О проблеме представления формальной модели политики безопасности операционных систем

д.т.н., доцент Девянин П.Н.
ФУМО ВО ИБ, г. Москва

Нормативные документы ФСТЭК России

ИНФОРМАЦИОННОЕ СООБЩЕНИЕ
об утверждении Требований безопасности информации к операционным системам

от 18 октября 2016 г. № 240/24/4893

В соответствии с подпунктом 13.1 пункта 8 Положения о Федеральной службе по техническому и экспортному контролю, утвержденного Указом Президента Российской Федерации от 16 августа 2004 г. № 1085, приказом ФСТЭК России от 19 августа 2016 г. № 119 (зарегистрирован Минюстом России 19 сентября 2016 г., регистрационный № 43691) утверждены Требования безопасности информации к операционным системам (далее – Требования), которые вступают в силу с 1 июня 2017 г.

ИТ.ОС.А2.ПЗ

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ТЕХНИЧЕСКОМУ И ЭКСПОРТНОМУ КОНТРОЛЮ
(ФСТЭК РОССИИ)

Утверждён ФСТЭК России
8 февраля 2017 г.

МЕТОДИЧЕСКИЙ ДОКУМЕНТ

**ПРОФИЛЬ ЗАЩИТЫ
ОПЕРАЦИОННЫХ СИСТЕМ ТИПА «А»
ЧЕТВЕРТОГО КЛАССА ЗАЩИТЫ**

ИТ.ОС.А4.ПЗ

Требование доверия	Руководящие документы, 1992 г. и 1997 г.	Требования, 2016 г.
Полное независимое тестирование	**Имеется**	**Имеется**
Анализ уязвимостей	**Отсутствует**	**Имеется**
Верификация формальной модели	**Частично**	**Имеется**

ГОСТ Р ИСО/МЭК 15408-3—2013

11.5.4 ADV_SPM.1 Формальная модель политики безопасности ОО
Зависимости: ADV_FSP.4 Полная функциональная спецификация
11.5.4.1 Элементы действий разработчика
11.5.4.1.1 ADV_SPM.1.1D
Разработчик должен представить формальную модель ПБО для [назначение: *список формально моделируемых политик*].

ОУД5+

Уровень доверия: Оценочный уровень доверия 3 (ОУД3), усиленный компонентами ADV_FSP.4 «Полная функциональная спецификация», ADV_IMP.2 «Полное отображение представления реализации ФБО», ADV_TDS.3 «Базовый модульный проект», ALC_CMC.4 «Поддержка генерации, процедуры приемки и автоматизация», ALC_FLR.1 «Базовое устранение недостатков», ALC_TAT.1 «Полностью определенные инструментальные средства разработки», AVA_VAN.5 «Усиленный методический анализ», расширенный компонентами ADV_IMP_EXT.3 «Реализация ОО», ALC_FPU_EXT.1 «Процедуры обновления программного обеспечения операционной системы», ALC_LCD_EXT.3 «Определенные разработчиком сроки поддержки», AMA_SIA_EXT.3 «Анализ влияния обновлений на безопасность операционной системы», AMA_SIA_EXT.6 «Анализ влияния внешних модулей уровня ядра на безопасность операционной системы» и AVA_CCA_EXT.1 «Анализ скрытых каналов».

«Глубина» проработки представления модели

ГОСТ Р ИСО/МЭК 18045—2013

10.7 Моделирование политики безопасности (ADV_SMP)

10.7.1 Подвид деятельности по оценке (ADV_SPM.1)

Общее руководство отсутствует; за консультациями по выполнению данного подвида деятельности следует обращаться в конкретную систему оценки.

?

Пример 1: Модель – кортеж (V, T)

V – множество состояний системы, задающее доступы (текущие доступы или права доступа) субъектов из множества S к объектам из множества O;
T – функция переходов системы из состояния в состояние

?

Пример 2: Модель Белла-ЛаПадулы (1976)

S – множество субъектов; O – множество объектов;
$R = \{read, write, append, execute\}$ – множество видов доступа;
$B = \{b \subseteq S \times O \times R\}$ – множество возможных текущих доступов;
$(L, \leq)$ – решётка уровней конфиденциальности;
$M = \{m_{|S| \times |O|}\}$ – множество возможных матриц доступов;
$(f_s, f_o, f_c) \in F = L^s \times L^o \times L^s$ – тройка функций уровня доступа, текущего уровня доступа субъектов и уровня конфиденциальности объектов;
$V = B \times M \times F$ – множество состояний системы;
Q – множество запросов; D – множество ответов по запросам;
$W \subseteq Q \times D \times V \times V$ – множество действий системы;
$\Sigma(Q, D, W, z_0)$ – система;
ss-свойство, *-свойство, *ds*-свойство;
Теорема БТБ. Система $\Sigma(Q, D, W, z_0)$ безопасна для безопасного z_0 тогда и только тогда, когда множество действий системы W удовлетворяет условиям теорем $A1$-$A3$.

Требования к представлению модели (1)

Представление описаний:

- Множеств учётных записей пользователей, субъектов, объектов (сущностей), устанавливающих классификацию элементов этих множеств, связи между этими множествами или внутри них функций (отношений), заданных на этих множествах отношений иерархии;
- Множеств реализуемых прав доступа и доступов субъектов к сущностям, используемых для задания прав доступа и доступов (непосредственно, с использованием групп, ролей, типов, атрибутов) множеств, функций (отношений);
- Решётки уровней целостности, используемых для задания уровней целостности учётных записей пользователей, субъектов, сущностей функций (отношений);
- Решётки уровней конфиденциальности (при необходимости), используемых для задания уровней доступа учётных записей пользователей и субъектов, уровней конфиденциальности сущностей функций (отношений);
- Множеств, функций (отношений), используемых для задания сущностей, функционально ассоциированных с доверенными субъектами или параметрически ассоциированных с учётными записями пользователей;
- Множеств, функций (отношений), используемых для задания сущностей-контейнеров, доступ к содержащимся в которых сущностях субъектами может быть разрешён без учёта уровней целостности или без учёта уровней конфиденциальности (при необходимости) таких сущностей-контейнеров;
- Элементов состояний, моделирующей ОС абстрактной системы, используемых для этого множеств, функций (отношений);

Требования к представлению модели (2)

Представление описаний (продолжение):

- Условий предоставления субъектам прав доступа и доступов к сущностям или субъектам и условий выполнения иных правил преобразования состояний (команд, операций, функций перехода) над учётными записями пользователей, субъектами и сущностями (создание, удаление, переименование, получение параметров), заданных для этого специальных элементов ОС (привилегией, ролей, административных ролей);
- Условий возникновения информационных потоков, за счёт реализации субъектами доступов к сущностям или субъектами, или получения субъектами контроля над другими субъектами;
- Условий получения субъектами контроля над другими субъектами за счёт использования сущностей, функционально ассоциированными с субъектами или параметрически ассоциированными с учётными записями пользователей, и информационных потоков между ними;
- Правил преобразования состояний (команд, операций, функций перехода), моделирующей ОС абстрактной системы, включая параметры каждого правила, условия и результаты его применения. Как минимум должны быть описаны: правила администрирования (создания, удаления, переименования, изменения прав доступа, уровней целостности, доступа или конфиденциальности (при необходимости), получения параметров) учётных записей пользователей, субъектов и сущностей; правила предоставления доступов субъектов к сущностям и субъектам; правила создания информационных потоков и получения субъектами контроля над другими субъектами;

Требования к представлению модели (3)

Представление описаний (продолжение):

- Доказательства выполнения при применении (корректности задания) правил преобразования состояний (команд, операций, функций перехода), моделирующей ОС абстрактной системы: условий предоставления субъектам прав доступа и доступов к сущностям или субъектам; условий выполнения иных правил преобразования состояний (команд, операций, функций перехода) над учётными записями пользователей, субъектов и сущностей (создание, удаление, переименование, получение параметров);
- Доказательства в рамках моделирующей ОС абстрактной системы того, что реализованный мандатный контроль целостности позволяет обеспечить защиту от несанкционированного изменения субъектом-нарушителем параметров или данных в сущностях, параметров или функциональности субъектов (захватить контроль над субъектом) с более высоким, чем у него уровнем целостности, и в результате нарушить целостность программно-аппаратной среды ОС;
- При необходимости реализации мандатного управления доступом доказательства в рамках моделирующей ОС абстрактной системы того, что реализованные мандатные контроль целостности и управление доступом позволяют обеспечить защиту от запрещённых информационных потоков (как минимум по памяти) от сущностей с более высоким уровнем конфиденциальности к сущностям с более низким уровнем конфиденциальности (защиту от информационных потоков «сверху-вниз»).

Требования к верификации модели (1)

Представление описаний:

- Основных функциональных возможностей, формализованного языка и порядка применения использованных для верификации модели политики безопасности управления доступом инструментальных средств;
- Представления модели политики безопасности управления доступом с использованием формализованного языка инструментальных средств верификации. При этом на формализованном языке должны быть выражены: элементы состояний, моделирующей ОС абстрактной системы, используемые для этого множества, функции (отношения); правила преобразования состояний (команды, операции, функции перехода), включая параметры каждого правила, условия и результаты его применения; условия выполнения мандатного контроля целостности и мандатного управления доступом (при необходимости);
- Если формализованный язык инструментальных средств не может точно выразить некоторые элементы модели политики безопасности управления доступом, то описание всех таких элементов и полуформальное обоснование того, что это не влияет на итоговый результат верификации;
- Результатов верификации модели политики безопасности управления доступом с использованием инструментальных средств верификации с указанием того, какие элементы модели были верифицированы в автоматическом режиме, а какие в полуавтоматическом (ручном) режиме;

Требования к верификации модели (2)

Представление описаний (продолжение):

- С применением инструментальных средств результатов верификации выполнения при применении (корректности задания) правил преобразования состояний (команд, операций, функций перехода), моделирующей ОС абстрактной системы: условий предоставления субъектам прав доступа и доступов к сущностям или субъектам; условий выполнения иных правил преобразования состояний (команд, операций, функций перехода) над учётными записями пользователей, субъектами и сущностями (создание, удаление, переименование, получение параметров);
- С применением инструментальных средств результатов верификации в рамках моделирующей ОС абстрактной системы того, что реализованный мандатный контроль целостности позволяет обеспечить защиту от несанкционированного изменения субъектом-нарушителем параметров или данных в сущностях, параметров или функциональности субъектов (захватить контроль над субъектом) с более высоким, чем у него уровнем целостности, и в результате нарушить целостность программно-аппаратной среды ОС;
- При необходимости с применением инструментальных средств результатов верификации в рамках моделирующей ОС абстрактной системы того, что реализованные мандатные контроль целостности и управление доступом позволяют обеспечить защиту от запрещённых информационных потоков (как минимум по памяти) от сущностей с более высоким уровнем конфиденциальности к сущностям с более низким уровнем конфиденциальности (защиту от утечки конфиденциальных данных, от информационных потоков «сверху-вниз»).

Пример выполнения требований ADV_SPM.1

Математическая нотация МРОСЛ ДП-модели

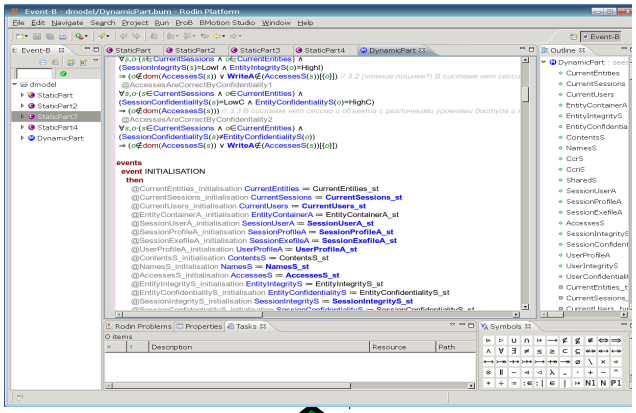
access_read(x, x', y)

$x, x' \in S, y \in E \cup R \cup AR$, существует $r \in R \cup AR$: $(x, r, read_a) \in AA$, [если $y \in E$, то $(y, read_r) \in PA(r)$ и либо $(execute\_container(x, y) = true$ и $f_e(y) \le f_s(x))$, либо $(x, downgrade\_admin\_role, read_a) \in AA$], [если $y \in R \cup AR$, то $(y, read_r) \in APA(r)$, $i_r(y) \le i_s(x)$, $Constraint_{AA}(AA') = true$, (для $e \in]y[$ либо $(x, e, read_a) \in A$, либо $(x, e, write_a) \in A$), (либо $f_r(y) \le f_s(x)$, либо $(x, downgrade\_admin\_role, read_a) \in AA$)], [если $y \in R \cup AR$ и $i_r(y) = i\_high$, то $(x', f_s(x)\_i\_entity, write_a) \in A$]	$S' = S$, $E' = E$, $APA' = APA$, $PA' = PA$, $user' = user$, $H_E' = H_E$, $F' = F$, если $y \in E$, то $[A' = A \cup \{(x, y, read_a)\}$, $AA' = AA]$, если $y \in R \cup AR$, то $[AA' = AA \cup \{(x, y, read_a)\}$, $A' = A]$

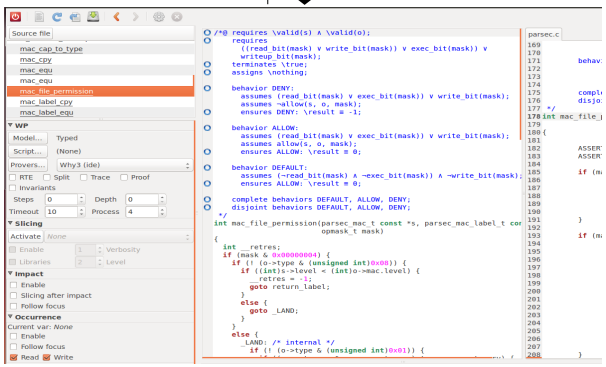
```
1  static int access_read(struct task_struct *subject,struct inode *entity)
2  {
3      struct task_security *subject_security;
4      struct inode_security *entity_security;
5      int ret = -EACCES;
6
7      subject_security = get_task_security(subject);
8      entity_security = get_entity_security(entity);
9
10     if(!subject_security || !entity_security)
11         return ret;
12
13     if(is_role(entity)) //проверяем возможность получения доступа на чтение к сущности
14     {
15         ret = can_access(&subject_security->roles, &entity_security->list, MAY_READ, 0);
16         if(ret != 0)
17             ret = can_access(&subject_security->admin_roles, &entity_security->list, MAY_WRITE, 0);
18         if(ret == 0)
19         {
20             ret = execute_container(subject, entity);
21             if(ret != 0)
22                 ret = is_downgrade_admin_role(&subject_security->admin_roles, MAY_READ);
23         }
24     }
25     else //проверяем возможность получения доступа на чтение к роли или административной роли
26     {
27         ret = can_admin_access(&subject_security->admin_roles, &entity_security->list, MAY_READ);
28     }
29     return ret;
```

Программный код механизма управления доступом ОССН

Автоматизированная дедуктивная верификация формализованной МРОСЛ ДП-модели



Rodin (Event-B), Alloy



Frama-C, Why3

Верификации кода на соответствие спецификациям

Спасибо за внимание!