**Предмет:** Информатика

**Дата проведения:** 20-21-23.04.2020г.

**Преподаватель:** Гаджимагомедов Ш.М

**Группа №**2-13

**Тема урока:** Основы информационной безопасности

**О необходимости объектно-ориентированного подхода к информационной безопасности**

В настоящее время *информационная безопасность* является относительно замкнутой дисциплиной, развитие которой не всегда синхронизировано с изменениями в других областях информационных технологий. В частности, в ИБ пока не нашли отражения основные положения *объектно-ориентированного подхода*, ставшего основой при построении современных информационных систем. Не учитываются в ИБ и достижения в технологии программирования, основанные на накоплении и многократном использовании программистских знаний. На наш взгляд, это очень серьезная проблема, затрудняющая прогресс в области ИБ.

Попытки создания больших систем еще в 60-х годах вскрыли многочисленные проблемы программирования, главной из которых является сложность создаваемых и сопровождаемых систем. Результатами исследований в области технологии программирования стали сначала *структурированное программирование*, затем *объектно-ориентированный подход*.

*Объектно-ориентированный подход* является основой современной технологии программирования, испытанным методом борьбы со сложностью систем. Представляется естественным и, более того, необходимым, стремление распространить этот подход и на системы информационной безопасности, для которых, как и для программирования в целом, имеет *место* упомянутая проблема сложности.

Сложность эта имеет двоякую природу. Во-первых, сложны не только аппаратно-программные системы, которые необходимо защищать, но и сами средства безопасности. Во-вторых, быстро нарастает сложность семейства нормативных документов, таких, например, как профили защиты на основе "*Общих критериев*", речь о которых впереди. Эта сложность менее очевидна, но ею также нельзя пренебрегать; необходимо изначально строить семейства документов *по* объектному принципу.

Любой разумный метод борьбы со сложностью опирается на *принцип "divide et impera" - "разделяй и властвуй"*. В данном контексте этот ***принцип*** означает, что *сложная система* (информационной безопасности) на верхнем уровне должна состоять из небольшого числа относительно независимых *компонентов*. Относительная независимость здесь и далее понимается как *минимизация* числа связей между *компонентами*. Затем *декомпозиции* подвергаются выделенные на первом этапе *компоненты*, и так далее до заданного *уровня детализации*. В результате система оказывается представленной в виде иерархии с несколькими уровнями абстракции.

Важнейший вопрос, возникающий при реализации *принципа "разделяй и властвуй"*, - как, собственно говоря, разделять? Упоминавшийся выше *структурный подход* опирается на алгоритмическую *декомпозицию*, когда выделяются *функциональные элементы* системы. Основная проблема *структурного подхода* состоит в том, что он неприменим на ранних этапах анализа и моделирования *предметной области*, когда до алгоритмов и функций дело еще не дошло. Нужен подход "широкого спектра", не имеющий такого концептуального разрыва с анализируемыми системами и применимый на всех этапах разработки и реализации *сложных систем*. Мы постараемся показать, что *объектно-ориентированный подход* удовлетворяет таким требованиям.

### Основные понятия объектно-ориентированного подхода

*Объектно-ориентированный подход* использует объектную *декомпозицию*, то есть поведение системы описывается в терминах взаимодействия *объектов*.

Что же понимается под *объектом* и каковы другие основополагающие понятия данного подхода?

Прежде всего, введем понятие *класса*. ***Класс*** - это абстракция множества сущностей реального мира, объединенных общностью структуры и поведения.

***Объект*** - это элемент *класса*, то есть абстракция определенной сущности.

Подчеркнем, что *объекты* активны, у них есть не только внутренняя структура, но и поведение, которое описывается так называемыми *методами объекта*. Например, может быть определен *класс* "*пользователь*", характеризующий "пользователя вообще", то есть ассоциированные с пользователями данные и их поведение ( *методы* ). После этого может быть создан *объект* "*пользователь* Иванов" с соответствующей конкретизацией данных и, возможно, *методов*.

К активности *объектов* мы еще вернемся.

Следующую группу важнейших понятий объектного подхода составляют *инкапсуляция*, *наследование* и *полиморфизм*.

Основным инструментом борьбы со сложностью в *объектно-ориентированном подходе* является ***инкапсуляция*** - сокрытие реализации *объектов* (их внутренней структуры и деталей реализации *методов* ) с предоставлением во вне только строго определенных интерфейсов.

Понятие " ***полиморфизм*** " может трактоваться как способность *объекта* принадлежать более чем одному *классу*. Введение этого понятия отражает необходимость смотреть на *объекты* под разными углами зрения, выделять при построении абстракций разные аспекты сущностей моделируемой предметной области, не нарушая при этом целостности *объекта*. (Строго говоря, существуют и другие виды *полиморфизма*, такие как перегрузка и параметрический *полиморфизм*, но нас они сейчас не интересуют.)

***Наследование*** означает построение новых *классов* на основе существующих с возможностью добавления или переопределения данных и *методов*. *Наследование* является важным инструментом борьбы с размножением сущностей без необходимости. Общая информация не дублируется, указывается только то, что меняется. При этом *класс* -потомок помнит о своих "корнях".

Очень важно и то, что *наследование* и *полиморфизм* в совокупности наделяют объектно-ориентированную систему способностью к относительно безболезненной эволюции. Средства информационной безопасности приходится постоянно модифицировать и обновлять, и если нельзя сделать так, чтобы это было экономически выгодно, ИБ из инструмента защиты превращается в обузу.

Мы еще вернемся к механизму *наследования* при рассмотрении *ролевого управления доступом*. Пополним рассмотренный выше классический набор понятий *объектно-ориентированного подхода* еще двумя понятиями: *грани объекта* и *уровня детализации*.

*Объекты* реального мира обладают, как правило, несколькими относительно независимыми характеристиками. Применительно к объектной модели будем называть такие характеристики ***гранями*** . Мы уже сталкивались с тремя основными *гранями* ИБ - доступностью, целостностью и конфиденциальностью. Понятие *грани* позволяет более естественно, чем *полиморфизм*, смотреть на *объекты* с разных точек зрения и строить разноплановые абстракции.

Понятие *уровня детализации* важно не только для визуализации *объектов*, но и для систематического рассмотрения *сложных систем*, представленных в иерархическом виде. Само *по* себе оно очень простое: если очередной уровень иерархии рассматривается с *уровнем детализации* n > 0, то следующий - с *уровнем* (n - 1). *Объект* с *уровнем детализации* 0 считается атомарным.

Понятие *уровня детализации* показа позволяет рассматривать иерархии с потенциально бесконечной высотой, варьировать детализацию как *объектов* в целом, так и их *граней*.

Весьма распространенной конкретизацией *объектно-ориентированного подхода* являются ***компонентные объектные среды***, к числу которых принадлежит, например, *JavaBeans*. Здесь появляется два новых важных понятия: *компонент* и *контейнер*.

Неформально ***компонент*** можно определить как многократно используемый *объект*, допускающий обработку в графическом инструментальном окружении и сохранение в долговременной памяти.

*Контейнеры* могут включать в себя множество *компонентов*, образуя общий контекст взаимодействия с другими *компонентами* и с окружением. *Контейнеры* могут выступать в роли *компонентов* других *контейнеров* .

*Компонентные объектные среды* обладают всеми достоинствами, присущими *объектно-ориентированному подходу*:

* *инкапсуляция* объектных *компонентов* скрывает сложность реализации, делая видимым только предоставляемый вовне интерфейс;
* *наследование* позволяет развивать созданные ранее *компоненты*, не нарушая целостность объектной оболочки;
* *полиморфизм* по сути дает возможность группировать *объекты*, характеристики которых с некоторой точки зрения можно считать сходными.

Понятия же *компонента* и *контейнера* необходимы нам потому, что с их помощью мы можем естественным образом представить защищаемую ИС и сами защитные средства. В частности, *контейнер* может определять границы контролируемой зоны (задавать так называемый "*периметр безопасности*").

Составить краткий конспект

Ответы прощу прислать на почту:mr.shakhrudin@mail.ru

в удобном для вас формате

С указанием группы и ФИО