o,r)\in b$ и не обладает *-свойством относительно f^*, то $(s,o,r)\notin b^*$.

Теорема A3. Система обладает ds-свойством для любого начального состояния z_o, обладающего ds-свойством, тогда и только тогда, когда $\forall(q,d,(b^*,A^*,f^*),(b,A,f))\in W$ удовлетворяет условиям: :
Условие 1. $\forall(s,o,r)\in b^*\backslash b$ выполняется $r\in a(s,o)$
Условие 2 Если $(s,o,r)\in b$ и $r \notin a(s,o)$, то $(s,o,r)\notin b^*$.

Классическая модель Белла-ЛаПадулы

Теорема А1.

Достаточность.

Пусть выполнены условия 1 и 2 и пусть $(x,y,z) \in \sum(Q,D,W,z_o)$ - произвольная реализация системы. *Тогда* $(x_t,y_t,(b_{t+1},A_{t+1},f_{t+1}),(b_t,A_t,f_t)) \in W$, где $z_{t+1}=(b_{t+1},A_{t+1},f_{t+1})$, $z_t=(b_t,A_t,f_t)$ для $\forall t \in N_0$.

Для любого $(s,o,r) \in b_{t+1}$ выполняется или $(s,o,r) \in b_{t+1} \setminus b_t$ или $(s,o,r) \in b_t$. Из условия 1 следует, что состояние системы z_{t+1} пополнилось доступами, которые обладают ss-свойством относительно *f** Из условия 2 следует, что доступы из b_t, которые не обладают ss-свойством относительно *f**, не входят в b_{t+1}. Следовательно, $\forall (s,o,r) \in b_{t+1}$ обладают ss-свойством относительно f* и по определению состояние z_{t+1} обладает ss-свойством для $\forall t \in N_0$. Так как по условию и состояние z_o обладает ss-свойством, то выбранная нами произвольная реализация (x,y,z) также обладает ss-свойством. Достаточность доказана.

Классическая модель Белла-ЛаПадулы

Теорема А1.

Необходимость.
Пусть система $\sum(Q,D,W,z_o)$ обладает ss-свойством. Будем считать, что в множество *W* входят только те действия системы, которые используются в ее реализациях. Тогда для каждого $(x,y,(b^*,A^*,f^*),(b,A,f))\in W$ существует реализация $(x, y, z)\in\sum(Q,D,W,z_o)$ и существует $t\in N_0$:

$$(x,y,(b^*,A^*,f^*),(b,A,f))= (x_t,y_t,(b_{t+1},A_{t+1},f_{t+1}),(b_t,A_t,f_t))=(x_t,y_t,z_{t+1},z_t)$$

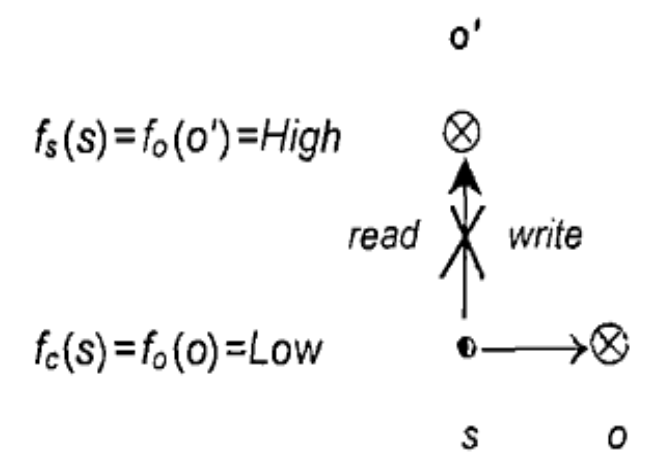
Так как реализация (x, y, z) обладает ss-свойством, то и состояние $z_{t+1}=(b^*,A^*,f^*)$ обладает ss-свойством по определению.
Следовательно, условия 1 и 2 очевидно выполняются.
Необходимость доказана.
■

Классическая модель Белла-ЛаПадулы

Пример.

Пусть субъект запрашивает доступ «read» на объект o'.
Возможны следующие ответы по запросу:

1. Запретить субъекту s запрашиваемый им доступ «read» на объект o'.
2. Закрыть доступ «write» на объект o. Повысить текущий уровень конфиденциальности $f_c(s)$ до High. Разрешить субъекту s запрашиваемый им доступ «read» на объект o'.



Модель Белла-ЛаПадулы – RW подход. 3

При изучении классической модели Белл-ЛаПадулы следует иметь ввиду, что она разрабатывалась для обеспечения безопасности конкретной защищенной операционной системы Multics. Поэтому некоторые элементы модели (например права доступа: append, execute, функция текущего уровня доступа субъектов…) реализованы в модели только для обеспечения соответствия условиям функционирования ОС Multics и не являются необходимыми для моделирования произвольной системы с мандатным (полномочным) управлением доступом.

Положим *R - {read, write} и* $f_s(s)=f_c(s)$. *Исключим из рассмотрения матрицу доступов A и множество ответов системы D. Вместо множества действий используем функцию переходов как описывалось выше* $\mathcal{F}_T : (V \times Q) \to V$ *и множество доступов* $b=\{(s,o,r)\}$

Переопределим ss-свойство и *-свойство. Так как основные ограничения на доступ *write следуют из *-свойства, то в ss-свойстве зададим* ограничения только на *read.*

Доступ (s,o,r) $\in$ *b обладает ss-свойством, если выполняется* одно из условий:

- *r=write*
- *r=read и* $f_s(s)>f_0(o)$.

Доступ (s,x,r) $\in$ *b обладает *-свойством, если выполняется* одно из условий:

- *r=read и не существует* $y \in O$: $(s, y, write) \in b$ *и* $f_0(x)>f_0(y)$;
- *rewrite и не существует* $y \in O$: $(s, y, read) \in b$ *и* $f_0(y)>f_0(x)$.

Модель Белла-ЛаПадулы – RW подход. 3

Заметим особо, что в *-свойстве не рассматривается уровень доступа субъекта посредника s. В этом нет необходимости, так как если требовать выполнения *-свойства и ss-свойства одновременно и считать, что субъект не может накапливать в себе информацию, то не возникают противоречия по существу с положениями мандатной (полномочной) политики безопасности. Субъект может читать информацию из объектов с уровнем секретности не выше его уровня доступа, и при этом субъект не может стать каналом утечки информации сверху вниз.

В данном подходе *-свойство определено таким образом, что его смысл - предотвращение возникновения информационных каналов сверху вниз становится более ясным, чем при использовании функции *fc в классической* модели БЛ.

По аналогии со свойствами классической модели БЛ определяются ss-свойство, *-свойство и свойство безопасности для состояния, реализации и системы в целом.

Модель Белла-ЛаПадулы – RW подход. 3

Теорема A1-RW. Система $\sum(V,T,z_o)$ *обладает ss-свойством для любого* начального состояния z_o, обладающего ss-свойством, тогда и только тогда, когда функция переходов $\mathcal{F}_T$: $(v \times q) = v^*$ *удовлетворяет условиям:*
Условие 1. Если $(s,o,read) \in b^* \setminus b$, *то* $f_s^*(s)=f_0^*(o)$.
Условие 2. Если $(s,o,read) \in b$ *и* $f_s^*(s) \leq f_0^*(o)$, *то* $(s,o,read) \notin b^*$.

Теорема A2-RW. Система $\sum(V,T,z_o)$ обладает *-свойством для любого начального состояния z_o, обладающего *-свойством, тогда и только тогда, когда функция переходов $\mathcal{F}_T$: $(v \times q) = v^*$ удовлетворяет следующим условиям.
Условие 1. Если $\{(s,x,read), (s,y,write)\} \subseteq b^* \setminus b$, *то* $f_o(y)=f_o(x)$.
Условие 2. Если $\{(s,x,reacf), (s,y,write)\} \subseteq b$ *и* $f_o(y)<f_o(x)$, *то* $\{(s,x,read),(s,y,write)\}$ *не принадлежит* b^*

Теорема BST-RW. Система $\sum(V,T,z_o)$ *безопасна для безопасного начального* состояния z_o тогда и только тогда, когда выполнены условия теоремы A1-RW и теоремы A2-RW.
Доказательство. Теорема BST-RW следует из теорем A1-RW, A2-RW. ■

Модель Белла-ЛаПадулы

- Ясность и простота реализации.
- Отсутствие проблемы "Троянских коней" (контролируется направленность потоков, а не взаимоотношения конкретного субъекта с конкретным объектом, поэтому недекларированный поток троянской программы «сверху-вниз» будет считаться опасным и отвергнут МБО).
- Каналы утечки не заложены в саму модель, а могут возникнуть только в практической реализации.
- Модель Белла-ЛаПадулы сыграла огромную роль в развитии теории компьютерной безопасности, и ее положения были введены в качестве обязательных требований к системам, обрабатывающим информацию, содержащую государственную тайну, в стандартах защищенных КС, в частности, в известной "Оранжевой книге" (1983г.).

Модель Белла-ЛаПадулы

Завышение уровня секретности - вытекает из одноуровневой природы объектов. Это означает, что некоторой информации может быть дан уровень секретности выше того, что она заслуживает. Пример - не секретный параграф в секретном сообщении.

Запись вслепую - это проблема, вытекающая из правила NRU. Субъект производит запись объекта с более высоким уровнем безопасности - эта операция не нарушает правила NWD. Однако после завершения операции субъект не может проверить правильность выполнения записи объекта путем выполнения контрольного чтения, так как это нарушает правило NRU.

Модель Белла-ЛаПадулы

Привилегированные субъекты - эта проблема связана с работой системного администратора. Функционирование системного администратора подразумевает выполнение в системе таких критических операций, как добавление и удаление пользователей, восстановление системы после аварий... очевидно, что такие операции не вписываются в модель

Удаленная запись - это проблема, вытекающая из правила NWD. В распределенных системах операция чтения инициируется запросом с одной компоненты на другую, что можно рассматривать в данном случае как посылку сообщения от субъекта с более высоким уровнем безопасности к объекту с более низким уровнем

Модель Белла-ЛаПадулы

Деклассификация - данная проблема заключается в том, что классическая модель не предотвращает систему от деклассификации объекта (изменение уровня секретности объекта вплоть до «не секретно» по желанию «совершенно секретного» субъекта). Например, пусть субъект с высоким уровнем доступа А читает информацию из объекта того же уровня секретности. Далее он понижает свой уровень доступа до низкого Б, и записывает считанную ранее информацию в объект, низкого уровня секретности Б. Таким образом, хотя формально модель нарушена не была, безопасность системы нарушена. Также, последовательно понижая свой уровень безопасности, субъект может получить доступ к нужному объекту по записи (Z-модель McLean)

Для решения этой проблемы вводят правила:
Правило сильного спокойствия — уровни безопасности субъектов и объектов никогда не меняются в ходе системной операции.
Правило слабого спокойствия — уровни безопасности субъектов и объектов никогда не меняются в ходе системной операции таким образом, чтобы нарушить заданную политику безопасности.

Безопасная функция перехода (*МакЛин*)

Недостаток основной теоремы безопасности Белла-ЛаПадулы состоит в том, что ограничения, накладываемые теоремой на функцию перехода, совпадают с критериями безопасности состояния, поэтому данная теорема является избыточной по отношению к определению безопасного состояния.

Кроме того все состояния, достижимые из безопасного состояния при определенных ограничениях, будут в некотором смысле безопасны, по при этом не гарантируется, что они будут достигнуты без потери свойства безопасности в процессе осуществления перехода.

Безопасная функция перехода (*МакЛин*)

Функция перехода $\mathcal{F}_T(v,q)=v^*$ *безопасна* по чтению когда:

1. Если ***read*** $\in A^*[s,o]$ и ***read*** $\notin A[s,o]$, то $f_{Ls}(s) \geq f_{Lo}(o)$ и $f_L = f_L^*$
2. Если $f_{Ls} \neq f_{Ls}^*$, то $A=A^*$, $f_{Lo}=f_{Lo}^*$, для $\forall s$ и o, у которых $f_{Ls}^*(s) < f_{Lo}^*(o)$, - ***read*** $\notin A[s,o]$
3. Если $f_{Lo} \neq f_{Lo}^*$, то $A=A^*$, $f_{Ls}=f_{Ls}^*$, для $\forall s$ и o, у которых $f_{Ls}^*(s) < f_{Lo}^*(o)$, - ***read*** $\notin A[s,o]$

Функция перехода $\mathcal{F}_T(v,q)=v^*$ *безопасна* по записи когда:

1. Если ***write*** $\in A^*[s,o]$ и ***write*** $\notin A[s,o]$, то $f_{Lo}(o) \geq f_{Ls}(s)$ и $f_L = f_L^*$
2. Если $f_{Ls} \neq f_{Ls}^*$, то $A=A^*$, $f_{Lo}=f_{Lo}^*$, для $\forall s$ и o, у которых $f_{Ls}^*(s) > f_{Lo}^*(o)$, - ***write*** $\notin A[s,o]$
3. Если $f_{Lo} \neq f_{Lo}^*$, то $A=A^*$, $f_{Ls}=f_{Ls}^*$, для $\forall s$ и o, у которых $f_{Ls}^*(s) > f_{Lo}^*(o)$, - ***write*** $\notin A[s,o]$

Безопасная функция перехода (*МакЛин*)

Смысл данных ограничений модели МакЛина в том, что:

1. В процессе перехода может меняться только один компонент системы безопасности:
   - ❑Ячейка матрицы доступа (отношение доступа определенного субъекта к определенному объекту);
   - ❑Решетка уровней безопасности субъектов;
   - ❑Решетка уровней безопасности объектов.

2. Переход может производиться только при условии того, что правила NRU и NWD будут соблюдаться как в предыдущем, так и в последующем состоянии.

Безопасная функция перехода (*МакЛин*)

Теорема безопасности МакЛина.

Система безопасна в любом состоянии и в процессе переходов между ними, если ее начальное состояние является безопасным, а ее функция перехода удовлетворяет критерию Мак-Лина.

Обратное утверждение неверно.
Система может быть безопасной по определению Белла-ЛаПадулы, но не иметь безопасной функции перехода.

Уполномоченные субъекты

В базовую модель дополнительно вводится подмножество доверенных субъектов, которым (и только им) разрешается инициировать переходы с изменениями уровней безопасности сущностей системы – $C(S)$

Соответственно функция переходов системы $\Sigma(v_0,Q,\ \mathcal{F}_T^a)$ – $\mathcal{F}_T^a$ приобретает дополнительный параметр авторизации

Функция перехода $\mathcal{F}_T^a(v,s,q)$ в модели с называется авторизованной тогда и только тогда, когда для каждого перехода $\mathcal{F}_T^a(v,s,q)=v^*$, при котором:

для $\forall x \in S \cup O$: **если** $f_L^*(x) \neq f_L(x)$, **то** $s \in C(S)$

Система $\Sigma(v_0,Q,\ \mathcal{F}_T^a)$ с доверенными субъектами безопасна если :

1.Начальное состояние v_0 безопасно и все достижимые состояния безопасны по критерию Белла-ЛаПадулы

2.Функция переходов $\mathcal{F}_T^a$ является *авторизованной*

Модель low-water-mark (LWM)

Суть модели LWM состоит в том, что если по определенным соображениям субъектам нельзя отказывать в доступе write к любым объектам системы, то для исключения опасных информационных потоков сверху вниз, необходимо дополнить определенными правилами операцию write и ввести дополнительную операцию reset.

В результате команды reset уровень безопасности объекта автоматически повышается до максимального уровня безопасности в системе. В результате объект становится доступным для записи для всех субъектов системы.

Если субъекту требуется внести информацию в объект с более низким, чем у субъекта, уровнем безопасности (что запрещено правилом NWD), то субъект может подать команду reset. При этом опасности перетекания информации сверху вниз не создается, так как по чтению система руководствуется правилом NRU.

Модель low-water-mark (LWM)

Операция	Условие выполнения	Результат выполнения операции
Read (s, o)	$f_s(s) \geq f_o(o)$	$f^* = f$; $b^* = b \cup \{(s, o, read)\}$
Write (s, o)	$f_s(s) = f_o(o)$	$f_s^{\cdot} = f_s$, $\forall o' \neq o$ $f_o^{\cdot}(o') = f_o(o')$, $f_o^{\cdot}(o) = f_s(s)$, *If* $(f_o^{\cdot}(o) < f_o(o))$ *Then* $o = \varnothing$, $b^* = b \cup \{(s, o, read)\}$
Reset (s, o)	$f_s(s) > f_o(o)$	$f_s^{\cdot} = f_s$, $\forall o' \neq o$ $f_o^{\cdot}(o') = f_o(o')$, $f_o^{\cdot}(o) = \max(L)$

Модель low-water-mark (LWM)

Доступ $(s,o,r) \in SxOxR$ обладает ss-свойством относительно $f \in F$, если $r \in \{read, write\}$ и $f_s(s) > f_0(o)$.

Доступ $(s,o,r) \in SxOxR$ обладает *-свойством относительно
относительно $f \in F$, если он удовлетворяет одному из условий:
- $r = read$ и $f_s(s) > f_0(o)$.
- $r = write$ и $f_s(s) = f_0(o)$.

Состояние системы $(b,f) \in V$ обладает ss-свойством (*-свойством), если каждый элемент $(s,o,r) \in b$ обладает ss-свойством (*-свойством) относительно f.

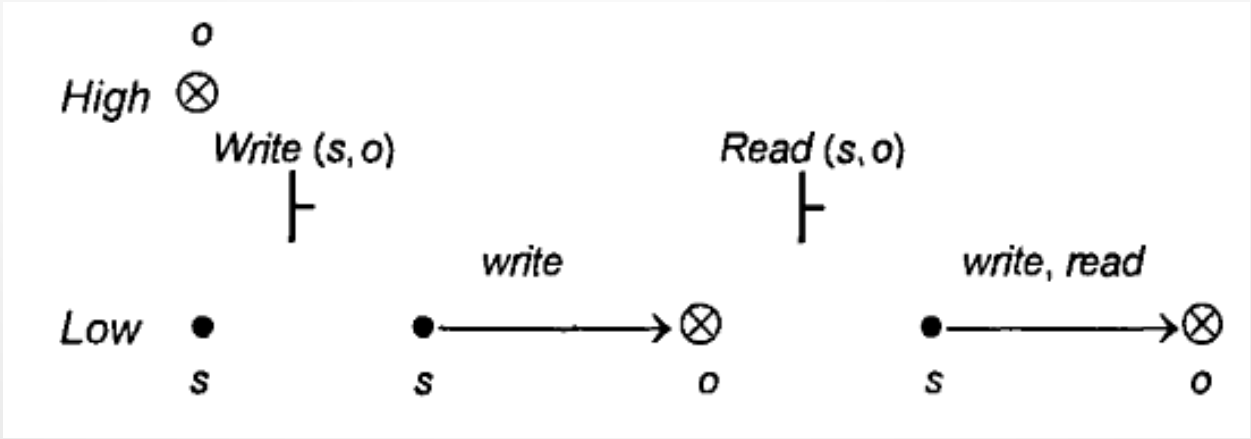
Состояние системы называется безопасным, если оно обладает ss-свойством и *-свойством одновременно.

Утверждение.
Операции Read, Write, Reset переводят систему из безопасного состояния в безопасное состояние.
Доказательство. Из описания операций Read, Write, Reset следует, что в результате их выполнения новое состояние системы обладает ss-свойством и *-свойством. ■

Модель low-water-mark (LWM)

Заметим, что условие стирания информации при выполнении операции Write является существенным. Хотя при его отсутствии утверждение формально останется истинным.

Однако в этом случае с позиций здравого смысла система не будет безопасной, так как возможно возникновение канала утечки информации. Субъект, запрашивая доступ на запись в объект, понижает его уровень секретности, после чего он может запросить доступ на чтение из этого объекта.



Модель low-water-mark (LWM)

Однако такая модель порождает тенденцию к «огрублению» системы разграничения доступа: все объекты через некоторое время могут получить высшие грифы секретности и, соответственно, стать недоступными по чтению для всех субъектов, не обладающих высшим уровнем доверия.
Поэтому в модели LWM операция дополняется следующим условием:
Если в результате операции вся в объект реально вносится информация и уровень безопасности объекта строго выше уровня безопасности субъекта, то:

- перед внесением информации в объект, вся прежняя информация из него удаляется;
- по окончании операции записи уровень безопасности объекта автоматически понижается до уровня безопасности субъекта.

Одним из вариантов является присвоение объекту при записи уровня доступа не максимально возможного (командой reset), а соответствующего субъекту, осуществляющему запись.

Модель low-water-mark (LWM)

Модель является конкретизацией модели Белла-Ла-Падула, политика безопасности задается в следующих предположениях:

- все компоненты КС классифицированы по уровню конфиденциальности
  - $f_s(S)$ - уровень доступа субъекта,
  - $f_c(S)$ - текущий уровень доступа субъектов,
  - $f_o(O)$ - гриф (уровень) конфиденциальности объекта,
- поток информации (в данном случае рассматриваются потоки от объектов к ассоциированным объектам некоторого субъекта) разрешен только "снизу вверх" (в смысле повышения уровня конфиденциальности).

Модель MMS

Модель является конкретизацией модели Белла-Ла-Падула,

Модель совместного доступа

Для того, чтобы мандатная модель предусматривала совместный доступ необходимо ее модифицировать следующим образом.

- Вместо множества субъектов системы S будем рассматривать множество непустых подмножеств P(S).
- Расширим матрицу прав доступа, отражающую текущее состояние доступа в системе, путем добавления в нее строк, содержащих права групповых субъектов, и обозначим ее как M.
- Для определения функции уровня безопасности для групповых субъектов вводятся дополнительные функции определяющие наибольшую нижнюю границу и наименьшую верхнюю границу для уровней групповых субъектов.

Модель совместного доступа

Функцией уровня безопасности $F^L : S_G \rightarrow L$ называется однозначное отображение множества субъектов группового множества S_G во множество уровней безопасности L решетки Λ_L такое, что $F^L(S_G)$ является наибольшей нижней границей уровней безопасностей для множества субъектов s, входящих в S_G .

Функцией уровня безопасности $F^H : S_G \rightarrow L$ называется однозначное отображение множества субъектов группового множества S_G во множество уровней безопасности L решетки Λ_L такое, что $F^H(S_G)$ является наибольшей нижней границей уровней безопасностей для множества субъектов s, входящих в S_G .

Модель совместного доступа

Состояние системы называется безопасным по чтению тогда и только тогда, когда для каждого индивидуального или группового субъекта, осуществляющего в этом состоянии доступ чтения к объекту, уровень безопасности, задаваемый функцией F_L для индивидуального субъекта или функцией F^Lдля группового субъекта, доминирует над уровнем безопасности объекта.

Состояние системы называется безопасным по записи тогда и только тогда, когда для каждого индивидуального или группового субъекта, осуществляющего в этом состоянии доступ записи к объекту, уровень безопасности объекта доминирует над уровнем безопасности субъекта, задаваемого функцией F_L для bндивидуального субъекта или функцией F^H для группового субъекта.

Состояние системы называется безопасным когда оно безопасно по чтению и по записи.

Основные ограничения моделей мандатного доступа

- Все мандатные модели, в основном используют только два права доступа – чтение и запись.
- Невозможность ее применения для сетевых взаимодействий — нельзя построить распределенную систему, в которой информация передавалась бы только в одном направлении, потому что всегда будет существовать обратный поток информации, содержащий ответы на запросы, подтверждения получения и т. д.
- Для оценки возможности нарушений безопасности с использованием методов, основанных на несоответствии этих абстрактных операций и реальных механизмов доступа, применяется анализ т.н. скрытых каналов утечки информации. Чем больше потоков информации мы поставим под контроль мандатной модели, тем менее гибкой будет наша система, но и тем меньше потоков информации придется исследовать в процессе анализа скрытых каналов.
- В реальной жизни она используется только в системах, обрабатывающих классифицированную информацию, и применяется только в отношении ограниченного подмножества субъектов и объектов.

Проблемы использования модели БЛ

Пример 1. Временной канал утечки информации.
Пусть:
F_1 - секретный файл, который может содержать или запись “A" или запись “B";
F_2 - несекретный файл;
S_1 - субъект, работающий по программе:

```
Process S1 (Fi: file):
        Open F1 for read
        While F1 ="A“ Do End
        Close F1
End;
```

S_2 - субъект злоумышленник, работающий по программе:

```
Process S2(F1: file, F2: file):
        Start S1 (F1)
        Wait 10 seconds
        Open F2 for write
                If stop S1 Then
                        Write "B" to F2
                Else
                        Write "A" to F2
                End If
        Close F2
End;
```

Субъект-злоумышленник S_2, не открывая на чтение секретные файлы, запускает процесс S_1 и в зависимости от результатов его выполнения (процесс S_1 либо сразу завершится, либо "зависнет") записывает информацию в несекретный файл.
При этом предполагается, что злоумышленник S_2 имеет уровень доступа секретно, так как S_1 наследует права запустившего его процесса.

Проблемы использования модели БЛ

Пример 2. Каналы утечки информации через локальную и логическую переменные. Пусть F_1 и F_2- секретный и несекретный файлы соответственно. Приведенные ниже процедуры реализуют каналы утечки информации через локальную переменную (процедура P_1) и логическую переменную (процедура P_2).

```
Procedure P1( F1: file, F2: file):
        Open F1 for read
        Read A from F1
        Close
        Open F2 for write
        Write A to F2
        Close F2
End.
```

```
Procedure P2(F1: file, F2: file):
// Считаем, что файл Fi может содержать
//либо запись "A", либо запись"B"
        Open F1 for read
                If F1 - "A" Then
                        Close F1
                        Open F2 for write
                        Write "A" to F2
                Else
                        Close F1
                        Open F2 for write
                        Write "B“ to F2
                End If
        Close F2
End
```

MMS (military message system)-модель

Лендвер, МакЛин, 1984г

Основные элементы системы:

Классификация- обозначение, накладываемое на информацию, отражающее ущерб, который м.б. причинен неавторизованным доступом (TOP SECRET, SECRET, + возможно дополнотельно функциональное разграничение - CRYPTO, NUCLEAR и т.п.)

Пользователь- персона, уполномоченная для использования системы

Степень доверия пользователю- уровень благонадежности персоны (иначе допуск пользователя) - априорно заданная характеристика

Объект- одноуровневый блок информации. Это минимальный блок информации в системе, который может иметь классификацию, т.е. м.б. раздельно от других поименован. Объект не содержит других объектов (т.е. он не многоуровневый)

Контейнер- многоуровневая информационная структура. Имеет классификацию и может содержать объекты (со своей классификацией) и другие контейнеры (также со своей классификацией)

Операция- функция, которая м.б. применена к сущности – объекту или контейнеру (читать, модифицировать и т.д.). Некоторые операции могут использовать более одной сущности (z.b. Copy)

Множество доступа- множество троек (*Пользовательский идентификатор* или *роль* - *Операция* - *Индекс операнда*), которое связано с сущностью (т.е. дескрипторы доступа объекта)

Роль - работа, исполняемая пользователем. Пользователь в любой момент времени (после login до logon) всегда *ассоциирован* как минимум с одной ролью из нескольких. Для действий в данной роли пользователь д.б. *уполномочен*. Некоторые роли в конкркретный момент времени м.б. связаны *только с одним пользователем*. С любой ролью связана способность выполнения определенных *операций.*

.....