**Стандарт ISO/IEC 15408 "Критерии оценки безопасности информационных технологий"**

**Основные понятия**

Мы возвращаемся к теме оценочных стандартов, приступая к рассмотрению самого полного и современного среди них - "Критериев *оценки безопасности* информационных технологий" (издан 1 декабря 1999 года). Этот международный стандарт стал итогом почти десятилетней работы специалистов нескольких стран, он вобрал в себя опыт существовавших к тому времени документов национального и межнационального масштаба.

По историческим причинам данный стандарт часто называют "*Общими критериями*" (или даже ОК). Мы также будем использовать это сокращение.

**"Общие критерии"** на самом деле являются метастандартом, определяющим инструменты *оценки безопасности* ИС и порядок их использования. В отличие от "*Оранжевой книги*", ОК не содержат предопределенных *"классов безопасности"*. Такие классы можно строить, исходя из **требований безопасности**, существующих для конкретной организации и/или конкретной информационной системы.

С программистской точки зрения ОК можно считать набором библиотек, помогающих писать содержательные "программы" -**задания по безопасности**, типовые **профили защиты** и т.п. Программисты знают, насколько хорошая библиотека упрощает разработку программ, повышает их качество. Без библиотек, "с нуля", программы не пишут уже очень давно; оценка безопасности тоже вышла на сопоставимый уровень сложности, и "*Общие критерии*" предоставили соответствующий инструментарий.

Важно отметить, что **требования могут быть параметризованы**, как и полагается библиотечным функциям.

Как и "*Оранжевая книга*", ОК содержат два основных вида **требований** безопасности:

* **функциональные**, соответствующие активному аспекту защиты, предъявляемые к функциям безопасности и реализующим их механизмам;
* **требования доверия**, соответствующие пассивному аспекту, предъявляемые к технологии и процессу разработки и эксплуатации.

Требования безопасности предъявляются, а их выполнение проверяется для определенного **объекта оценки** - аппаратно-программного продукта или информационной системы.

Очень важно, что безопасность в ОК рассматривается не статично, а в привязке к жизненному циклу объекта оценки. Выделяются следующие этапы:

* определение назначения, условий применения, целей и требований безопасности;
* *проектирование* и разработка;
* испытания, оценка и сертификация;
* внедрение и эксплуатация.

В ОК объект оценки рассматривается в контексте **среды безопасности**, которая характеризуется определенными условиями и угрозами.

В свою очередь, угрозы характеризуются следующими параметрами:

* *источник угрозы*;
* метод воздействия;
* уязвимые места, которые могут быть использованы;
* ресурсы (активы), которые могут пострадать.

Уязвимые места могут возникать из-за недостатка в:

* требованиях безопасности;
* *проектировании* ;
* эксплуатации.

Слабые места по возможности следует устранить, минимизировать или хотя бы постараться ограничить возможный ущерб от их преднамеренного использования или случайной активизации.

С точки зрения технологии программирования в ОК использован устаревший библиотечный (не объектный) подход. Чтобы, тем не менее, структурировать пространство требований, в "Общих критериях" введена иерархия **класс-семейство-компонент-элемент**.

**Классы** определяют наиболее общую, "предметную" группировку требований (например, функциональные требования *подотчетности* ).

**Семейства** в пределах класса различаются по строгости и другим нюансам требований.

**Компонент** - минимальный набор требований, фигурирующий как целое.

**Элемент** - неделимое требование.

Как и между библиотечными функциями, между компонентами ОК могут существовать зависимости. Они возникают, когда компонент сам по себе недостаточен для достижения **цели безопасности**. Вообще говоря, не все комбинации компонентов имеют смысл, и понятие зависимости в какой-то степени компенсирует недостаточную выразительность библиотечной организации, хотя и не заменяет объединение функций в содержательные объектные интерфейсы.

Как указывалось выше, с помощью библиотек могут формироваться два вида нормативных документов: *профиль защиты* и задание по безопасности.

*Профиль защиты* (ПЗ) представляет собой типовой набор требований, которым должны удовлетворять продукты и/или системы определенного класса (например, операционные системы на компьютерах в правительственных организациях).

Задание по безопасности содержит совокупность требований к конкретной разработке, выполнение которых обеспечивает достижение поставленных *целей безопасности*.

Выше мы отмечали, что в ОК нет готовых *классов защиты*. Сформировать классификацию в терминах "*Общих критериев*" - значит определить несколько иерархически упорядоченных (содержащих усиливающиеся требования) профилей защиты, в максимально возможной степени использующих стандартные функциональные требования и требования **доверия безопасности**.

Выделение некоторого подмножества из всего множества профилей защиты во многом носит субъективный характер. По целому ряду соображений (одним из которых является желание придерживаться объектно-ориентированного подхода) целесообразно, на наш взгляд, сформировать сначала отправную точку классификации, выделив базовый (минимальный) ПЗ, а дополнительные требования компоновать в *функциональные пакеты*.

**Функциональный пакет** - это неоднократно используемая совокупность компонентов, объединенных для достижения определенных *целей безопасности*. "*Общие критерии*" не регламентируют структуру пакетов, процедуры *верификации*, регистрации и т.п., отводя им роль технологического средства формирования ПЗ.

Базовый *профиль защиты* должен включать требования к основным (обязательным в любом случае) возможностям. Производные профили получаются из базового путем добавления необходимых пакетов расширения, то есть подобно тому, как создаются производные классы в объектно-ориентированных языках программирования.

**Функциональные требования**

Функциональные требования сгруппированы на основе выполняемой ими роли или обслуживаемой цели безопасности. Всего в "Общих критериях" представлено 11 функциональных классов, 66 семейств, 135 компонентов. Это, конечно, значительно больше, чем число аналогичных сущностей в "*Оранжевой книге*".

Перечислим классы функциональных требований ОК:

* *идентификация* и *аутентификация* ;
* **защита данных пользователя** ;
* **защита функций безопасности** (требования относятся к целостности и контролю данных *сервисов безопасности* и реализующих их механизмов);
* **управление безопасностью** (требования этого класса относятся к управлению атрибутами и параметрами безопасности);
* **аудит безопасности** (выявление, регистрация, хранение, анализ данных, затрагивающих безопасность объекта оценки, реагирование на возможное нарушение безопасности);
* **доступ к объекту оценки** ;
* **приватность** (защита пользователя от раскрытия и несанкционированного использования его идентификационных данных);
* **использование ресурсов** (требования к доступности информации);
* **криптографическая поддержка** (управление ключами);
* **связь** ( *аутентификация* сторон, участвующих в обмене данными);
* **доверенный маршрут/канал** (для связи с сервисами безопасности).

Опишем подробнее два класса, демонстрирующие особенности современного подхода к ИБ.

Класс "Приватность" содержит 4 семейства функциональных требований.

**Анонимность**. Позволяет выполнять действия без раскрытия идентификатора пользователя другим пользователям, субъектам и/или объектам. Анонимность может быть полной или выборочной. В последнем случае она может относиться не ко всем операциям и/или не ко всем пользователям (например, у уполномоченного пользователя может оставаться возможность выяснения идентификаторов пользователей).

**Псевдонимность**. Напоминает анонимность, но при применении псевдонима поддерживается ссылка на идентификатор пользователя для обеспечения *подотчетности* или для других целей.

**Невозможность ассоциации**. Семейство обеспечивает возможность неоднократного использования *информационных сервисов*, но не позволяет ассоциировать случаи использования между собой и приписать их одному лицу. Невозможность ассоциации защищает от построения профилей поведения пользователей (и, следовательно, от получения информации на основе подобных профилей).

**Скрытность**. Требования данного семейства направлены на то, чтобы можно было использовать информационный сервис с сокрытием факта использования. Для *реализации* *скрытности* может применяться, например, широковещательное распространение информации, без указания конкретного адресата. Годятся для *реализации* *скрытности* и методы стеганографии, когда скрывается не только содержание сообщения (как в криптографии), но и сам факт его отправки.

Еще один показательный (с нашей точки зрения) класс функциональных требований - "Использование ресурсов", содержащий требования доступности. Он включает три семейства.

**Отказоустойчивость**. Требования этого семейства направлены на сохранение доступности *информационных сервисов* даже в случае сбоя или отказа. В ОК различаются активная и пассивная отказоустойчивость. Активный механизм содержит специальные функции, которые активизируются в случае сбоя. Пассивная отказоустойчивость подразумевает наличие избыточности с возможностью нейтрализации ошибок.

**Обслуживание по приоритетам**. Выполнение этих требований позволяет управлять использованием ресурсов так, что низкоприоритетные операции не могут помешать высокоприоритетным.

**Распределение ресурсов**. Требования направлены на защиту (путем применения механизма *квот*) от несанкционированной монополизации ресурсов.

Мы видим, что "*Общие критерии*" - очень продуманный и полный документ с точки зрения функциональных требований. В то же время, хотелось бы обратить внимание и на некоторые недостатки.

Первый мы уже отмечали - это отсутствие объектного подхода. Функциональные требования не сгруппированы в осмысленные наборы (объектные интерфейсы), к которым могло бы применяться наследование. Подобное положение, как известно из технологии программирования, чревато появлением слишком большого числа комбинаций функциональных компонентов, несопоставимых между собой.

В современном программировании ключевым является вопрос накопления и многократного использования знаний. Стандарты - одна из форм накопления знаний. Следование в ОК "библиотечному", а не объектному подходу сужает круг фиксируемых знаний, усложняет их корректное использование.

К сожалению, в "Общих критериях" отсутствуют *архитектурные требования*, что является естественным следствием избранного старомодного программистского подхода "снизу вверх". На наш взгляд, это серьезное упущение. Технологичность средств безопасности, следование общепризнанным рекомендациям по протоколам и программным интерфейсам, а также апробированным архитектурным решениям, таким как менеджер/агент, - необходимые качества изделий информационных технологий, предназначенных для поддержки критически важных функций, к числу которых, безусловно, относятся функции безопасности. Без рассмотрения *интерфейсных аспектов* системы оказываются нерасширяемыми и изолированными. Очевидно, с практической точки зрения это недопустимо. В то же время, обеспечение *безопасности интерфейсов* - важная задача, которую желательно решать единообразно.

**Требования доверия безопасности**

Установление доверия безопасности, согласно "*Общим критериям*", основывается на активном исследовании объекта оценки.

Форма *представления требований* доверия, в принципе, та же, что и для функциональных требований. Специфика состоит в том, что каждый элемент требований доверия принадлежит одному из трех типов:

* действия **разработчиков** ;
* представление и содержание **свидетельств** ;
* действия **оценщиков**.

Всего в ОК 10 классов, 44 семейства, 93 компонента требований доверия безопасности. Перечислим классы:

* разработка (требования для поэтапной детализации функций безопасности от краткой спецификации до *реализации* );
* поддержка жизненного цикла (требования к *модели жизненного цикла*, включая порядок *устранения недостатков* и защиту среды разработки);
* *тестирование* ;
* **оценка уязвимостей** (включая оценку *стойкости функций безопасности*);
* **поставка и эксплуатация** ;
* управление конфигурацией;
* руководства (требования к эксплуатационной документации);
* **поддержка доверия** (для поддержки этапов жизненного цикла после сертификации);
* оценка *профиля защиты*;
* оценка задания по безопасности.

Применительно к *требованиям доверия* в "Общих критериях" сделана весьма полезная вещь, не реализованная, к сожалению, для функциональных требований. А именно, введены так называемые оценочные уровни доверия (их семь), содержащие осмысленные комбинации компонентов.

*Оценочный уровень доверия* 1 (начальный) предусматривает анализ **функциональной спецификации**, спецификации интерфейсов, эксплуатационной документации, а также независимое *тестирование*. Уровень применим, когда угрозы не рассматриваются как серьезные.

*Оценочный уровень доверия* 2, в дополнение к первому уровню, предусматривает наличие **проекта верхнего уровня** объекта оценки, выборочное независимое *тестирование*, *анализ стойкости функций безопасности*, поиск разработчиком явных уязвимых мест.

На третьем уровне ведется контроль среды разработки и управление конфигурацией объекта оценки.

На уровне 4 добавляются полная спецификация интерфейсов, **проекты нижнего уровня**, анализ подмножества *реализации*, применение неформальной **модели** *политики безопасности*, независимый *анализ уязвимых* мест, автоматизация управления конфигурацией. Вероятно, это самый высокий уровень, которого можно достичь при существующей технологии программирования и приемлемых затратах.

Уровень 5, в дополнение к предыдущим, предусматривает применение *формальной модели* *политики безопасности*, полуформальных *функциональной спецификации* и *проекта верхнего уровня* с **демонстрацией соответствия** между ними. Необходимо проведение *анализа скрытых каналов* разработчиками и оценщиками.

На уровне 6 *реализация* должна быть представлена в структурированном виде. Анализ соответствия распространяется на проект нижнего уровня.

Оценочный уровень 7 (самый высокий) предусматривает формальную *верификацию* проекта объекта оценки. Он применим к ситуациям чрезвычайно высокого риска.