Математические структуры и моделирование
2008, вып. 18, с. 98–101

УДК 65.012.810

ОБЪЕКТНО-ОРИЕНТИРОВАННАЯ МОДЕЛЬ КОМПЬЮТЕРНОЙ СИСТЕМЫ

С.В. Белим, С.Ю. Белим

Рассмотрена объектно-ориентированная модель функционирования компьютерной системы. Для построенной модели определена дискреционная политика безопасности.

Введение

Традиционно описание системы безопасности компьютерных систем строится на основе субъектно - объектного подхода. В рамках данного подхода компьютерная система представлена в виде композиции пассивных сущностей, называемых объектами, и активных сущностей, называемых субъектами. Анализ безопасности проводится исходя из рассмотрения доступов субъектов к объектам на основе постулата, сформулированного в «Оранжевой книге» [1]. При этом доказательство безопасности компьютерной системы основывается на соответствии доступов некоторому набору ограничений, называемому политикой безопасности. Следует оговориться, что термин «политика безопасности» в общем случае носит более широкий характер и кроме программных ограничений на доступ включает в себя меры технического и организационного характера по защите информации. Однако при моделировании компьютерной системы рассмотрение политики безопасности можно свести к набору условий на доступ без обсуждения способа их достижения. В дальнейшем на основе рассмотрения доступов в системе строятся модели политик безопасности, позволяющие проводить анализ возможных каналов утечки информации.

В последнее время, в связи с широким распространением объектно-ориентированного подхода в построении программного обеспечения компьютерных систем, возникают трудности в разделении частей компьютерной системы на активные и пассивные. Складывается двойственная ситуация, когда один и тот же набор данных в одном случае интерпретируется как объект (пассивный), в другом случае как субъект (активный). В качестве примера можно привести объект «процесс» в операционной системе Windows. Более того, все объекты современных операционных систем построены на основе объектно-ориентированного подхода и, кроме полей данных, содержат методы их обработки. В связи с этим, для более адекватного описания компьютерных систем,

Омский государственный университет.
E-mail: belim@univer.omsk.su

необходимо применение объектно-ориентированного подхода и соответствующая модификация моделей политик безопасности.

1. Объектно-ориентированная модель компьютерной системы

Будем рассматривать компьютерную систему в виде множества объектов **O**, имеющих открытые поля и скрытые поля, а также методы обработки полей.

Определение 1. Полем будем называть некоторую область памяти фиксированной длины, которая может содержать произвольные данные и изменяться в процессе функционирования системы.

Определение 2. Методом будем называть некоторое отображение, использующее в качестве аргументов поля и не изменяющееся в процессе функционирования компьютерной системы.

Метод может находиться в двух состояниях – активном и неактивном. В начале функционирования системы будем считать, что все методы кроме одного, обычно называемого инициализирующим систему, находятся в неактивном состоянии. Переход в активное состояние происходит через процесс активизации.

Определение 3. Активизацией метода будем называть выделение ему необходимых ресурсов и определение условия передачи ему управления центральным процессорным устройством.

Определенная таким образом система практически совпадает с объектно-субъектным подходом, если установить соответствие между понятиями поле – объект, метод – субъект. Для построения объектно-ориентированной модели необходимо однозначно связать поля и методы.

Определение 4. Классом объектов будем называть произвольный список полей и методов, каждому из которых присвоено одно значение из множества $\{private, public\}$. Поля и методы, которым присвоено значение *private*, называются скрытыми. Поля и методы, которым присвоено значение *public*, называются открытыми.

Определение 5. Совокупность полей и методов, построенных по списку, задаваемому классом, называется объектом класса. Объект класса, для которого несущественна в данной задаче принадлежность к конкретному классу, будем называть объектом.

Как уже было сказано выше, множество всех объектов будем обозначать через **O**. Для каждого объекта $O_i \in \mathbf{O}$ определим множества закрытых полей $O_i.P$, множество открытых полей $O_i.F$ и множество методов $O_i.S$.

Определение 6. Доступом метода s к полю f будем называть активизацию метода s таким образом, что метод s является аргументом соответствующего отображения.

Доступ к скрытым полям объекта имеют только методы этого объекта. Кроме того, методы могут обращаться к открытым полям других объектов.

Определение 7. Возможность доступа к скрытым полям объекта только методами этого же объекта будем называть инкапсуляцией.

Как и в любой другой модели, для построения объекто-ориентированной модели будем использовать ряд предположений, которые не могут быть доказаны в рамках рассматриваемой модели, а обеспечиваются более низкоуровневыми сервисами на этапе построения и поддержки функционирования модели.

Предположение 1. Инкапсуляция реализована корректно, то есть доступ к полям объекта возможен только через вызов методов этого же объекта.

Предположение 2. Активизация метода объекта может быть осуществлена только в процессе доступа к этому методу активного метода этого же либо другого объекта.

Определение 8. Активацию метода объекта методом этого же объекта будем называть автоактивацией.

Определение 9. Два объекта будем считать тождественными, если значения их полей и методов совпадают, как слова, записанные в одном алфавите.

Определение 10. Будем говорить, что объект O_1 осуществляет доступ к объекту O_2 ($O_1 \mapsto O_2$), если существует активный метод $O_1.s_i$, который осуществляет доступ к одному из открытых полей $O_2.f_j$ (прямой доступ к полям объекта), либо активизирует один из методов объекта $O_2.s_k$ (косвенный доступ к полям объекта).

Определение 11. Объект O_1 называется корректным относительно объекта O_2, если O_1 осуществляет только косвенный доступ к полям объекта O_1.

Определение 12. Объекты будем считать взаимно корректными, если они осуществляют только косвенный доступ к полям друг друга.

Определение 13. Между объектами O_1 и O_2 существует поток информации ($O_1 \leftrightarrow O_2$), если $O_1 \mapsto O_2$ или $O_2 \mapsto O_1$.

Утверждение. Если в компьютерной системе все объекты взаимно корректны, то невозможны несанкционированные потоки информации.

Доказательство. Справедливость утверждения следует из Определения 13 и Предположения 1. ■

Определение 14. Под состоянием объекта будем понимать содержимое его полей и список активных методов.

Определение 15. Под состоянием системы будем понимать состояние всех входящих в нее объектов.

2. Дискреционная политика безопасности

Построим дискреционную политику безопасности для объектно ориентированной модели компьютерной системы по аналогии с субъектно - объектной моделью. В субъектно - объектной модели произвольное разграничение доступа строится на основе отображения:

$$M : \mathbf{S} \times \mathbf{O} \to 2^R,$$

где $\mathbf{S}$ – множество субъектов системы, $\mathbf{O}$ – множество объектов системы, R – множество видов доступа. Данное отображение принято записывать в виде

таблицы, строки которой соответствуют субъектам системы, а столбцы объектам системы. Таблицу обычно называют «матрицей доступов», и она является общей для всей системы.

В рамках объектно-ориентированной модели у объектов существует два вида полей – *private* и *public*. К скрытым полям (*private*) доступ осуществляется методами самого объекта, поэтому разрешения на доступ к таким полям сводятся к разрешениям активизации соответствующих методов объекта. Для открытых полей (*public*) будем считать верным следующее предположение.

Предположение 3. Открытые поля всех объектов имеют одно и то же множество возможных типов доступа.

Множество доступов к открытым полям обозначим через $\mathbf{A}$.

В силу того что набор методов работы со скрытыми полями у каждого объекта свой, определение общей матрицы доступов для всей компьютерной системы лишено смысла. Для построения системы дискреционного разделения доступа модифицируем все объекты системы, введя для каждого объекта $O_i \in \mathbf{O}$ дополнительное *private* поле M, содержащее локальную матрицу доступов, и методы работы с матрицей доступов.

$$O_i.M : \mathbf{O} \times (O_i.F \cup O_i.S) \to 2^{\mathbf{A}} \cup \{0, 1\}.$$

Причем

$$O_i.M[O_j, O_i.f] \in 2^{\mathbf{A}}(O_i, O_j \in \mathbf{O}),\ if\ O_i.f \in O_i.F,$$

то есть для открытых полей в явном виде задается множество разрешенных доступов, и

$$O_i.M[O_j, O_i.s] \in \{0, 1\}(O_i, O_j \in \mathbf{O}),\ if\ O_i.s \in O_i.S,$$

то есть для методов определяем разрешение (1) или запрет (0) вызова.

В рамках построенного объектно-ориентированного подхода к описанию системы безопасности компьютерных систем возможно определение элементарных операторов, преобразующих матрицу доступов по аналогии с [2].

Литература

1. Зегжда Д.П., Ивашко А.М. Основы безопасности информационных систем. М.: Горячая линия-Телеком, 2000.
2. Harrison M., Ruzzo W., Ullman J. Protection in operating system // Communication of ACM. 1976. V. 19. P. 461–471.