

Министерство образования и науки Российской Федерации
Институт механики сплошных сред УрО РАН

На правах рукописи

УДК 004.4

СОЗЫКИН Андрей Владимирович

СЕМАНТИЧЕСКАЯ ИНТЕГРАЦИЯ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ

Специальность 05.13.11 «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей»

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель
кандидат технических наук
старший научный сотрудник
Г.Ф. Масич

ПЕРМЬ – 2008

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	2
ВВЕДЕНИЕ.....	5
1 ОБЗОР ТЕХНОЛОГИЙ ИНТЕГРАЦИИ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ.....	10
1.1 Технологии управления доступом к сервисам	10
1.1.1 Модель управления доступом к сервисам	10
1.1.2 Идентификация и аутентификация.....	11
1.1.2.1 Типовая схема идентификации и аутентификации.....	11
1.1.2.2 Парольная аутентификация	13
1.1.2.3 Многофакторная аутентификация	13
1.1.2.4 Биометрическая аутентификация.....	14
1.1.2.5 Аутентификация с использованием криптографии	15
1.1.2.6 Аутентификация с нулевой передачей знаний	19
1.1.3 Однократная регистрация	21
1.1.4 Авторизация	21
1.1.4.1 Дискреционное управление доступом.....	21
1.1.4.2 Мандатное управление доступом	24
1.1.4.3 Ролевое управление доступом.....	25
1.1.4.4 Атрибутное управление доступом	26
1.1.5 Делегирование.....	27
1.1.6 Анализ состояния технологий в области управления доступом к сервисам	27
1.2 Методы интеграции информационных систем	28
1.3 Существующие системы интеграции управления доступом к сервисам.....	30
1.3.1 Традиционный подход	30
1.3.2 Централизованное управление.....	31
1.3.3 Федеративная идентификация.....	34
1.3.4 Интегрированное управление доступом к сервисам	34
1.3.5 Оценка существующих систем интеграции управления доступом к сервисам	35
1.4 Онтологии предметной области управления доступом к сервисам.....	36
1.4.1 Специализированные онтологии управления доступом	36
1.4.2 Онтологии верхнего уровня	37
1.4.3 Оценка онтологий в области управления доступом к сервисам	38
1.5 Официальные документы в области управления доступом.....	39

1.6	Выводы по главе	42
2	СЕМАНТИЧЕСКАЯ ИНТЕГРАЦИЯ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ	44
2.1	Анализ и критика прототипа	44
2.1.1	Анализ прототипа	44
2.1.2	Недостатки прототипа	45
2.1.3	Предлагаемое решение	46
2.2	Методика интеграции управления доступом к сервисам на основе семантического подхода	47
2.3	Выводы по главе	49
3	СИСТЕМА МОДЕЛЕЙ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ	50
3.1	Онтология управления доступом к сервисам	50
3.1.1	Структура системы управления доступом к сервисам	50
3.1.1.1	Объекты	50
3.1.1.2	Свойства	52
3.1.2	Динамика управления доступом	53
3.1.2.1	Состояния	53
3.1.2.2	События	54
3.1.2.3	Операции	55
3.1.3	Диаграмма состояний системы управления доступом к сервисам	57
3.2	Алгебраическая запись правил разграничения доступа к сервисам	58
3.2.1	Базовый уровень	59
3.2.2	Контейнеры	62
3.2.3	Роли	64
3.2.4	Делегирование полномочий управления доступом к сервисам	66
3.2.5	Применение методов авторизации в алгебраической записи	68
3.3	Выводы по главе	69
4	РЕАЛИЗАЦИЯ И ПРИМЕНЕНИЕ РАЗРАБОТАННЫХ МЕТОДОВ И ТЕХНОЛОГИЙ	71
4.1	Комплекс программ по управлению доступом к сервисам	71
4.1.1	Функции комплекса программ	71
4.1.2	Уровни интеграции	71
4.1.3	Логическая архитектура	72
4.1.3.1	Уровень технических служб	73
4.1.3.2	Уровень приложений	74
4.1.3.3	Уровень представления	74
4.1.4	Взаимодействие сервисов с комплексом программ	75
4.1.5	Архитектура развертывания	76

4.1.6	Реализация.....	77
4.1.6.1	Репозиторий правил разграничения доступа.....	77
4.1.6.2	Адаптеры сервисов.....	78
4.1.6.3	Консоль управления доступом к сервисам.....	78
4.2	Практическое применение.....	81
4.2.1	Сеть Пермского научного центра.....	81
4.2.2	Сервисы сети Пермского научного центра.....	82
4.2.3	Схема реализации системы управления доступом к сервисам.....	82
4.2.4	Интеграция с системой статистики использования сервисов.....	83
4.3	Оценка эффективности использования системы семантической интеграции управления доступом к сервисам.....	84
4.3.1	Удобство работы с сервисами.....	84
4.3.2	Сокращение затрат на управления доступом.....	85
4.4	Выводы по главе.....	86
ЗАКЛЮЧЕНИЕ.....		87
ПРИЛОЖЕНИЕ 1. ОТОБРАЖЕНИЕ ПОНЯТИЙ ФОРМАЛЬНОЙ АЛГЕБРАИЧЕСКОЙ ЗАПИСИ ПРАВИЛ РАЗГРАНИЧЕНИЯ ДОСТУПА В LDAP.....		93
ПРИЛОЖЕНИЕ 2. АКТЫ ВНЕДРЕНИЯ.....		94
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....		96
СПИСОК ИЛЛЮСТРАЦИЙ.....		105
СПИСОК ТАБЛИЦ.....		107

ВВЕДЕНИЕ

Актуальность темы. Современные научные и коммерческие организации строят корпоративные сети, предоставляющие сотрудникам сервисы разных типов: сетевые (электронная почта, доступ в Интернет), вычислительные (кластеры, многопроцессорные серверы), информационные (справочные системы, порталы, системы управления предприятием, прикладные научные системы). В таких сетях повышенное внимание уделяется управлению доступом к сервисам в целях обеспечения безопасности и удобства работы.

В крупных сетях с большим количеством сервисов и пользователей, управление доступом к сервисам связано с рядом проблем. Управление является трудоемкой задачей, создающей большую нагрузку на администраторов и предъявляющей высокие требования к их квалификации. Для управления доступом к сервисам разных типов приходится использовать несколько разных систем управления и выполнять большое число операций. Сервисы разных типов используют отдельные репозитории правил разграничения доступа, часть информации в которых дублируется и требует синхронизации при изменении. Пользователям работать с сервисами неудобно из-за большого количества идентификаторов и паролей, требуемых для разрешения доступа к сервисам. Безопасность работы с сервисами находится на низком уровне: пользователи выбирают простые пароли, которые легко подобрать, сложные пароли записывают, нетрудно реализуются атаки социальных инженеров.

Актуальной является задача повышения эффективности процесса управления доступом и удобства работы с сервисами путем интеграции механизмов управления доступом к сервисам разных типов. Интеграция осложняется тем, что сервисы используют различные протоколы управления доступом: RADIUS, KERBEROS, LDAP, SAML, WS-Security и др. Для хранения правил разграничения доступа применяются различные репозитории: текстовые файлы, XML, реляционные СУБД, каталоги LDAP. В последнее время все большей популярностью пользуются интегрированные системы, позволяющие управлять доступом

к сервисам разных типов, независимо от деталей взаимодействия. При этом расхождения в базовой семантической модели этих систем приводят к проблемам интероперабельности и существенно ограничивают круг поддерживаемых сервисов.

В работе предложена формальная основа для интеграции механизмов управления доступом к сервисам разных типов на основе семантического подхода и разработан комплекс программ для интеграции управления доступом к сервисам разных типов. Данный комплекс обеспечивает хранение правил разграничения доступа, реализацию процедур защиты информации (идентификация, аутентификация и авторизация), предоставляет единую систему управления правилами разграничения доступа для всех сервисов, используемых в организации, независимо от их типа.

Целью диссертационной работы является повышение эффективности процесса управления доступом к сервисам за счет интеграции механизмов управления доступом к сервисам разных типов (информационным, сетевым и вычислительным). В работе исследованы и решены следующие **задачи**:

1. Исследование и сравнительный анализ существующих подходов к управлению доступом с точки зрения возможности интеграции управления доступом к сервисам разных типов.
2. Разработка методики семантической интеграции управления доступом к сервисам разных типов.
3. Построение семантической модели системы управления доступом к сервисам.
4. Создание средств описания правил разграничения доступа к сервисам в формальном виде.
5. Разработка комплекса программ, интегрирующего управление доступом к сервисам разных типов.
6. Исследование эффективности применения семантической интеграции управления доступом к сервисам в сетях научных организаций.

Объект исследования: процесс управления доступом к сервисам.

Предмет исследования: интеграция управления доступом к сервисам разных типов.

Научная новизна. В диссертационной работе получены следующие новые результаты:

- Применен семантический подход для интеграции управления доступом к сервисам разных типов, что позволило значительно расширить интероперабельность. Разработана методика интеграции управления доступом к сервисам на основе семантического подхода.

- Предложена унифицированная онтологическая модель, определяющая базовые понятия и операции управления доступом к сервисам. Модель делает возможной интеграцию управления доступом к сервисам разных типов, использующих различные модели, методы и технологии управления доступом на основе семантического подхода.

- Разработана алгебраическая запись правил разграничения доступа, представляющая собой формальную запись понятий разработанной онтологической модели. Алгебраическая запись позволяет в формальном виде описывать правила разграничения доступа с использованием различных методов управления доступом.

- Реализован комплекс программ управления доступом к сервисам на основе разработанных технологий.

На защиту выносятся:

1. Методика семантической интеграции управления доступом к сервисам, позволяющая значительно расширить круг поддерживаемых сервисов и методов управления доступом.

2. Система моделей управления доступом к сервисам, включающая онтологическую модель управления доступом к сервисам, и алгебраическую запись правил разграничения доступа. Модели предоставляют основу для интеграции управления доступом к сервисам разных типов: онтология задает общую семантику понятий предметной области управления доступом и операций

над ними, общий формальный синтаксис задает алгебраическая запись правил разграничения доступом.

3. Результаты оценки эффективности применения семантической интеграции управления доступом к сервисам в сетях научных организаций.

Практическая ценность. Разработанные модели, методы, технологии и созданный на их основе комплекс программ позволяют интегрировать управление доступом к сервисам разных типов. Работа пользователей с сервисами становится более удобной за счет интеграции учетных записей для доступа ко всем сервисам с возможностью однократной регистрации. Администраторам, отвечающим за управление доступом к сервисам, предоставляется единая, удобная, интуитивно понятная система управления.

Краткое содержание работы

В главе 1 рассмотрены существующие технологии управления доступом к сервисам. Описаны методы интеграции информационных систем, выполнен обзор существующих систем интеграции управления доступом к сервисам. Описаны существующие онтологии и стандарты в области управления доступом к сервисам.

В главе 2 представлен метод интеграции управления доступом к сервисам на основе семантического подхода. Разработана методика семантической интеграции управления доступом к сервисам.

В главе 3 предложена система моделей управления доступом к сервисам, состоящая из онтологии управления доступом и алгебраической записи правил разграничения доступа. Система моделей служит основой для семантической интеграции управления доступом к сервисам, предложенной во второй главе.

В главе 4 описан комплекс программ, реализующий семантическую интеграцию управления доступом к сервисам. Рассмотрено внедрение комплекса в сети Пермского научного центра Уральского отделения Российской Академии Наук (ПНЦ УрО РАН). Выполнен анализ эффективности использования семантической интеграции управления доступом к сервисам разных типов.

В заключении приведены основные результаты диссертационной работы.

Приложение содержит детали реализации и акты внедрения.

1 ОБЗОР ТЕХНОЛОГИЙ ИНТЕГРАЦИИ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ

1.1 Технологии управления доступом к сервисам

1.1.1 Модель управления доступом к сервисам

Управление доступом (access control) – это процесс проверки запросов на доступ к сервису с целью определения разрешить или запретить доступ [1]. Большинство современных систем управления доступом строится на основе модели, предложенной Лампсоном в работе [2] (рис 1.1).

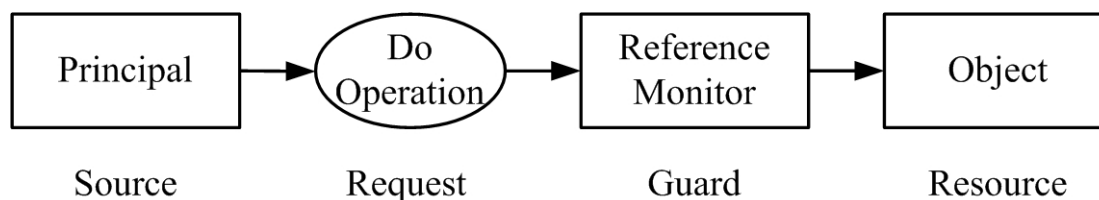


Рис 1.1. Модель управления доступом Лампсона [2]

Модель Лампсона включает следующие элементы:

- **Principal** – автор запроса (принципал).
- **Request** – запрос на выполнение операции с объектом.
- **Reference Monitor** - диспетчер доступа, проверяющий все запросы к объекту и принимающий решение о разрешении или запрещении доступа.
- **Object** – ресурс информационный, сетевой или вычислительный.

В некоторых системах диспетчер доступа может не выделяться. Но использование диспетчера значительно упрощает управление доступом [3].

В качестве принципала могут выступать [4]:

- Люди и вычислительные машины.
- Объединение принципалов. Возможен случай, когда операцию разрешено выполнять принципалам А и В совместно, но не по отдельности принципалу А или принципалу В.
- Группы. Часто неудобно задавать всех принципалов, имеющих право выполнять некоторые операции с объектом, так как их слишком много или они

часто меняются. Группы предоставляют удобный механизм непрямого назначения прав доступа.

- Принципал в некоторой роли.
- Принципал от имени другого принципала (делегирование).

Диспетчер доступа принимает решение на основе того, какой принципал запрашивает доступ, какую операцию необходимо выполнить с объектом и правил разграничения доступа, определяющих, какие операции может принципал выполнять с объектом. Для принятия решения диспетчеру доступа необходимо выяснить, кто выполняет запрос и выполнить проверку правил разграничения доступа. Определение источника запроса называется **идентификацией**, подтверждение подлинности источника называется **аутентификацией**, анализ правил разграничения доступа называется **авторизацией**. Таким образом, идентификация дает ответ на вопрос «Кто запрашивает доступ», аутентификация – на вопрос «Тот ли это, за кого себя выдает», авторизация – на вопрос «Кому разрешен доступ к объекту». Далее рассмотрены распространенные реализации этих процедур.

1.1.2 Идентификация и аутентификация

1.1.2.1 Типовая схема идентификации и аутентификации

Рассмотрим типовую схему идентификации и аутентификации, применяемую во многих системах, возможно, с некоторыми модификациями [5]. Для каждого субъекта доступа существует аутентифицирующий объект, представляющий собой пару (ID_i, K_i) :

- ID_i – идентификатор, позволяющий однозначно выделить i -й субъект доступа из множества всех субъектов, зарегистрированных в системе.
- K_i – аутентифицирующая информация, подтверждающая подлинность i -го субъекта доступа.

В целях безопасности аутентифицирующий объект не хранится в системе в открытом виде, вместо этого используется объект-эталон, хранящий данные в защищенном формате. Объект-эталон представляет собой пару (ID_i, E_i) , где

$E_i = F(ID_i, K_i)$. Трудоемкость определения K_i по E_i должна быть выше некоторого порогового значения T_0 . Для пары K_i и K_j возможно совпадение соответствующих значений E , что может привести к ложной аутентификации с некоторой вероятностью P_0 . Для практического применения задают $T_0 = 10^{20} \dots 10^{30}$, $P_0 = 10^{-7} \dots 10^{-9}$ [6].

В общем виде протокол идентификации и аутентификации выглядит следующим образом [5]:

1. Субъект доступа предъявляет свой идентификатор ID .
2. Система управления доступом проверяет, существует ли среди зарегистрированных идентификаторов $ID_i = ID$. Если существует, то устанавливается, что субъект, назвавшийся субъектом доступа i , прошел идентификацию. В противном случае в доступе отказывается.
3. Система управления доступом запрашивает у пользователя его аутентификатор K .
4. Система управления доступом вычисляет значение $Y = F(ID_i, K)$.
5. Система управления доступом проводит сравнение значений Y и E_i . При совпадении этих значений устанавливается, что данный субъект доступа прошел аутентификацию. В противном случае в доступе отказывается.

Распространено несколько модификаций типовой схемы идентификации и аутентификации, обладающих лучшими характеристиками.

Структура объекта-эталона допускает изменение следующим образом: $E_i = (S_i, K_i)$, где S_i – случайный вектор, задаваемый при создании идентификатора субъекта доступа [5]. Данная схема применяется в ОС UNIX. Идентификатором ID служит имя пользователя, аутентификатором K_i – пароль пользователя, функция F представляет собой хэш-функции MD5 [7] или SHA [8].

В случае, когда необходимо выполнить взаимную проверку подлинности двух сторон, используется процедура «рукопожатия» [9], при которой каждая сторона проверяет аутентификатор партнера.

При высоких требованиях к безопасности (например, для карт электронных платежных систем) используются протоколы с нулевой передачей знаний

[10], позволяющие установить подлинность субъекта доступа без передачи какой-либо конфиденциальной информации.

1.1.2.2 Парольная аутентификация

Наиболее распространенным методом идентификации и аутентификации является парольная аутентификация [5]. При этом субъекту доступа назначается имя (идентификатор) и выделяется пароль, представляющие собой текстовые строки. При начале работы субъект доступа предъявляет системе свой идентификатор, затем система запрашивает пароль.

Основное достоинство парольной аутентификации – простота использования.

Парольная аутентификация имеет пониженную стойкость, поскольку выбор аутентифицирующей информации происходит из относительно небольшого множества осмысленных слов. На практике используются простые и удобные для запоминания пароли, еще более снижающие стойкость аутентификации. Другим недостатком является возможность перехвата пароля при передаче по сети.

Некоторые недостатки парольной аутентификации устраняются системами управления паролями. В этих системах пароли генерируются автоматически, их не требуется запоминать людям, поэтому используются более сложные пароли, повышающие стойкость аутентификации. Системы управления паролями обеспечивают возможность применения одноразовых паролей, перехватывать которые бесполезно.

1.1.2.3 Многофакторная аутентификация

При многофакторной аутентификации проверка подлинности субъекта доступа выполняется в несколько стадий. Наиболее распространен вариант, при котором субъект должен продемонстрировать знание пароля (или персонального идентификатора, PIN) и предъявить некоторый уникальный предмет, например, USB-токен или смарт-карту.

Использование многофакторной аутентификации значительно повышает безопасность. На предметы просто наложить административные ограничения, например, сдавать и принимать под роспись.

Недостатком многофакторной аутентификации является меньшее удобство по сравнению с парольной. Многофакторная аутентификация требует дополнительного оборудования и программного обеспечения, что увеличивает стоимость системы управления доступом.

1.1.2.4 Биометрическая аутентификация

Биометрическая аутентификация позволяет идентифицировать и аутентифицировать человека путем измерения его физиологических параметров и характеристик [5]. В качестве биометрических признаков используются:

- Узор радужной оболочки и сетчатки глаз.
- Отпечатки пальцев.
- Геометрическая форма руки.
- Форма и размеры лица.
- Особенности голоса.
- Биомеханические характеристики рукописного почерка.
- Биомеханические характеристики «клавиатурного почерка».

Преимуществами биометрической аутентификации являются высокая степень достоверности из-за уникальности биометрических признаков, трудность фальсификации биометрических признаков [6].

Биометрическая аутентификация применима только для людей, а не для других типов субъектов доступа (программ, аппаратных комплексов и т.п.). Высока вероятность ложных отрицательных срабатываний из-за погрешности приборов, считывающих биометрические признаки. При повреждении биометрических признаков (например, если порезать палец), пройти биометрическую аутентификацию невозможно.

1.1.2.5 Аутентификация с использованием криптографии

Для аутентификации широко применяется криптография. Протокол аутентификации с использованием шифрования (как симметричного, так и асимметричного) был предложен Нидхэмом и Шредером в 1978 г. [12]. Протокол предназначен для двусторонней аутентификации с использованием выделенного сервера аутентификации.

Рассмотрим протокол Нидхема-Шредера для случая симметричного шифрования. При симметричном шифровании в процессе аутентификации используется один секретный ключ, известный субъектам A и B и серверу аутентификации AS . Серверу аутентификации также известны секретные ключи A и B , которые не подлежат разглашению. Протокол выглядит следующим образом [12]:

1. Субъект A генерирует идентификатор I_A , который будет использоваться только один раз и отправляет его серверу аутентификации, зашифровав своим секретным ключом. Сообщение содержит $(A, B, I_A)^{KA}$. Верхний индекс KA показывает, что сообщение зашифровано секретным ключом субъекта A .

2. Сервер аутентификации генерирует ключ сессии CK , который будет использоваться для процедуры аутентификации и отправляет субъекту A сообщение вида: $(I_A, B, CK, (CK, A)^{KB})^{KA}$. Идентификатор I_A показывает, что сообщение является ответом на первое сообщение субъекта A , B указывает субъекта, с которым требуется установить аутентификацию, CK – ключ сессии. Часть сообщения $(CK, A)^{KB}$ субъект A расшифровать не сможет, т.к. она зашифрована секретным ключом B .

3. Субъект A посылает субъекту B сообщение $(CK, A)^{KB}$, полученное от сервера аутентификации. Расшифровав сообщение, субъект B также становится обладателем ключа CK . Дальнейшее взаимодействие между субъектами осуществляется с использованием шифрования с ключом CK .

4. Для того чтобы удостовериться в подлинности A , субъект B генерирует свой идентификатор для сессии I_B и посылает его A в сообщении $(I_B)^{CK}$.

5. Субъект A для подтверждения своей подлинности отправляет ответ $(I_B - I)^{CK}$.

В модифицированном варианте протокол Нидхема-Шредера с симметричным шифрованием применяется в Kerberos [13].

В случае шифрования с открытым ключом субъекты доступа A и B используют по паре открытый ключ – закрытый ключ ($PKA-SKA$ и $PKB-SKB$ соответственно). Открытые ключи субъектов распространяются через сервер аутентификации, закрытые – держатся в секрете. Протокол Нидхема-Шредера для асимметричного шифрования содержит следующие шаги [12]:

1. Субъект A запрашивает у сервера аутентификации открытый ключ субъекта B .

2. Сервер аутентификации отправляет субъекту A сообщение $(PKB, B)^{SKAS}$. Сообщение зашифровано секретным ключом сервера аутентификации. Субъект A должен знать открытый ключ сервера аутентификации, чтобы расшифровать это сообщение.

3. Субъект A уведомляет субъекта B о намерении вступить с ним в связь, отправив сообщение $(I_A, A)^{PKB}$. Сообщение содержит сгенерированный идентификатор I_A , который будет использоваться только один раз.

4. Субъект B запрашивает у сервера аутентификации открытый ключ A .

5. Сервер аутентификации отправляет субъекту B открытый ключ A в сообщении вида $(PKA, A)^{SKAS}$. Сообщение зашифровано секретным ключом сервера аутентификации. Субъект B должен знать открытый ключ сервера аутентификации, чтобы расшифровать это сообщение.

6. Субъект B генерирует идентификатор I_B , который будет использоваться только один раз и отправляет его субъекту A в сообщении $(I_A, I_B)^{PKA}$.

7. Субъект A подтверждает получение идентификатора и подлинность B сообщением $(I_B)^{PKB}$.

В случае асимметричного шифрования протокол содержит 7 шагов, что больше чем при симметричном шифровании. Но количество шагов можно сократить до 3 при условии, что субъектам известны открытые ключи друг друга.

В настоящее время разработана улучшенная версия протокола Нидхема-Шредера, обеспечивающая большую безопасность [12].

Основная проблема при использовании криптографии состоит в распространении ключей. Системы симметричного шифрования, использующие один ключ, требуют использования безопасных каналов для передачи ключей. Асимметричные системы с открытым и закрытым ключами позволяют передавать открытые ключи по небезопасным сетям. Но при этом необходимо быть уверенным, что субъект, с которым осуществляется взаимодействие с помощью алгоритмов с открытым ключом, является собственником закрытого ключа. Возможна замена открытого ключа законного участника открытым ключом злоумышленника без изменения идентификатора. Для предотвращения такой ситуации разработана популярная в настоящее время инфраструктура открытых ключей [14].

Инфраструктура открытого ключа использует сертификаты, которые являются структурами данных, связывающими значения открытого ключа с субъектом. Основные компоненты инфраструктуры открытого ключа (рис. 1.2.) [14]:

- End-entity (EE) – конечный участник, для которого выпущен сертификат. Конечный участник может быть как человеком, так и приложением.
- Certification Authority (CA) – сертификационный центр, который создает и подписывает сертификаты открытого ключа. СА отвечает за сертификаты не только в момент их выпуска, а на протяжении всего времени жизни сертификата.
- Public Key Certificate (PKC) – сертификат открытого ключа, содержащий открытый ключ участника и другую информацию, подписанную закрытым ключом СА, выпустившем данный сертификат.
- Registration Authority (RA) – регистрационный центр, необязательный участник, выполняющий некоторые административные функции: идентификация конечного участника, проверка, знает ли конечный участник закрытый ключ, соответствующий открытому ключу в сертификате, и некоторые другие.

– Certification Policy (CP) – политика сертификата – множество правил, определяющее применимость открытого ключа для конкретного сообщества или класса приложений с общими требованиями безопасности.

– Relying party (RP) – проверяющая сторона, которая использует сертификат для надежного получения открытого ключа конечного участника и некоторой другой дополнительной информации.

– Репозиторий – система или набор распределенных систем, которые хранят сертификаты и предназначены для распределения этих сертификатов между конечными участниками.

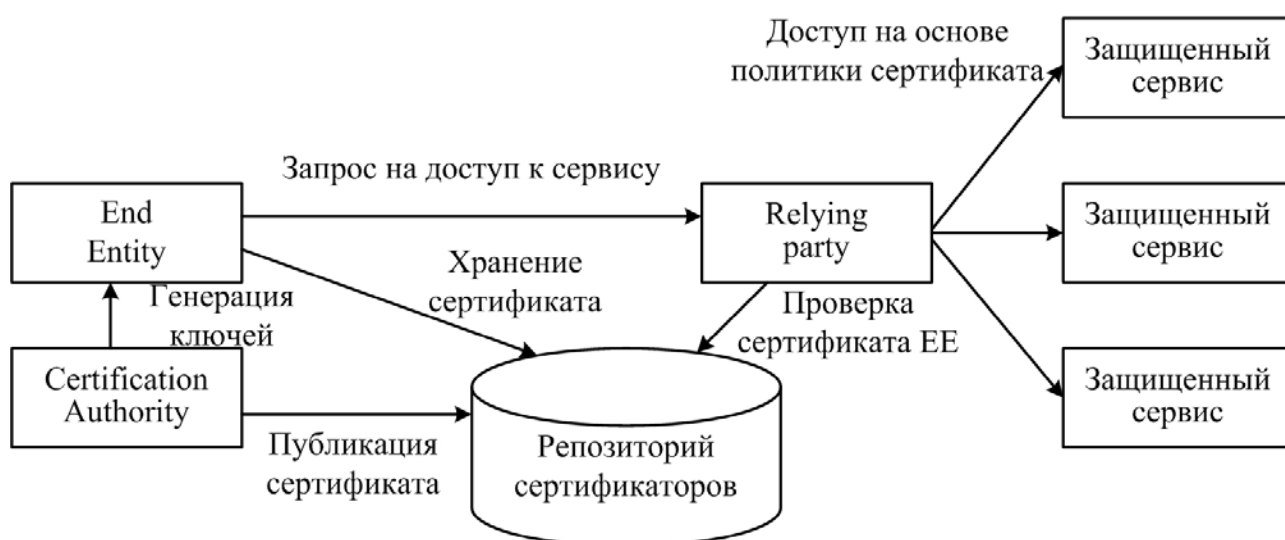


Рис 1.2. Основные компоненты инфраструктуры открытого ключа

Алгоритм работы инфраструктуры открытых ключей следующий. СА генерирует пару закрытый/открытый ключ для ЕЕ. Закрытый ключ передается ЕЕ. На основе открытого ключа и некоторых дополнительных атрибутов конечного участника СА генерирует сертификат и подписывает его своим закрытым ключом. Полученный сертификат публикуется в репозитории, к которому могут получить доступ все заинтересованные участники. При обращении конечного участника к защищенным ресурсам RP проверяет сертификат участника: подписан ли он сертификационным центром, которому доверяет RP, не истек ли срок действия сертификата, имеет ли конечный участник закрытый ключ, соответствующий открытому ключу в сертификате. Если все проверки прошли

успешно, конечному участнику разрешается доступ к защищенным ресурсам в соответствии с политикой сертификата.

Политики сертификата предоставляют возможность доступа конечных участников к защищенным сервисам. Различные сертификаты выпускаются, следуя разным процедурам и практикам, и могут быть предназначены для различных приложений. Например, в организации может быть две политики сертификатов - «Простые операции» и «Финансовые операции». Политика «Простые операции» предназначена для доступа к электронной почте, Интернет и т.п. Ключи создаются и хранятся с помощью недорогих систем на программной реализации. Политика «Финансовые операции» предназначена для защиты финансовых операций и предполагает более безопасные методы, например, хранение сертификатов в аппаратных токенах.

Существует несколько стандартов сертификатов, самый распространенный из них X.509 [15, 16].

Инфраструктура открытого ключа в настоящее время широко применяется на практике для аутентификации [17].

1.1.2.6 Аутентификация с нулевой передачей знаний

Алгоритмы аутентификации с нулевой передачей знаний разработаны для применения в случаях, когда требования к безопасности очень высоки [10]. Такие алгоритмы позволяют подтвердить подлинность субъекта доступа без передачи конфиденциальной информации.

Наиболее известный алгоритм идентификации с нулевой передачей знаний предложили Фейге, Фиат и Шамир в 1986 г. [11]. По этому алгоритму выбирается случайное значение модуля n (длина от 512 до 1024 бит) и распределяется между участниками взаимодействия. Обозначим за A сторону, доказывающую свою подлинность, за B сторону проверяющую подлинность стороны A . Доверенный центр вычисляет открытый и закрытый ключи для A . В качестве открытого ключа выбирается число V , являющееся квадратичным вычетом по модулю n (уравнение $x^2 = V(mod\ n)$ имеет решение и существует целое число

$V^l \bmod n$). Секретным ключом является наименьшее число S , для которого $S = \sqrt{V^{-1}} \pmod{n}$.

Алгоритм аутентификации выглядит следующим образом [11]:

1. Сторона А вычисляет значение $x = r^2 \bmod n$, где r – некоторое случайное число, такое что $r < n$.
2. Сторона В посылает стороне А случайный бит b .
3. Если $b = 0$, тогда А отправляет стороне В значение r . Если $b = 1$, то А отправляет стороне В значение $y = r \cdot S \bmod n$.
4. Если $b = 0$, сторона В проверяет, что $x = r^2 \bmod n$, чтобы убедиться, что сторона А знает \sqrt{x} . Если $b = 1$, сторона В проверяет, что $x = y^2 \cdot V \bmod n$, чтобы убедиться, что сторона А знает $\sqrt{V^{-1}}$.

Перечисленные шаги образуют один цикл протокола, называемый аккредитацией. Стороны повторяют этот цикл несколько раз при разных случайных значениях r и b , пока сторона В не убедится, что А знает секретный ключ.

Существует параллельная модификация алгоритма Фейге-Фиата-Шамира, позволяющая увеличить число аккредитаций, выполняющихся за 1 цикл [11].

Алгоритм аутентификации с нулевой передачей знаний, предложенный Гиллоу и Куискуотер, позволяет уменьшить количество аккредитаций, необходимое для подтверждения подлинности до одной [18]. Но объем вычислений при использовании этого алгоритма выше, чем у алгоритма Фейге-Фиата-Шамира.

Преимуществом алгоритмов аутентификации без передачи знаний является высокая безопасность – никакие конфиденциальные данные по сети не передаются и, следовательно, не могут быть перехвачены.

К недостаткам можно отнести сложность инфраструктуры аутентификации (необходима третья сторона – доверенный центр, занимающийся созданием и распределением ключей), высокая вычислительная нагрузка.

1.1.3 Однократная регистрация

В модели управления доступом Лампсона [2] субъект должен проходить идентификацию, аутентификацию и авторизацию при выполнении каждой операции с объектом. На практике это очень неудобно для субъектов доступа – людей. Для решения этой проблемы системы управления доступом реализуют функцию однократной регистрации.

Однократная регистрация предоставляет возможность проходить идентификацию и аутентификацию только один раз при первом обращении к объекту доступа. В случае успешного прохождения система «запоминает» субъекта (создает «сессию»). После этого субъект может обращаться к другим объектам доступа без повторных идентификации и аутентификации.

С понятием однократной регистрации связано понятие единого выхода. При этом в случае выхода из любой системы, поддерживающего однократную регистрацию, производится выход из всех систем.

1.1.4 Авторизация

В настоящее время разработано и используется несколько моделей авторизации: дискреционное, мандатное, ролевое, атрибутное управление. Допускается как отдельное, так и совместное применение различных моделей управления доступом в одной системе.

1.1.4.1 Дискреционное управление доступом

Дискреционное управление – появившийся первым и наиболее простой метод [19], при котором ведется управление доступом поименованных субъектов к поименованным объектам. Для каждой пары «субъект»-«объект» задается перечень операций, которые субъект может выполнять с объектом. Права доступа к объекту задает субъект, который является владельцем объекта. Полученные права доступа к объекту субъект может передавать любому другому субъекту.

При дискреционном управлении используется несколько форматов представлений прав доступа. **Матрица доступа** предложена Лампсоном [20] в 1974

году. В столбцах такой матрицы приводятся все объекты, существующие в системе, в строках – все субъекты, а в ячейках – права доступа субъектов к объектам. Пример матрицы доступа пользователей к файлам показан в табл. 1.1

Таблица 1.1.

Пример матрицы доступа

-	File1	File2	File3
bob	read	read, write	-
alice	read, write	-	read
joe	-	-	read, write, execute

Другой формат представления – **списки контроля доступа**. Для каждого объекта заводится список, в котором указывается, какие субъекты могут с ним работать и какие операции может выполнять каждый субъект.

Списки возможностей связывает каждого субъекта с доступным ему списком операций над объектами.

В упрощенном варианте списки контроля доступа являются представлением столбцов матрицы доступа, а списки возможностей – представлением строк (рис 1.3). Однако списки возможностей имеют ряд неочевидных, на первый взгляд, преимуществ, описанных в работе [21]. Модель списков контроля доступа предполагает наличие пространства имен, в котором субъект ищет объекты доступа. Список возможностей в явном виде содержит как список доступных субъекту объектов, так и список операций, которые можно с ними выполнять. В модели списков контроля доступа управление ведется на уровне субъектов. Тот, кто имеет право менять список контроля доступа, может внести в него любой субъект. А имея право менять список возможностей, можно назначать права только одному конкретному субъекту. Такой подход значительно облегчает распределение полномочий между администраторами по организационному или территориальному принципу.

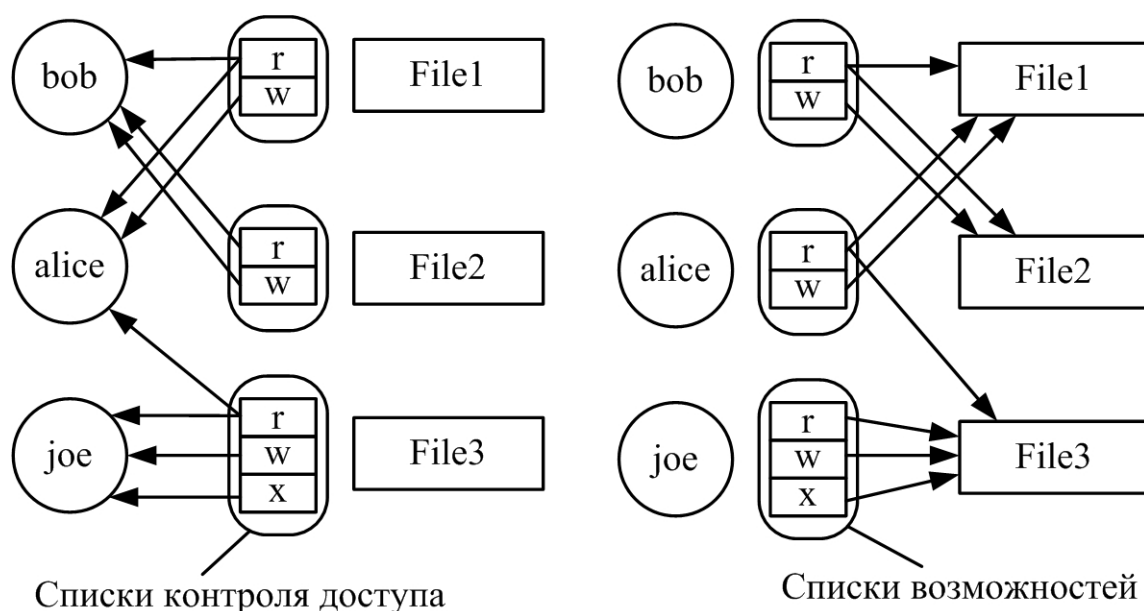


Рис 1.3. Списки контроля доступа и возможностей

Недостатком дискреционного управления доступом является высокая сложность управления в крупных организациях с большим количеством субъектов и объектов доступа. Матрица доступа (или ее представление в виде списков контроля доступа и возможностей) становится огромной, внесение изменений является сложным и длительным процессом.

Частично упростить процесс управления при дискреционном подходе помогает механизм групп. При этом права доступа назначаются не каждому субъекту в отдельности, а группе. Субъекты могут добавляться или исключаться из группы динамически.

Другой недостаток дискреционного управления возможность получения доступа субъектами, которые не должны иметь такой доступ. Согласно дискреционной модели субъект, получивший право на выполнение некоторой операции с объектом, может передать это право другому субъекту, без уведомления владельца объекта. Другой вариант несанкционированного доступа – субъект, получивший доступ на чтение к объекту, создает копию объекта и передает права на доступ к копии любым другим субъектам.

Централизация управления при дискреционном подходе является сложной задачей, т.к. распределением прав доступа занимаются владельцы объектов. Одним из подходов к централизации в дискреционной модели управления

является выделение среди всех субъектов доступа суперпользователя, который может назначать права доступа любому объекту любого владельца. Такой подход применяется в Unix, где суперпользователем является root.

1.1.4.2 Мандатное управление доступом

Модель мандатного управления доступом предложена Беллом и ЛаПадула в работе [22]. Модель разрабатывалась для применения в военных системах и обеспечивает высокую безопасность. Объекты доступа делятся на несколько уровней, обозначаемых метками безопасности. Субъекты доступа также делятся на уровни с использованием тех же меток. Широко применяемый на практике набор меток: неклассифицированно \leq конфиденциально \leq секретно \leq совершенно секретно. Пример использования мандатного управления доступом показан на рис. 1.4.

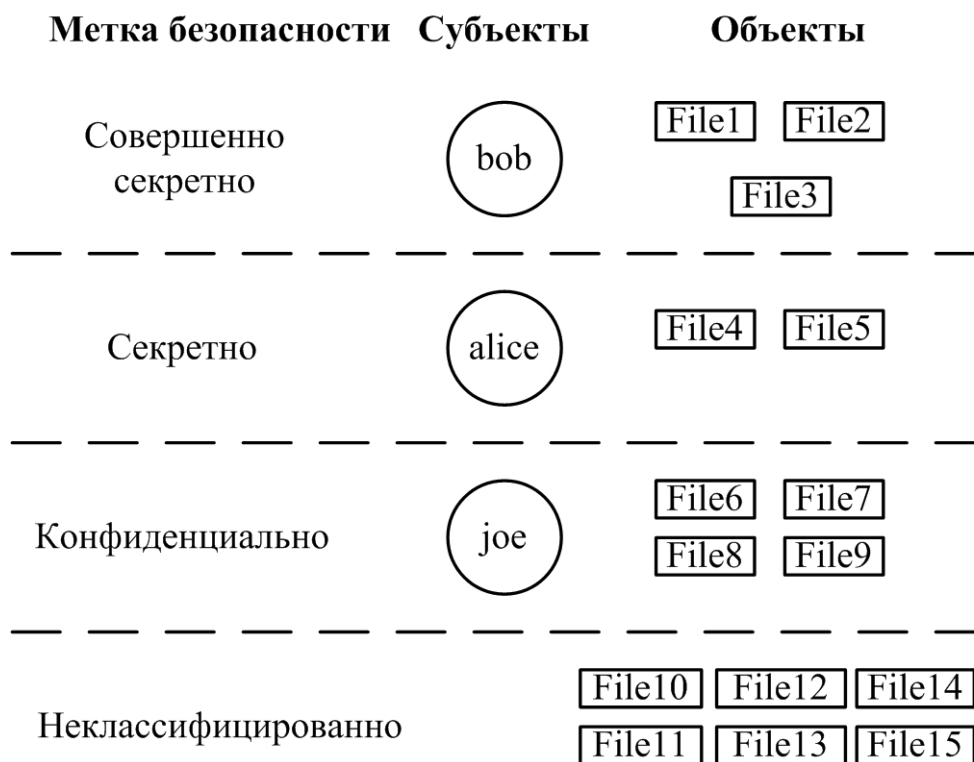


Рис 1.4. Мандатное управление доступом

Субъекты доступа имеют право чтения всех объектов, имеющих метку безопасности одного с ними уровня и ниже. Право записи субъекты имеют только в объекты, имеющие метку безопасности одного с ними уровня. Это сделано для предотвращения попадания конфиденциальных данных в объекты

с более низким уровнем доступа. Например, чтобы генерал не смог случайно выдать секреты младшим офицерам.

Модель мандатного управления доступом разрабатывалась для военных систем с высокими требованиями к безопасности и плохо применимо в других системах. В коммерческих и научных организациях сложно выстроить жесткую иерархию уровней доступа, подобную военной, поэтому требуются более гибкие методы.

1.1.4.3 Ролевое управление доступом

Ролевое управление доступом [23] предлагает более гибкий метод, по сравнению с мандатным, и менее трудоемкий, по сравнению с дискреционным. При ролевом управлении права доступа назначаются не субъектам доступа, а ролям, которые соответствуют обязанностям, выполняемым субъектом в организации («Директор», «Аспирант», «Инженер»). Роли позволяют быстро и согласованно выделить субъекту весь набор прав доступа, необходимый ему для работы.

Модель ролевого управления доступом $RBAC_0$, предложенная в работе [23], показана на рис.1.5. Ролям из множества R назначаются права доступа из множества P с помощью отношения PA . Роли назначаются пользователем из множества U с помощью отношения UA . При работе пользователя создается сессия (множество S), в которой пользователь активирует одну или несколько доступных ему ролей и получает соответствующие права доступа.

В работе [23] определяются другие модели ролевого управления, позволяющие использовать иерархию ролей, ограничения на допустимые сочетания компонентов модели, делегирование полномочий управления доступом.

Роли позволяют реализовать принцип «разграничения обязанностей»: роли со взаимоисключающими правами доступа не могут быть назначены одному пользователю (например, один человек не может одновременно использовать роли «Директор» и «Аспирант»).

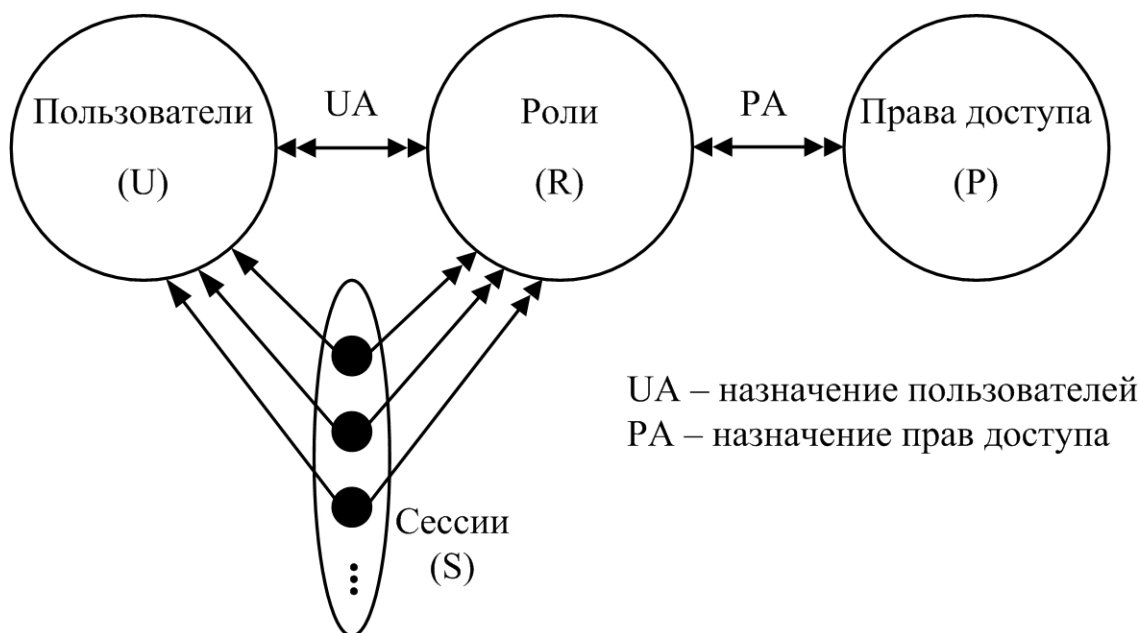


Рис 1.5. Модель ролевого управления доступом [23]

С помощью ролей можно временно ограничить доступный набор прав доступа. Например, субъект доступа, которому назначена роль системного администратора, может выполнять большую часть текущей работы, не активируя эту роль, чтобы снизить последствия возможных ошибок или работы вредоносных программ. Субъект активирует роль системного администратора только тогда, когда необходимо выполнить операции, требующие прав доступа системного администратора, и деактивирует роль сразу же, после окончания выполнения операций.

Роли предоставляют эффективный и удобный механизм управления доступом, поэтому широко используются на практике.

1.1.4.4 Атрибутное управление доступом

Атрибутное управление доступом [24] возникло благодаря широкому распространению сети Интернет и появлению большого числа сервисов, доступных через сеть. В таком случае модель управления доступом, при которой для каждого субъекта создается учетная запись, и назначаются права доступа, подходит плохо, т.к. в Internet существует много субъектов и каждый субъект использует большое количество объектов, предоставляемых разными организациями. Поэтому для принятия решения о разрешении или запрещении работы с

объектом выполняется анализ описательных атрибутов субъекта, а не назначенных ему прав доступа.

Наиболее часто атрибутное управление применяется для контроля доступа к Web-сервисам. Например, Web-сервис бронирования автомобиля может быть доступен только тем пользователям, у которых есть водительское удостоверение (или другими словами атрибут «hasDriversLicense» имеет значение «true»).

1.1.5 Делегирование

Формальная модель делегирования предложена Лампсоном в работе [2]. Субъект может делегировать часть своих прав доступа другому субъекту. Например, администратор организации может делегировать часть полномочий по управлению доступом администраторам подразделений. Делегирование является популярным методом распределения обязанностей по управлению доступом между администраторами.

1.1.6 Анализ состояния технологий в области управления доступом к сервисам

Большая часть современных информационных систем для управления доступом к сервисам использует модель Лампсона, предложенную в 1974 г. Несмотря на общую модель, на практике используется большое количество различных методов реализации процедур защиты информации: идентификации, аутентификации и авторизации. Такое многообразие обеспечивает гибкость: разработчики каждой системы выбирают метод управления доступом, наиболее полно отвечающий требованиям данной системы.

В сетях крупных организаций, с большим количеством сервисов и пользователей, многообразие технологий приводит к повышению сложности процесса управления доступом. Сервисы разных типов, использующие разные технологии управления доступом, требуют создания инфраструктуры управления: репозитория правил разграничения доступа, системы управления доступом и т.п. Большое количество сервисов разных типов приводит к большому количе-

ству инфраструктур управления доступом, которые нужно создавать и поддерживать. Часть информации в таких инфраструктурах дублируется и требует синхронизации при изменениях.

Управление доступом требует высокой квалификации администраторов: необходимо знать тонкости большого числа различных технологий управления доступом, часть из которых отличаются высокой сложностью.

Пользователям неудобно работать с сервисами, использующими различные технологии управления доступом, т.к. для работы с разными сервисами требуется разная идентификационная информация: имена и пароли, сертификаты, токены или смарт-карты. Пароли пользователям необходимо помнить, токены и смарт-карты всегда иметь при себе и предоставлять при каждом обращении к сервису.

Для упрощения процесса управления доступом необходимо выполнить интеграцию систем управления доступом к сервисам, использующим различные технологии управления доступом.

1.2 Методы интеграции информационных систем

В настоящее время распространены три подхода к интеграции информационных систем, обеспечивающие различные уровни интероперабельности [25] (рис 1.6.):

- **синтаксическая** интеграция основывается на использовании согласованных форматов данных;
- **структурная** интеграция обеспечивает согласование структур данных путем преобразования форматов с применением метаданных;
- **семантическая** интеграция устанавливает смысловое соответствие между сущностями.

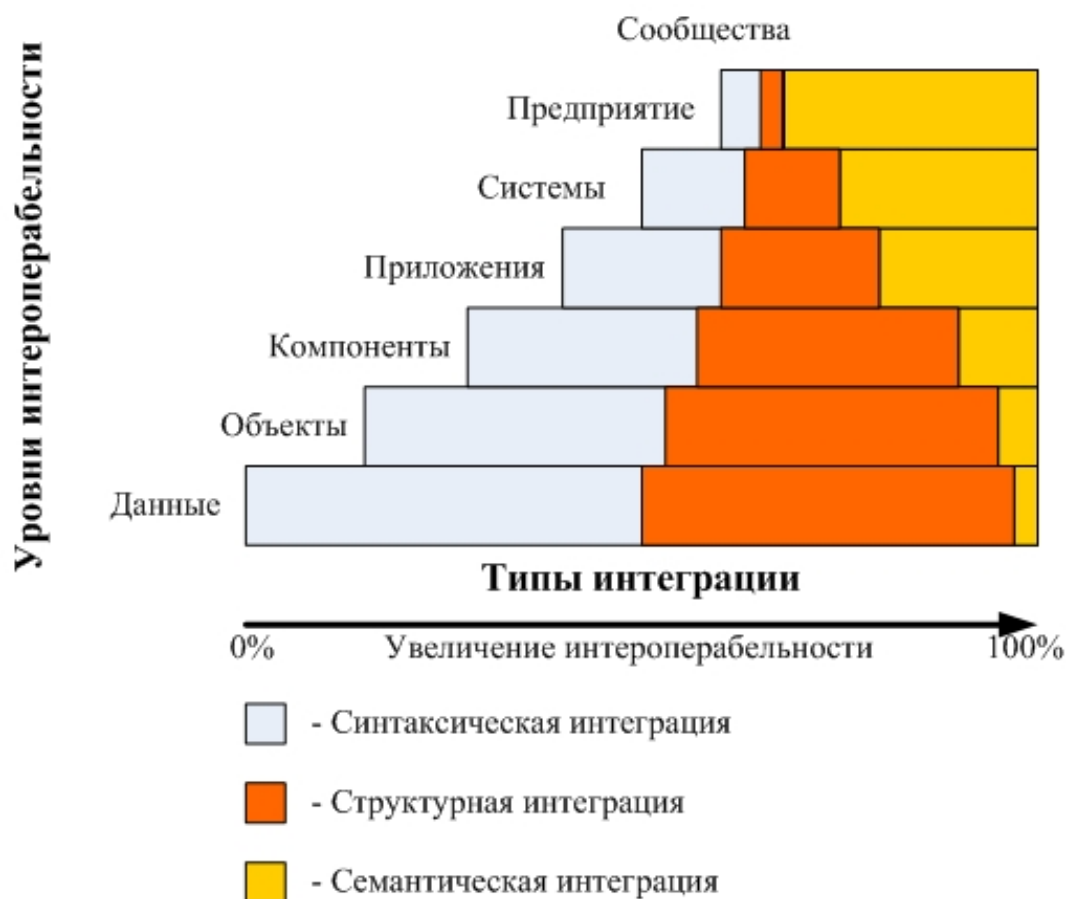


Рис 1.6. Типы интеграции и уровни интероперабельности [25]

Наиболее полную интероперабельность интегрируемых систем обеспечивает семантическая интеграция [25]. Интеграционные решения, использующие синтаксический и структурный подход являются частными, рассчитанными на определенные системы. Широкое применение таких решений затруднено.

В настоящее время наиболее популярной технологией семантической интеграции являются онтологии. Онтологии определяют общий словарь предметной области, который может совместно использоваться людьми или информационными системами.

Пример семантической интеграции систем управления доступом к сервисам показан на рис. 1.7. На этом рисунке представлен фрагмент онтологии системы управления доступом к сервисам, содержащий класс «Субъект доступа» с атрибутами «Идентификатор», «Пароль», «ФИО» и др. Онтология определяет смысловую нагрузку понятий системы управления доступом к сервисам, представленных в разных форматах с различной структурной организацией (таблица реляционной СУБД, текстовый файл /etc/passwd в UNIX, каталог LDAP).

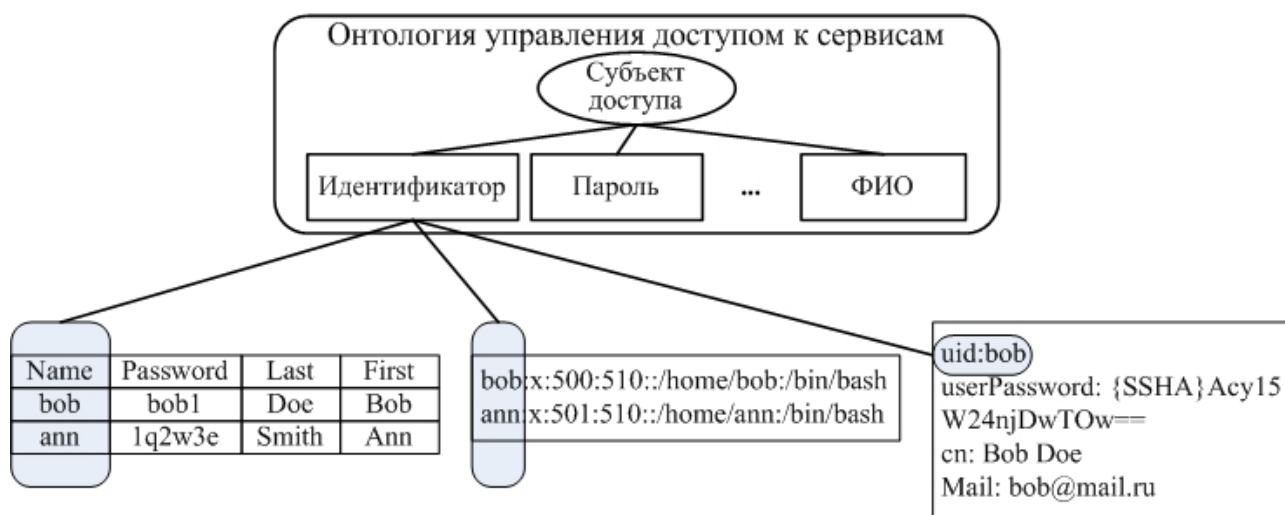


Рис 1.7. Пример семантической интеграции на основе онтологии

Для разработки онтологий существует широкий набор инструментальных средств: язык описания онтологий OWL (Web Ontology Language) [26], являющийся стандартом W3C, среды редактирования онтологий Protégé [27], Ontolingua [28], Chimaera [29], готовые и доступные к использованию онтологии, описывающие различные предметные области.

Таким образом, интеграцию систем управления доступом к сервисам необходимо выполнять с помощью семантического подхода, что обеспечит максимальную интероперабельность. В качестве инструментария интеграции целесообразно использовать онтологии.

1.3 Существующие системы интеграции управления доступом к сервисам

В данном разделе выполнен обзор существующих систем интеграции управления доступом к сервисам. Выявлены аналоги трех типов: системы централизованного управления, системы федеративной идентификации, системы интегрированного управления.

1.3.1 Традиционный подход

Традиционный подход к управлению доступом состоит в том, что каждый сервис реализует задачу управления собственными средствами: имеет собственный репозиторий учетных записей и правил разграничения доступа, собственные системы управления и учета. Например, в ОС Unix учетные записи

хранятся в текстовых файлах `/etc/passwd` и `/etc/groups`, управление ведется с помощью команд ОС (`useradd`, `usermod` и т.п.). Достоинством подхода является удобство и автономность: все, что нужно для управления уже есть в составе сервиса. В крупных сетях, с большим количеством пользователей и сервисов, традиционный подход порождает много проблем. В первую очередь это сложность администрирования. Большое количество сервисов приводит к большому количеству репозиторий, которые нужно поддерживать и обеспечивать актуальность информации. Управление доступом выполняется отдельно к каждому сервису, через свою систему, что существенно снижает продуктивность работы и повышает вероятность ошибки.

Традиционный подход делает неудобной работу пользователей с сервисами. Из-за сложностей администрирования и ошибок, пользователи часто по долгу не могут получить права на работу с необходимым сервисом. Для каждого сервиса требуются отдельные идентификатор и пароль, которые нужно вводить при каждом обращении. Большое количество паролей трудно запомнить, поэтому применяются простые пароли, а сложные записываются на бумаге, что негативно сказывается на безопасности. Запросы пользователей по смене паролей создают большую нагрузку на службу поддержки.

1.3.2 Централизованное управление

Для устранения недостатков традиционного подхода были разработаны системы централизованного управления доступом. В таких системах сервисы сами не выполняют никаких действий по управлению доступом, а обращались для этого к внешним службам.

Один из подходов к созданию систем централизованного управления определен в стандарте RFC 2903 [30], описывающем архитектуру Authentication, Authorization and Accounting (AAA). Эта архитектура реализована в протоколах RADIUS [31], TACACS [32], DIAMETER [33], нашедших широкое применение в телекоммуникационном оборудовании. В научной среде протокол RADIUS применяется в проекте EDUROAM [34].

Для централизованного хранения данных о пользователях в ОС Unix была создана система NIS (Network Information Service) [35]. В NIS текстовый файл наподобие `/etc/passwd` редактируются только на одной машине – сервере NIS, который передает их на все остальные UNIX машины (клиенты NIS). Для повышения надежности серверы NIS дублируются. В настоящее время NIS практически не применяется, ее заменили системы с большими возможностями на основе LDAP или Kerberos.

Каталоги LDAP сейчас пользуются наибольшей популярностью и применяются в большинстве распространенных операционных систем: Active Directory в Microsoft Windows, Sun Java System Directory Server в Solaris, Novell eDirectory в Netware. Каталоги, как и NIS, позволяют создать централизованный репозиторий учетных записей. Доступ к такому каталогу осуществляется по протоколу LDAP [36]. Достоинством LDAP является наличие стандартных схем для описания данных о пользователях [37, 38, 39], что обеспечивает интероперабельность.

Система Kerberos [13], кроме централизованного хранения учетных записей и управления доступом, предоставляет полезную функцию однократной регистрации. Аутентификацию пользователей в Kerberos выполняет централизованный сервер аутентификации. Пройдя аутентификацию, пользователь получает специальный билет (ticket), имея который может обращаться к нужному ему серверу без повторной аутентификации. В целях безопасности Kerberos выполняет шифрование передаваемых по сети билетов. Протокол Kerberos стандартизован [40], поддерживается в Windows и в UNIX.

Бесплатная система централизованного управления и однократной регистрации пользователей Web-приложений A-Select [41], использует архитектуру, аналогичную Kerberos: сервер аутентификации выдает билеты для доступа к различным Web-приложениям. Система обеспечивает однократную регистрацию и широко применяется в научной среде [42].

На управление доступом к Web рассчитаны системы, использующие протокол SAML (Security Assertion Markup Language), определенный в стандарте

OASIS [43]. В SAML для описания прав доступа используется XML. Существует бесплатная реализация OpenSAML [44].

Интерес представляют системы наподобие Microsoft Windows Life ID [45] (.Net Passport), в которых всю работу по хранению данных о пользователях берет на себя сторонняя компания, в данном случае Microsoft. Доступ к данным о пользователях и правилах разграничения доступа осуществляется с помощью API, предоставляющего дополнительные возможности, например, однократную регистрацию. Однако не много компаний готовы доверить хранение персональных данных своих сотрудников сторонней организации.

Системы централизованного управления доступом решают многие проблемы управления, присущие традиционному подходу: обеспечивают централизованное хранение данных о пользователях и правилах разграничения доступа, единую систему управления, однократную регистрацию. Тем не менее, ни один из разработанных протоколов идентификации не получил повсеместного распространения. В крупной сети невозможно вести управление всеми сервисами на основе только одного протокола, т.к. часть сервисов могут не поддерживать этот протокол. Применение систем централизованного управления ведет к созданию «островков управления»: в организации применяется несколько подобных систем, каждая обслуживает определенный набор сервисов. Взаимодействия между такими системами практически не происходит.

Другим недостатком централизованных систем является указанная в названии ориентация на создание единого центра управления. Сейчас в связи с широким распространением Интернет, появилось большое число научных организаций, предоставляющих сервис в сети. Такие организации хотят предоставить безопасный доступ к сервисам своим организациям-партнерам. Управлять доступом к сервисам, предоставляемым разными организациями, находящимися в разных зонах административной или организационной ответственности (доменах) из одного центра очень сложно.

1.3.3 Федеративная идентификация

Федеративная идентификация используется для управления доступом к сервисам, находящимся в разных доменах [46]. В каждом домене управление идентификацией выполняется независимо от других, используются собственные репозитории данных о пользователях и правила разграничения доступа. Системы федеративной идентификации позволяют создать круг доверия: объединить учетные записи одного пользователя, хранящиеся в разных доменах, и согласованно управлять правами доступа. Это делает возможным реализацию ряда удобных функций, например, однократной регистрации: после аутентификации на сервисе одного из доменов для работы с сервисами других доменов повторная регистрация не требуется. Популярной системой федеративной идентификации, основанной на открытых стандартах, является Liberty [47]. Альтернативное решение, предлагаемое компаниями Microsoft и IBM, основывается на WS-Security и WS-Federation [48]. В научной среде [49] используется Shibboleth [50], разработанный для проекта Internet2.

1.3.4 Интегрированное управление доступом к сервисам

Системы интегрированного управления доступом к сервисам призваны устранить зависимость от конкретных протоколов централизованного или федеративного управления. Системы предлагаются многими ведущими производителями программного обеспечения – IBM Tivoli Identity Manager [51], Sun Identity Manager [52], Oracle Identity Manager [53], Microsoft Identity Integration Server [54], Novell Identity Manager [55]. Эти решения во многом похожи друг на друга, они используют одинаковые принципы построения и стандартные протоколы взаимодействия. Для примера рассмотрим Sun Identity Manager, который признается Gartner лидирующим в классе [56]. Решение состоит из нескольких компонентов [52]: Directory Server служит для хранения данных о пользователях и правах доступа с возможностью масштабирования и распределенного взаимодействия. Access Manager обеспечивает аутентификацию, авторизацию и однократную регистрацию пользователей. Federation Manager добав-

ляет возможность федеративного управления идентификацией. Важным преимуществом Sun Identity Manager является бесплатное распространение, что делает возможным его применение в научных организациях. В дополнение к бесплатному распространению компания Sun Microsystems начала открывать исходный код своих продуктов, в частности исходные коды Access Manager доступны в рамках проекта OpenSSO [57].

Недостатком подобных систем является узкий круг поддерживаемых протоколов и сервисов из-за отсутствия в их основе четкой семантической модели управления. В результате задача интеграции механизмов управления доступом к информационным, сетевым и вычислительным сервисам с использованием существующих систем затруднена. Чтобы сделать возможной такую интеграцию, необходимо разработать семантическую модель управления доступом к сервисам разных типов.

1.3.5 Оценка существующих систем интеграции управления доступом к сервисам

Оценка наиболее близких систем интеграции управления доступом к сервисам приведена в табл. 1.2. Системы оценены по десятибалльной шкале (1 – минимальная оценка, 10 – максимальная).

Анализ позволил выбрать прототип – Sun Identity Manager. Основным недостатком прототипа и других аналогов является низкая интероперабельность. Это объясняется применением синтаксической (системы централизованного управления и федеративной идентификации) и структурной (системы интегрированного управления) интеграции. В результате задача интеграции механизмов управления доступом к сервисам разных типов с использованием существующих систем затруднена. Расширить интероперабельность можно с помощью семантической интеграции, для чего необходимо разработать четкую семантическую модель управления доступом к сервисам.

Таблица 1.2.

Оценка существующих систем интеграции управления доступом к сервисам

Название	Интероперабельность	Функциональные возможности	Методы управления доступом	Схема распространения	Область применения	Всего
Системы централизованного управления						
AAA	1	6	1	10	5	23
LDAP	1	6	1	10	5	23
KERBEROS	1	6	1	10	5	23
A-Select	1	6	1	10	5	23
SAML	1	7	1	10	5	24
Системы федеративной идентификации						
Liberty	4	7	7	8	5	31
WS-Federation	3	7	7	5	5	27
OpenID	2	5	5	8	3	23
Windows CardSpace	2	5	6	5	3	21
Системы интегрированного управления						
IBM Tivoli Identity Manager	5	10	10	5	10	40
Sun Identity Manager	5	10	10	10	10	45
Oracle Identity Manager	5	10	10	5	10	40
Novell Identity Manager	5	9	8	5	10	37
Microsoft Identity Intergation Server	5	9	8	5	10	37

1.4 Онтологии предметной области управления доступом к сервисам

1.4.1 Специализированные онтологии управления доступом

В области управления доступом к сервисам разработано большое количество онтологий. Существуют онтологии для различных методов управления доступом: на основе списков доступа [58, 59], ролевого [60], атрибутного [61], на базе правил [62, 63] и с учетом контекста [64]. Разрабатываются онтологии для описания управления доступом к конкретным информационным системам:

HL7 [60], LSDIS [65], Access-eGov [66]. Есть онтологии для решения некоторых задач управления доступом, например, автоматической декомпозиции правил управления доступом в распределенной среде [67].

Применение существующих онтологий для интеграции механизмов управления доступом к сервисам разных типов затруднено. Онтологии, описывающие управление доступом к конкретным информационным системам, трудно применить даже для других информационных систем из-за отличий в наборе информационных ресурсов, выполняемых операций и, как следствие, прав доступа. А использовать онтологию управления доступом, разработанную для конкретной информационной системы, для управления доступом к сетевым и вычислительным устройствам практически невозможно.

Онтологии, описывающие методы управления доступом (ролевое, атрибутное и т.п.) могут применяться для интеграции управления доступом к сервисам разных типов. Но в созданной на основе такой онтологии системе будет возможно применение только одного метода управления, что существенно ограничивает функциональные возможности и снижает эффективность использования системы.

1.4.2 Онтологии верхнего уровня

Наиболее популярным и доказавшим свою эффективность методом разработки онтологий является использование так называемых «высокоуровневых» онтологий [68]. При этом в отличие от разработчиков существующих онтологий управления доступом, сразу начинающих описание специализированной предметной области управления доступом к сервисам, сначала рассматриваются общие для всех информационных систем понятия (входящие в высокоуровневую онтологию), а затем предлагается сужение общих понятий на предметную область управления доступом к сервисам. Использование высокоуровневых онтологий для интеграции информационных систем позволяет построить более полный словарь предметной области и обеспечить взаимодействие информационных систем без привязки к конкретному содержанию, тем самым обеспечивая более высокий уровень интероперабельности [68].

В настоящее время разработано несколько высокоуровневых онтологий: Cys [69], SUMO [70], GOL [71], DOLCE [72], BWW [73, 74]. Онтологии Cys, SUMO, GOL ставят задачу описания всего, что существует в мире. Онтологии DOLCE и BWW более узкие и сконцентрированы на информационных системах.

1.4.3 Оценка онтологий в области управления доступом к сервисам

Оценка онтологий в области управления доступом к сервисам разных типов приведена в табл. 1.3. Онтологии оценены по десятибалльной шкале (1-минимальная оценка, 10 – максимальная).

Таблица 1.3.

Оценка онтологий в предметной области управления доступом

Онтология	Широта применения	Ориентация на информационные системы	Практическое использование	Всего
Специализированные онтологии				
HL7	1	10	1	12
LSDIS	1	10	1	12
Access-eGov	1	10	1	12
Онтологии для методов управления доступом к сервисам				
Access Control Lists ontology	3	10	3	16
Role base access control ontology	3	10	3	16
Attribute base access control ontology	3	10	2	15
Rule base access control ontology	3	10	2	15
Context aware access control ontology	2	10	1	13
Онтологии верхнего уровня				
Cys	10	1	10	21
SUMO	10	2	9	21
GOL	8	2	6	16
BWW	6	10	8	24
DOLCE	7	8	8	23

На основе результатов оценки в качестве прототипа была выбрана онтология BWW (Bunge, Wand and Weber ontology).

Основные понятия онтологии BWW приведены в табл. 1.4.

Таблица 1.4.

Основные понятия онтологии BWW

Понятие	Описание
Thing	Объект, базовая единица онтологии BWW. Мир состоит из объектов.
Property	Свойство объекта.
Class	Класс – множество объектов, обладающих одинаковым набором свойств
State	Состояние объекта – множество значений свойств объекта в определенный момент времени.
Event	Событие – изменение состояния объекта. Задается начальным и конечным состоянием, а также правилом изменения состояния (transformation).
Transformation	Преобразование – правило изменения состояния объекта.
History	История объекта – список событий объекта.
Interaction	Взаимодействие – два объекта взаимодействуют, если история хотя бы одного объекта зависит от истории другого.
System	Система – множество взаимодействующих объектов.

Недостатком онтологии BWW является слишком высокий уровень абстракции: онтология описывает информационные системы в целом, без учета особенностей конкретных информационных систем. Необходимо выполнить адаптацию онтологии BWW к предметной области управления доступом к сервисам разных типов с учетом существующих Российских и международных стандартов и распространенных методов управления доступом.

1.5 Официальные документы в области управления доступом

Попытки определить набор понятий в области информационной безопасности в целом, и в области управления доступом к сервисам в частности, дела-

ются в нормативных документах. Основными зарубежными стандартами в этой сфере являются ISO 2382-8:1998 [75] и ISO/IEC 27000 [76] (находится в стадии разработки). В России основные термины в области информационной безопасности определены ГОСТ Р 50739-95 [77] в Руководящем документе Гостехкомиссии России [78]. Приведем некоторые определения из Руководящего Документа, непосредственно относящиеся к области управления доступом к сервисам.

Таблица 1.5.

Термины и определения Руководящего документа Гостехкомиссии России [78]

Термин		Определение
Доступ к информации	Access to information	Ознакомление с информацией, ее обработка, в частности, копирование, модификация или уничтожение информации
ПРД - Правила разграничения доступа	Security policy	Совокупность правил, регламентирующих права доступа субъектов доступа к объектам доступа
Субъект доступа	Access subject	Лицо или процесс, действия которых регламентируются правилами разграничения доступа
Объект доступа	Access object	Единица информационного ресурса автоматизированной системы, доступ к которой регламентируется правилами разграничения доступа
Уровень полномочий субъекта доступа	Subject privilege	Совокупность прав доступа субъекта доступа
Система разграничения доступа	Security policy realization	Совокупность реализуемых правил разграничения доступа в средствах ВТ

Таблица 1.5. (продолжение)

Термин		Определение
Идентификатор доступа	Access identifier	Уникальный признак субъекта или объекта доступа
Идентификация	Identification	Присвоение объектам и субъектам доступа идентификатора и (или) сравнение предъявляемого идентификатора с перечнем присвоенных идентификаторов
Пароль	Password	Идентификатор субъекта доступа, который является его (субъекта) секретом
Аутентификация	Authentication	Проверка принадлежности субъекту доступа предъявленного им идентификатора, подтверждение подлинности
Модель защиты	Protection model	Абстрактное описание комплекса программно-технических средств и/или организационных мер защиты от НСД
Дискреционное управление доступом	Discretionary access control	Разграничение доступа между поименованными субъектами и поименованными объектами. Субъект с определенным правом доступа может передать это право любому другому объекту
Мандатное управление доступом	Mandatory access control	Разграничение доступа субъектов к объектам, основанное на характеризуемой меткой конфиденциальности информации, содержащейся в объектах, и официальном разрешении (допуске) субъектов обращаться к информации такого уровня конфиденциальности

Таблица 1.5. (продолжение)

Термин		Определение
Концепция диспетчера доступа	Reference monitor concept	Концепция управления доступом, относящаяся к абстрактной машине, которая посредничает при всех обращениях к объектам
Администратор системы защиты	Security administrator	Субъект доступа, ответственный за защиту автоматизированной системы от НСД

Руководящий Документ определяет достаточно большое количество терминов в области управления доступом. В нем нашли отражения понятия модели управления доступом Лампсона (субъект доступа, объект доступа, диспетчер доступа), процедуры защиты информации (идентификация, аутентификация, авторизация), разные методы управления доступом (дискреционное и мандатное).

Однако предлагаемый словарь далеко не полный. Из методов управления доступом представлены только два: дискреционное и мандатное. Отсутствует понятие однократной регистрации, и связанное с ним понятие сессии. Не описано понятие делегирования управления доступом к сервисам. Словарь предметной области управления доступом к сервисам требует более детального описания. Руководящий Документ может служить хорошей основой для такого описания, т.к. определяет стандарт на терминологию в области управления доступом в России.

1.6 Выводы по главе

Обзор состояния технологий управления доступом к сервисам показал, что основной причиной сложности процесса управления доступом к сервисам является многообразие технологий управления доступом. Для упрощения управления доступом необходима интеграция систем управления, использующих различные технологии управления доступом к сервисам.

В настоящее время существует три подхода к интеграции информационных систем: синтаксическая, структурная и семантическая. Наибольшую интероперабельность обеспечивает семантический подход, поэтому его целесообразно использовать для интеграции управления доступом к сервисам разных типов. Инструментом семантической интеграции являются онтологии.

Анализ существующих систем интеграции управления доступом к сервисам выявил их основной недостаток: низкую интероперабельность. Существующие системы обеспечивают интеграцию управления доступом к небольшому числу сервисов из-за существенных различий в системе понятий и наборах операций по управлению доступом. Низкая интероперабельность объясняется применением в существующих системах синтаксического и структурного подходов к интеграции. Увеличить интероперабельность можно с помощью семантической интеграции. Оценка аналогов позволила выбрать прототип создаваемой системы: Sun Identity manager [52].

Семантическая интеграция управления доступом к сервисам разных типов требует создания семантической модели управления доступом к сервисам. В настоящее время популярным инструментом создания семантических моделей являются онтологии. Обзор существующих онтологий в области управления доступом к сервисам позволил выявить прототип для онтологии управления доступом – онтологию верхнего уровня BWW [73]. Как и любая онтология верхнего уровня, онтология BWW является слишком абстрактной: описывает информационные системы в целом, без учета особенностей конкретных информационных систем. Необходимо выполнить адаптацию онтологии BWW к предметной области управления доступом к сервисам разных типов с учетом существующих Российских и международных стандартов и распространенных методов управления доступом.

2 СЕМАНТИЧЕСКАЯ ИНТЕГРАЦИЯ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ

2.1 Анализ и критика прототипа

2.1.1 Анализ прототипа

Схема прототипа приведена на рис. 2.1.

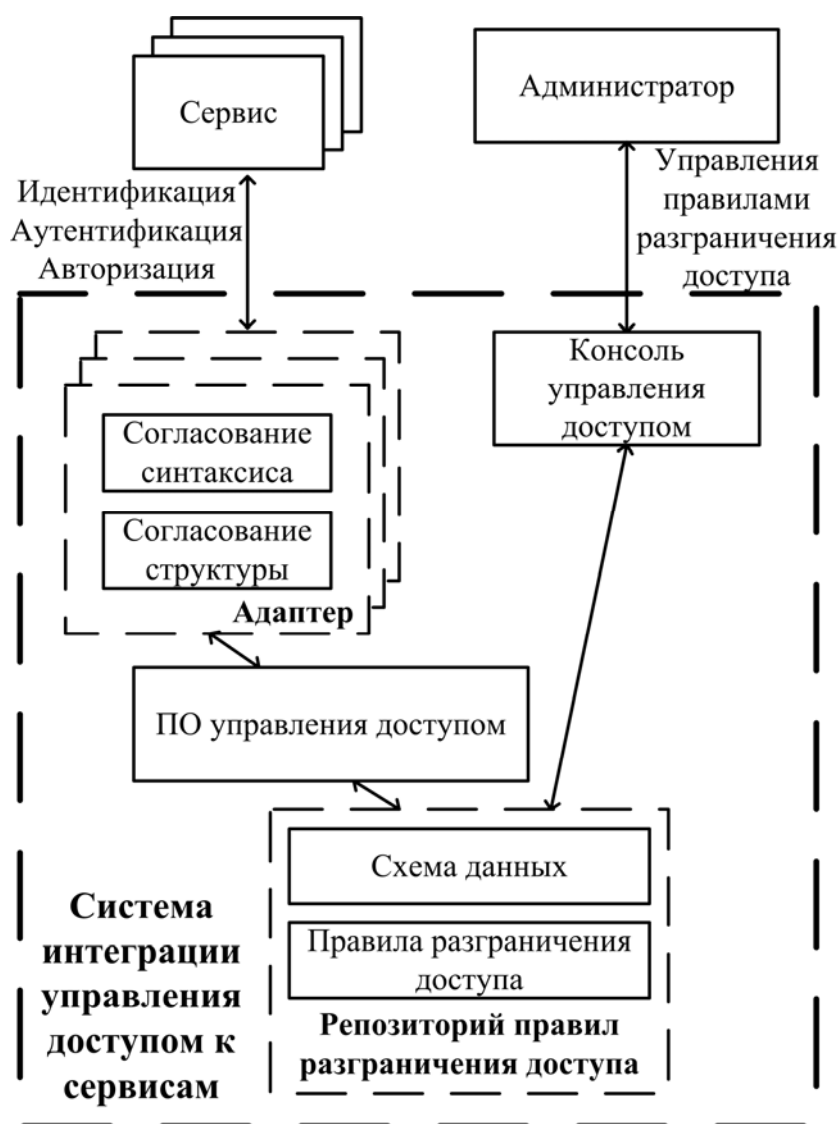


Рис 2.1. Схема прототипа

Прототип состоит из следующих компонентов:

- **Репозиторий правил разграничения доступа** хранит правила разграничения доступа к сервисам разных типов. Формат хранения правил разграничения доступа описывается общей схемой данных. Особенности конкретных

сервисов учитываются в схемах данных сервисов, специфичных для каждого типа сервисов. Между общей схемой данных и схемами данных сервисов устанавливается соответствие.

- **Программное обеспечение управления доступом** реализует процедуры защиты информации на основе данных из репозитория правил разграничения доступа с использованием общей схемы данных.

- **Адаптеры сервисов** обеспечивают подключение сервисов к системе управления доступом. Адаптеры состоят из двух блоков: согласование структуры и согласования синтаксиса. Блок согласования структуры обеспечивает преобразование структуры данных из формата, описываемого общей схемой данных, в формат, специфичный для каждого типа сервисов. Блок согласования синтаксиса обеспечивает преобразование протоколов для передачи данных по сети.

- **Консоль управления доступом** служит для управления правилами разграничения доступа к сервисам в репозитории правил разграничения доступа. Консоль является единым интерфейсом для управления доступом к сервисам всех типов.

С прототипом взаимодействуют:

- **Сервисы** разных типов обращаются к системе интеграции управления доступом для выполнения процедур защиты (идентификация, аутентификация, авторизация). Сервисы могут использовать различные протоколы и схемы данных управления доступом. Подключение сервисов к системе управления доступом осуществляется с помощью адаптеров сервисов, осуществляющих согласование протоколов и схем данных взаимодействия.

- **Администраторы** используют консоль для управления доступом к сервисам.

2.1.2 Недостатки прототипа

Основным недостатком прототипа является низкая интероперабельность, вызванная использованием структурной интеграции. Структурный подход ос-

нован на интеграции с применением схем данных: для интеграции нескольких систем разрабатывается общая схема данных и устанавливается соответствие между общей схемой и схемами конкретных систем. При появлении новой системы, которую нужно интегрировать, необходимо задать соответствие между общей схемой данных и схемой новой системы. В некоторых случаях задать такое соответствие очень сложно из-за существенных отличий в наборах понятий и операций интегрируемых систем. Часто при подключении новой системы требуется изменение общей схемы данных и последующий пересмотр соответствий общей схемы со схемами каждой системы.

На практике низкая интероперабельность проявляется в узком круге сервисов, интеграция управления доступом к которым возможна.

2.1.3 Предлагаемое решение

Для увеличения интероперабельности предлагается использовать семантическую интеграцию. Схема предлагаемого решения показана на рис. 2.2. Решение включает следующие новые блоки:

- **Онтология** управления доступом к сервисам, определяющую общую семантическую модель управления доступом.
- Блок **согласования семантики** в адаптере сервиса, обеспечивающий, дополнительно к согласованию синтаксиса и структуры, согласование семантики данных на основе онтологии.

Кроме введения новых блоков предлагается модифицировать существующие: ПО управления доступом, консоль управления, схемы данных и правила разграничения доступа, чтобы все перечисленные блоки могли использовать онтологию.



Рис 2.2. Схема прототипа и предлагаемого решения (новые блоки закрашены, развитые отмечены уголком)

2.2 Методика интеграции управления доступом к сервисам на основе семантического подхода

Для создания системы интеграции управления доступом к сервисам разных типов нужно выполнить следующие шаги:

1. Задать множество сервисов S , которые подлежат интеграции.
2. Задать множество моделей M , используемых для управления доступом к сервисам из множества S .

3. Задать множество репозиториев правил разграничения доступа R , которые подлежат интеграции.
4. Задать множество протоколов управления доступом P , которые используются сервисами из множества S .
5. Разработать унифицированную модель управления доступом M_0 . Модель включает семантическое описание базовых понятий и операций систем управления доступом (представленное в виде онтологии) и синтаксис записи правил разграничения доступа с использованием данной семантики.
6. Определить отображение моделей управления доступом $m_i \in M$ в унифицированную модель M_0 : $F: m_i \rightarrow M_0 | m_i \in M$.
7. Создать консолидированный репозиторий правил разграничения доступа R_0 , использующий унифицированную модель M_0 .
8. Разработать адаптеры AR , обеспечивающие перенос данных из репозиториев $r_i \in R$ в консолидированный репозиторий R_0 .
9. Создать программное обеспечение управления доступом, реализующее процедуры защиты информации с использованием правил разграничения доступа в консолидированном репозитории R_0 .
10. Разработать адаптеры AS протоколов управления доступом $p_i \in P$, обеспечивающие взаимодействие сервисов $s_i \in S$ с ПО управления доступом для выполнения процедур защиты информации с использованием протокола $p_i \in P$.
11. Настроить сервисы $s_i \in S$ для взаимодействия с ПО управления доступом с помощью адаптеров $a_i \in AS$ по протоколам $p_i \in P$.
12. Разработать систему управления правилами разграничения доступа в консолидированном репозитории R_0 .

Структурная схема системы семантической интеграции управления доступом к сервисам разных типов, получаемой с помощью предлагаемой методики, показана на рис. 2.3.

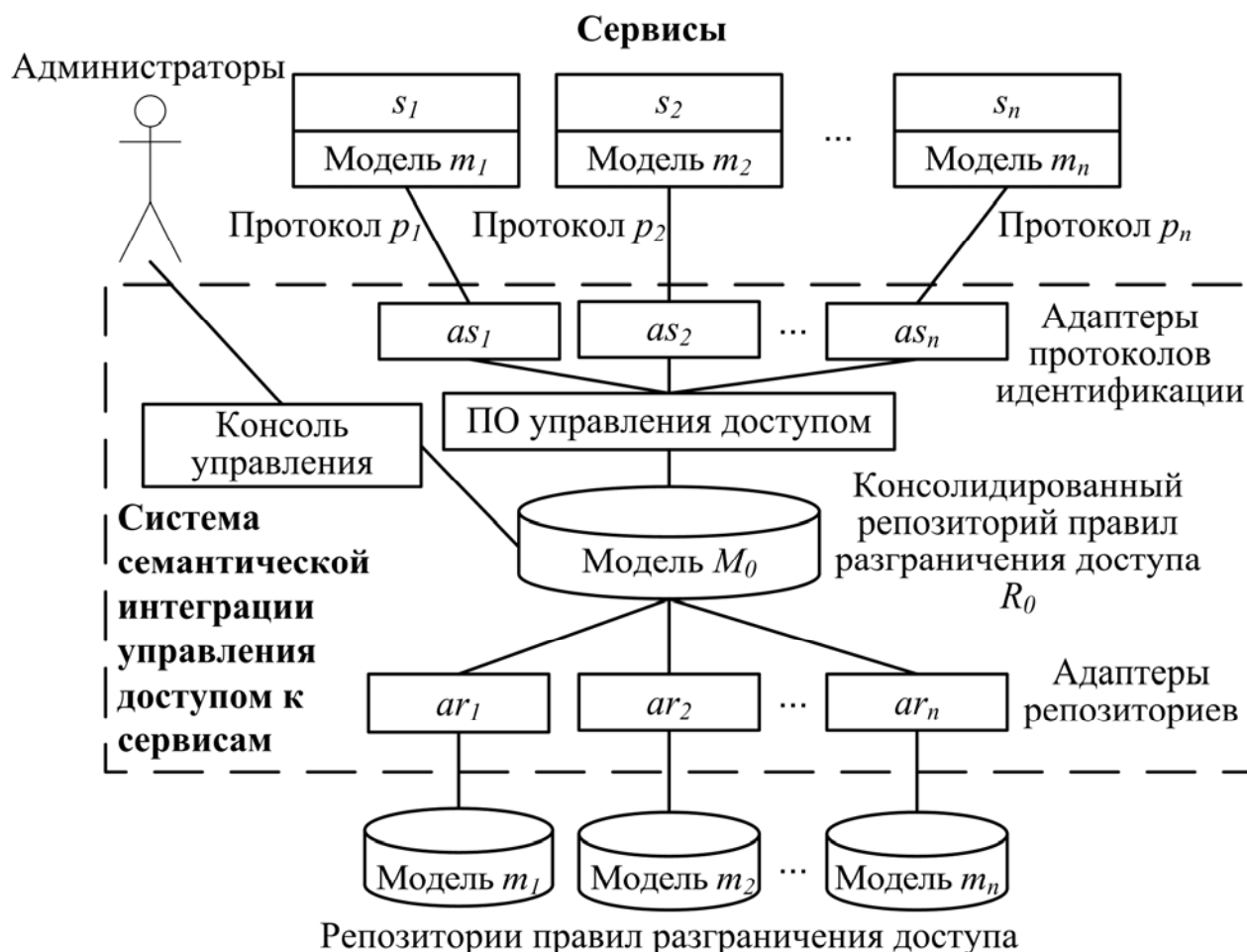


Рис 2.3. Структурная схема системы семантической интеграции управления доступом к сервисам

2.3 Выводы по главе

Анализ прототипа показал, что его основной недостаток – низкая интероперабельность, причиной которой является использование в прототипе структурного подхода к интеграции управления доступом к сервисам. В целях увеличения интероперабельности предлагается использовать семантический подход к интеграции. Построенная с применением данного метода система обеспечит интеграцию более широкого круга сервисов и использование большего количества методов управления доступом.

3 СИСТЕМА МОДЕЛЕЙ УПРАВЛЕНИЯ ДОСТУПОМ К СЕРВИСАМ

В данной главе представлена система моделей управления доступом к сервисам. В соответствии с методикой интеграции управления доступом к сервисам, предложенной во второй главе, система моделей состоит из двух частей:

- **Семантическая модель**, описывающая базовые понятия и операции предметной области управления доступом к сервисам.
- **Алгебраическая модель**, определяющая синтаксис записи правил разграничения доступа.

3.1 Онтология управления доступом к сервисам

Онтология управления доступом к сервисам состоит из двух частей, в соответствии с онтологией BWW. Первая часть содержит статическое описание систем управления доступом к сервисам и включает объекты предметной области управления доступом и их свойства. Вторая часть описывает динамическое поведение систем управления доступом и включает набор состояний таких систем, правила смены состояний и события, возникающие в системе.

3.1.1 Структура системы управления доступом к сервисам

Статическое описание структуры системы в терминах онтологии BWW включает объекты (thing), существующие в системе и их свойства (properties).

3.1.1.1 Объекты

Основные объекты разработанной онтологии управления доступом к сервисам показаны на UML-диаграмме (рис. 3.1.). В верхней части диаграммы представлены базовые понятия онтологии BWW, в нижней их сужение на предметную область управления доступом.

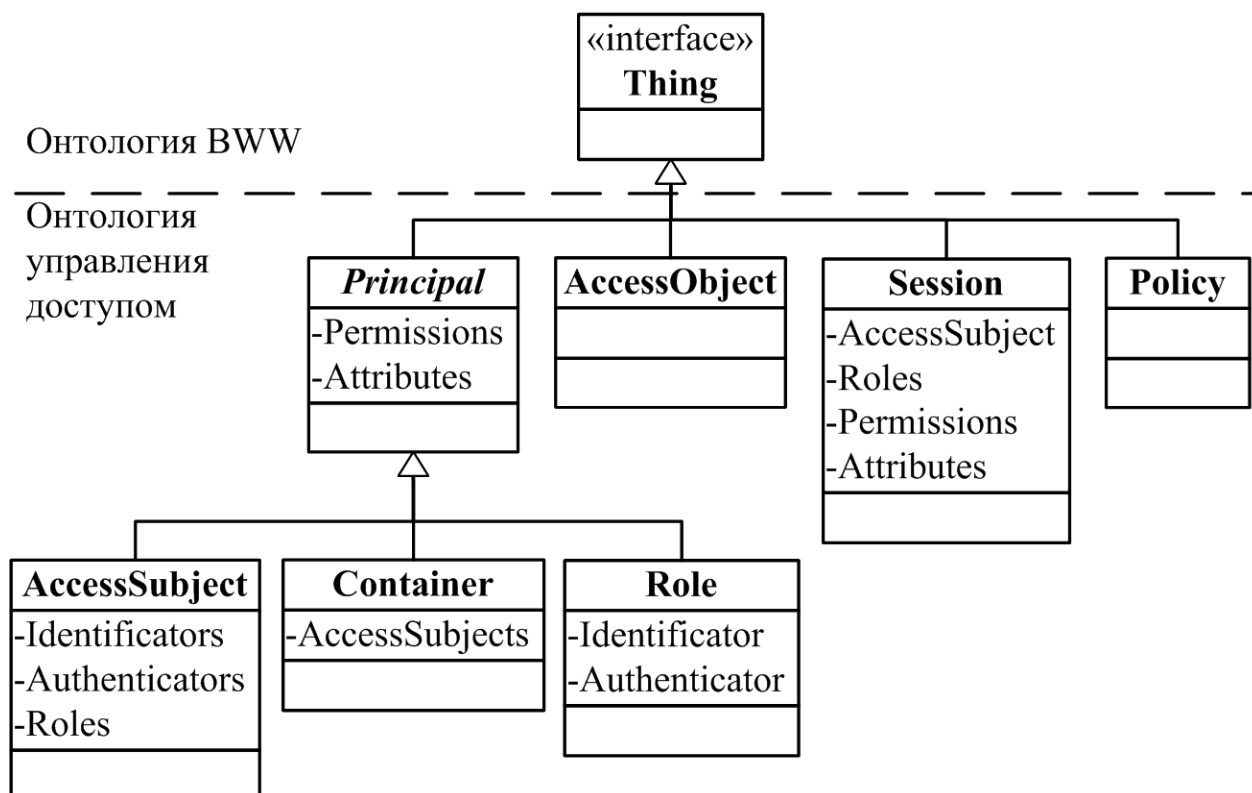


Рис 3.1. Структура систем управления доступом к сервисам. Объекты.

Основными объектами являются:

- **AccessObject** – объект доступа. Система управляет доступом к этому объекту.
- **Principal** – сущность, наделенная правами доступа к объектам доступа и атрибутами. Principal – абстрактный класс, который не может иметь экземпляров. Класс имеет три наследника: **AccessSubject** (субъект доступа), **Container** (контейнер) и **Role** (роль). Субъект доступа представляет собой персону, программу или систему, взаимодействующую с объектом доступа. Контейнеры и роли введены для упрощения процесса управления.
- **Policy** – политика управления доступом, включающая данные об объектах доступа, субъектах доступа и права доступа субъектов к объектам.
- **Session** – сессия, возникающая при обращении субъекта к объекту.

Ключевым компонентом онтологии является субъект доступа. Ему назначаются идентификатор и аутентификатор (возможно, несколько). Субъекту назначаются права доступа к объектам, а также описательные атрибуты (ФИО и контактные данные для персоны, название и адрес сервера для системы и т.п.).

Назначение всех необходимых прав каждому субъекту в крупных системах является трудоемкой задачей. Для ее упрощения в онтологию вводятся понятия контейнеров и ролей [23]. Контейнеру и роли можно назначить набор прав доступа и атрибутов, после чего быстро выделить их нужному субъекту путем включения в контейнер или назначения роли.

Основное отличие ролей от контейнеров – динамическая природа ролей. Права доступа и атрибуты контейнеров, в которые он включен, доступны пользователю всегда, а чтобы задействовать права доступа роли ее надо активировать в сессии.

Сессия возникает при обращении субъекта к одному или нескольким объектам доступа. Сессия включает субъект доступа, список активированных ролей и набор прав доступа и атрибутов, действующих в сессии. В этот набор входят права доступа и атрибуты, назначенные непосредственно субъекту, контейнерам, в которые он включен, а также ролям, которые субъект активировал в сессии.

3.1.1.2 Свойства

Основные свойства разработанной онтологии управления доступом к сервисам показаны на UML-диаграмме (рис. 3.2.). В верхней части диаграммы представлены базовые понятия онтологии BWW, в нижней их сужение на предметную область управления доступом.

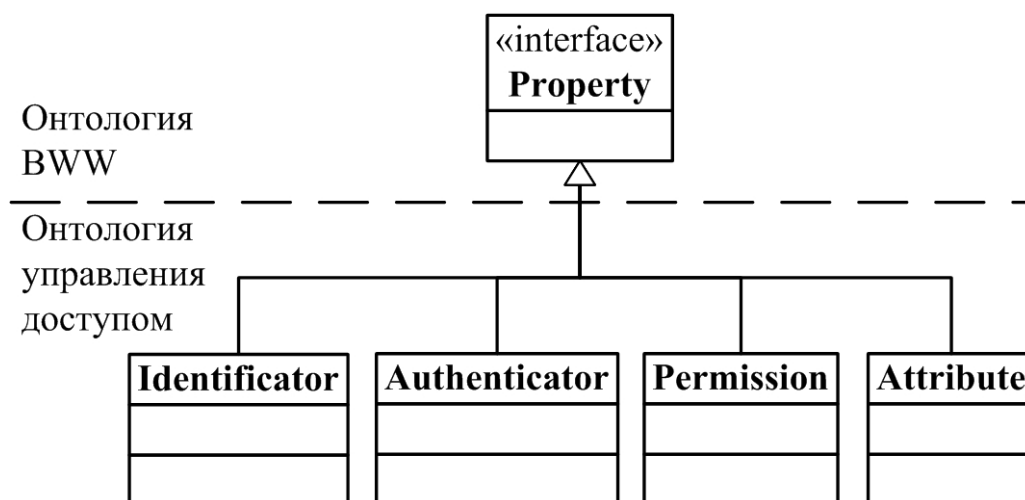


Рис 3.2. Структура систем управления доступом к сервисам. Свойства.

Основными свойствами являются:

- **Identifier** – уникальный признак объекта, позволяющий однозначно выделить объект среди всего множества существующих объектов.
- **Authenticator** – аутентификатор, позволяющий определить подлинность объекта.
- **Permission** – права доступа субъекта к объекту. Онтология не накладывает никаких ограничений на семантику и синтаксис прав доступа. Анализ прав и принятие решения о разрешении или запрещении доступа – это задача каждого конкретного сервиса. Такой подход делает возможным управление доступом к широкому кругу сервисов с различными требованиями к семантике и форматам представления прав доступа.
- **Attribute** – атрибут, содержащий описательную информацию. Как и в случае с правами доступа, на семантику и синтаксис атрибутов не накладывается никаких ограничений. В частности, допускается использовать атрибуты не только для целей описания объектов, но и для управления доступом на основе атрибутов [24], при наличии соответствующей поддержки со стороны сервиса.

3.1.2 Динамика управления доступом

Для описания динамического поведения системы в онтологии BWB используются состояния, события и операции.

3.1.2.1 Состояния

Состояния системы управления доступом приведены на UML-диаграмме (рис.3.3.).

Состояния системы управления доступом являются сужением понятия состояния объекта State онтологии BWB на предметную область управления доступом. Определяется пять состояний:

- **SubjectUnknown** – субъект доступа неизвестен.
- **SubjectIdentified** – субъект доступа идентифицирован.
- **SubjectAuthenticated** – подлинность субъекта доступа подтверждена.

- **SubjectAuthorized** – проверка прав доступа субъекта к объекту выполнена.

- **SessionExists** – состояние, введенное для реализации однократной регистрации. Означает, что субъект прошел идентификацию и аутентификацию, и для субъекта создана сессия.

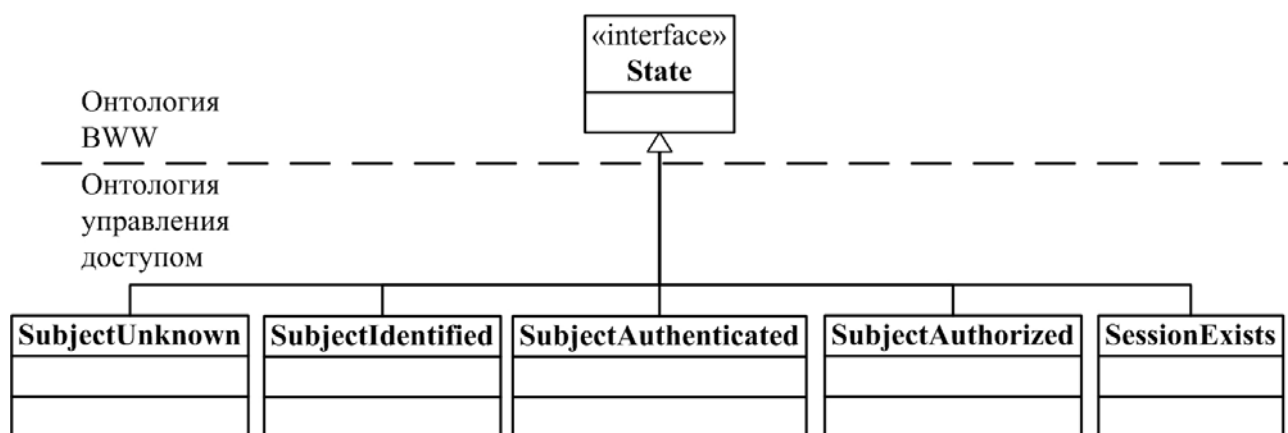


Рис 3.3. Состояния управления доступом к сервисам

3.1.2.2 События

События системы управления доступом приведены на UML-диаграмме (рис.3.4.).

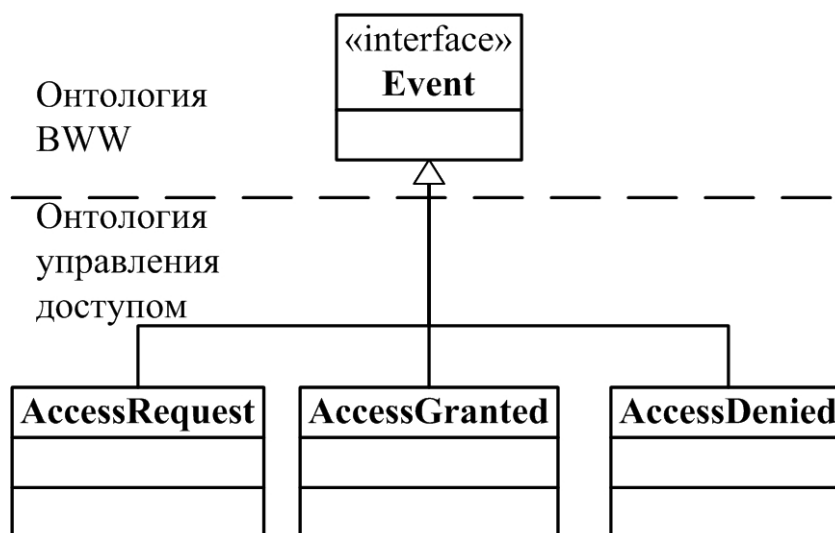


Рис 3.4. События управления доступом к сервисам

События системы управления доступом являются сужением понятия состояния объекта Event онтологии BWW на предметную область управления доступом. Определено три основных события:

– **AccessRequest** – возникает при обращении субъекта доступа к объекту доступа. С появлением этого события начинаются процедуры контроля доступа.

– **AccessGranted** – событие, возникающее в случае, если в результате работы процедур контроля принято решение разрешить доступ.

– **AccessDenied** – событие, возникающее в случае, если в результате работы процедур контроля принято решение запретить доступ.

3.1.2.3 Операции

Правила преобразования состояний (операции) системы управления доступом к сервисам показаны на рис 3.5.

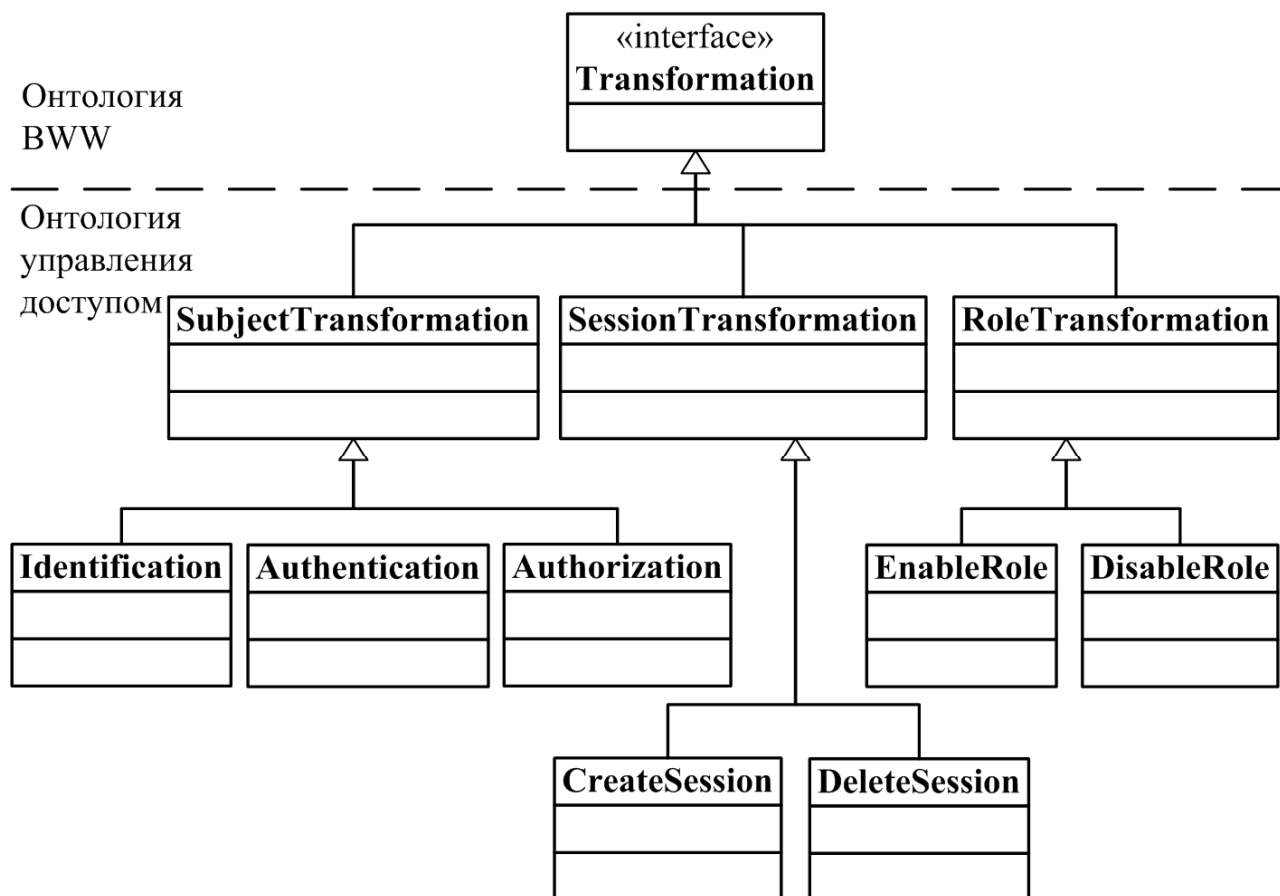


Рис 3.5. Операции управления доступом к сервисам

Операции, выполняемые при управлении доступом, разделены на три типа:

– **SubjectTransformation** – операции с субъектом доступа.

– **SessionTransformation** – операции с сессией.

- **RoleTransformation** – операции с ролью.

С объектом доступа возможны операции идентификации (**Identification**), аутентификации (**Authentication**) и авторизации (**Authorization**). При этом под авторизацией понимается проверка прав доступа субъекта, действующих в текущей сессии.

Операция создания сессии **CreateSession** включает в себя следующие действия:

- Запоминается субъект доступа, для которого создается сессия. Пока сессия действует субъекту не надо будет повторно проходить идентификацию и аутентификацию.
- В набор действующих в сессии прав доступа и атрибутов включаются права доступа и атрибуты субъекта.
- В набор действующих в сессии прав доступа и атрибутов включаются права доступа и атрибуты контейнеров, в которые входит субъект.

Операция удаления сессии **DeleteSession** выполняет действия противоположные операции **CreateSession**:

- Очищается список действующих в сессии прав доступа.
- Стирается информация о субъекте доступа.

Удаление сессии приводит к необходимости прохождения субъектом идентификации и аутентификации до начала работы с субъектом доступа.

Операция активации роли **EnableRole** добавляет название роли в список действующих в сессии ролей и включает права доступа и атрибуты роли в список прав доступа и атрибутов, действующих в сессии. Операция удаления роли **DisableRole** выполняет противоположные действия.

Операция авторизации выполняет проверку прав доступа, действующих в сессии, куда, кроме прав доступа субъекта, входят права доступа контейнеров, в которые включается субъект, а также права активированных в сессии ролей.

3.1.3 Диаграмма состояний системы управления доступом к сервисам

Диаграмма состояний системы управления доступом к сервисам (основной успешный сценарий) показана на рис. 3.6.

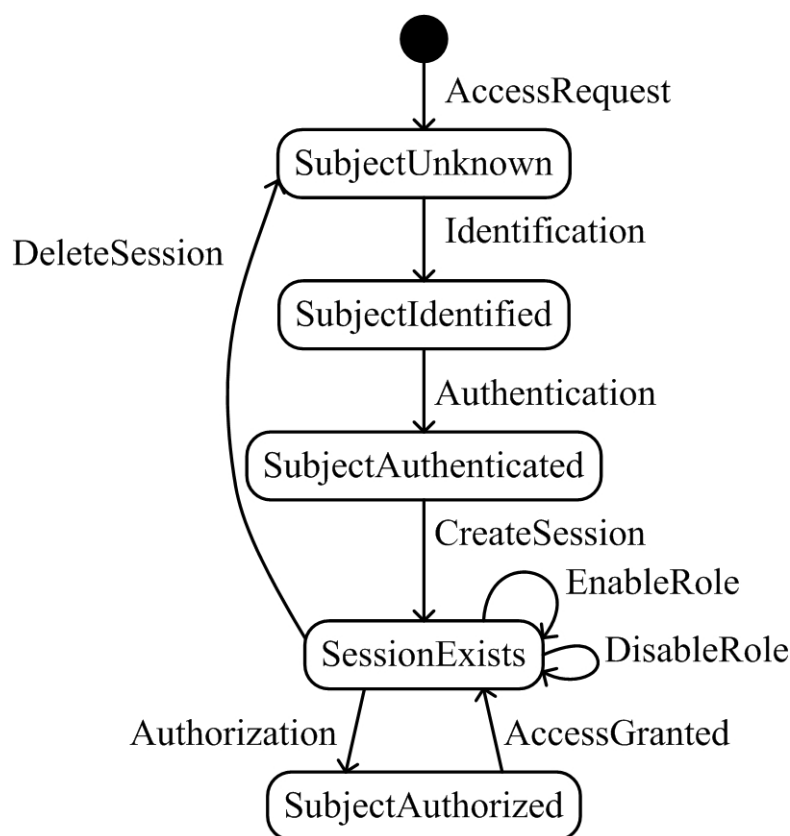


Рис 3.6. Диаграмма состояний системы управления доступом к сервисам

Работа системы начинается с появления события **AccessRequest** – запроса субъекта на доступ к объекту. При этом какая-либо информация о субъекте пока отсутствует (состояние **SubjectUnknown**). На первом этапе выполняется идентификация субъекта (действие **Identification**, переход в состояние **SubjectIdentified**). Затем проверяется подлинность субъекта (действие **Authentication**, переход в состояние **SubjectAuthenticated**). Когда субъект доступа достоверно известен, создается сессия (действие **CreateSession**, переход в состояние **SessionExists**). После создания сессии проверяется проверка прав доступа субъекта к объекту (операция **Authorization**, переход в состояние **SubjectAuthorized**). В случае успешной проверки прав, доступ субъекта к объекту разрешается (возникает событие **AccessGranted**) и выполняется переход в состояние **SessionExists**.

В случае появления события `AccessRequest`, когда система находится в состоянии `SessionExists`, в котором субъект доступа достоверно известен, сразу выполняется операции авторизации, без операции идентификации и аутентификации (однократная регистрация).

В состоянии `SessionExist` субъект может активировать роль, расширив таким образом свои права доступа (действие `EnableRole`). После завершения выполнения операций, требующих прав доступа роли, роль можно деактивировать (действие `DisableRole`).

Действие `DeleteSession` выполняет уничтожение сессии. Вместе с сессией уничтожается информация о субъекте доступа, для которого создавалась сессия и выполняется переход в состояние `SubjectUnknown`. Действие `DeleteSession` выполняется по запросу субъекта («выход из системы»), либо автоматически через заданное время с момента создания сессии («время жизни» сессии).

3.2 Алгебраическая запись правил разграничения доступа к сервисам

Формализация управления доступом в виде онтологической модели позволяет полностью описать семантику базовых понятий управления доступом к сервисам. При этом полученная семантическая модель не определяет представления описания прав доступа к сервисам в форме удобной для анализа человеком или автоматизированными анализаторами. В связи с этим была разработана формальная алгебраическая запись политик доступа к сервисам, семантика которых определена онтологической моделью.

Определено несколько уровней алгебраической записи с разными выразительными возможностями. Базовый уровень содержит основные компоненты, необходимые для управления доступом к сервисам. Первый уровень позволяет использовать контейнеры, второй – ролевое управление доступом, а третий обеспечивает возможность делегирования полномочий управления доступом к сервисам. Компоненты модели более высокого уровня включают все компоненты моделей уровней ниже.

3.2.1 Базовый уровень

Базовый уровень алгебраической записи правил разграничения доступа включает основные понятия, необходимые для управления доступом к сервисам, показанные на рис. 3.7.



Рис 3.7. Базовый уровень алгебраической записи правил разграничения доступа

Алгебраическая запись включает 7 множеств: субъектов (U), идентификаторов (I), аутентификаторов (A), прав доступа (P), атрибутов (V), сессий (S) и ограничений. Семантика множеств алгебраической записи определяется онтологической моделью системы управления доступом. Субъекты доступа из множества U выполняют взаимодействие с объектами доступа. Каждому субъекту соответствует набор уникальных идентификаторов, принадлежащий множеству I . Идентификаторы служат для однозначного распознавания субъектов среди

всех элементов множества U . С помощью аутентификаторов из множества A , субъект может подтвердить, что он именно тот, за кого себя выдает.

Множество P содержит все возможные права доступа к сервисам. Между этим множеством и множеством субъектов U установлено отношение «многие-ко-многим», определяющее права доступа каждого конкретного пользователя.

Множество V содержит все атрибуты, используемые при доступе к сервисам (например, максимальный размер почтового ящика или скорость доступа в Интернет). Между множеством атрибутов V и множеством субъектов U установлено отношение «многие-ко-многим», задающее атрибуты каждого субъекта.

Алгебраическая запись не накладывает никаких ограничений на синтаксис и семантику прав доступа и атрибутов. Интерпретацией прав доступа и атрибутов занимаются сервисы. Такой подход обеспечивает возможность управления доступом к широкому кругу сервисов с разными требованиями к наборам прав доступа и атрибутов субъектов.

Иерархии позволяют структурировать определения прав доступа и атрибутов, выражая одни понятия через другие. Иерархия задается путем введения на множестве частичного порядка и графически представляется в виде диаграммы Хассе [79] (рис. 3.8). Права доступа, расположенные выше на диаграмме, включают все права, расположенные ниже.

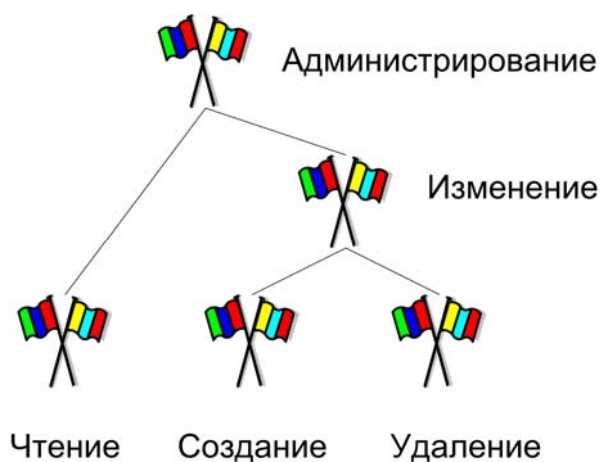


Рис 3.8. Пример иерархии прав доступа

Сессия (множество S) создается (как определяет онтологическая модель), когда субъект при первом обращении к объекту проходит идентификацию и ау-

тентификацию. Идентификатор и аутентификатор пользователя запоминаются в сессии и не требуют повторного ввода при следующих обращениях к сервисам. Сессии позволяют реализовать однократную регистрацию.

В целях гибкого управления доступом вводятся ограничения на допустимые сочетания компонентов модели и некоторые их свойства. Примерами важных ограничений являются: «время жизни» сессии, зависимость набора прав доступа от времени суток, использованного метода аутентификации (парольная, двухфакторная, биометрическая) или типа соединения (с шифрованием или без него).

Приведем формальное определение базового уровня алгебраической записи политик.

Определение 1. Базовый уровень алгебраической записи политик управления доступом к сервисам включает (рис. 3.7):

- Множества U, I, A, P, V, S (субъектов, идентификаторов, аутентификаторов, прав доступа, атрибутов и сессий соответственно).
- $user: I \rightarrow U$ – функция, отображающая каждый идентификатор i_j в единственный субъект доступа $user(i_j)$.
- $K(i_u) = \{i_j \mid user(i_j) = u\}$ – классы эквивалентности идентификаторов i_u , задающее разбиение множества идентификаторов I на подмножества идентификаторов, принадлежащих одному субъекту u .
- $id: A \rightarrow I$ – функция, отображающая каждый аутентификатор a_j в единственный идентификатор $id(a_j)$.
- $UP \subseteq U \times P$ – отношение, задающее соответствие между субъектами и правами доступа.
- $UV \subseteq U \times V$ – отношение, задающее соответствие между субъектами и атрибутами.
- $suser: S \rightarrow U$ – функция, отображающая каждую сессию s_j в единственный субъект $suser(s_j)$.

- $sid : S \rightarrow I$ – функция, отображающая каждую сессию s_j в единственный идентификатор $sid(s_j)$.
- $sauth : S \rightarrow A$ – функция, отображающая каждую сессию s_j в единственный аутентификатор $sauth(s_j)$.
- $PH \subseteq P \times P$ – частичный порядок на множестве прав доступа P , называемый иерархией прав доступа и обозначаемый \geq .
- $VH \subseteq V \times V$ – частичный порядок на множестве атрибутов V , называемый иерархией атрибутов и обозначаемый \geq .
- $perm : U \rightarrow 2^P$ – функция, ставящая в соответствие субъекту u_j множество прав доступа $perm(u_j) \subseteq \left\{ p \mid (\exists p' \geq p) \text{ и } ((u_j, p') \in UP) \right\}$
- $attr : U \rightarrow 2^V$ – функция, ставящая в соответствие субъекту u_j множество атрибутов $attr(u_j) \subseteq \left\{ v \mid (\exists v' \geq v) \text{ и } ((u_j, v') \in UC) \right\}$
- Множество ограничений, определяющих, какие сочетания компонент модели являются допустимыми. Разрешены только допустимые компоненты. ■

3.2.2 Контейнеры

В целях упрощения процесса управления доступом вводится понятие контейнера для субъектов (рис. 3.9.). Распространенными примерами контейнеров являются организация или группа. Права доступа и атрибуты назначаются не только напрямую пользователям, но и контейнерам. Все пользователи, входящие в контейнер, получают права доступа и атрибуты контейнера.

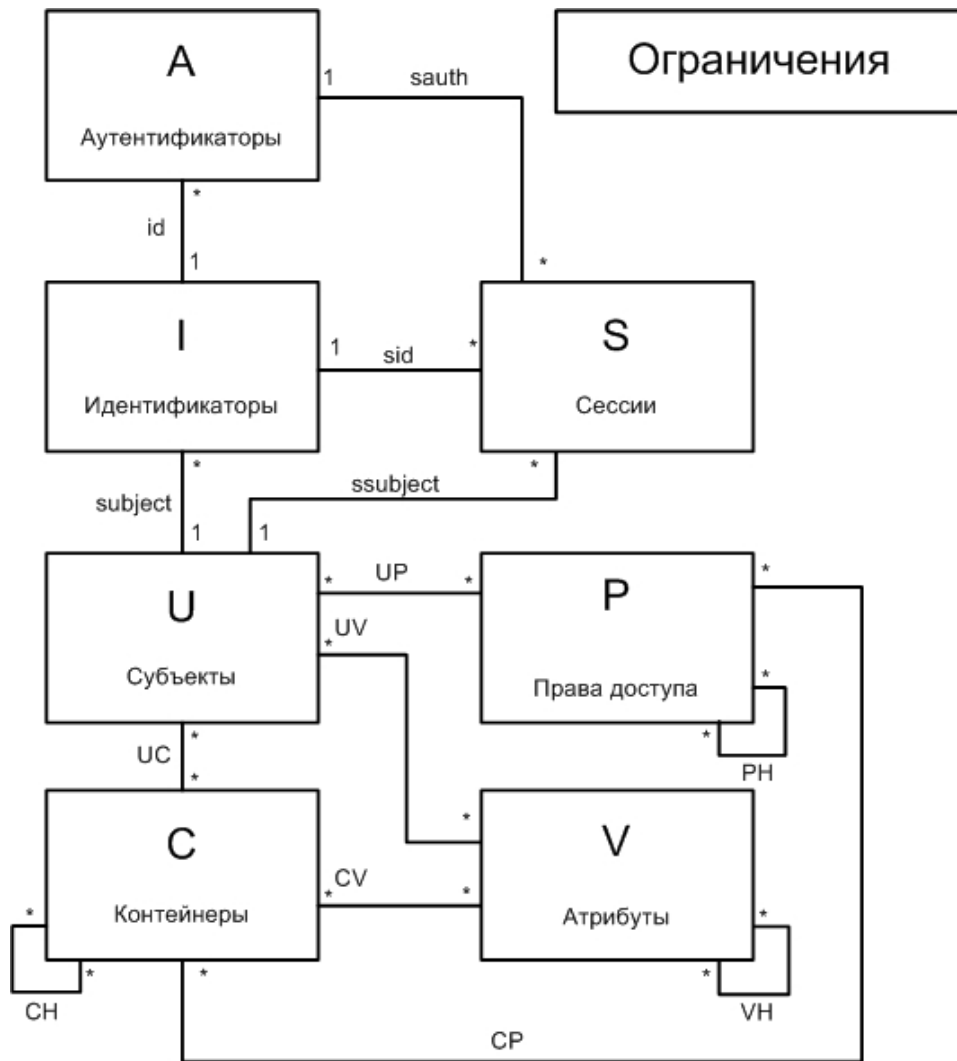


Рис 3.9. Алгебраическая запись правил разграничения доступа. Уровень 1.

Определение 2. Алгебраическая запись правил разграничения доступа к сервисам уровня 1 включает (рис. 3.9.):

- $U, I, A, P, V, S, user, K(i_u), id, UP, UV, suser, sid, sauth, PH, VH$ и ограничения, те же самые, что и в базовом уровне алгебраической записи.
- Множество контейнеров C .
- $CH \subseteq C \times C$ – частичный порядок на множестве контейнеров C , называемый иерархией контейнеров и обозначаемый \geq .
- $CP \subseteq C \times P$ – отношение, задающее соответствие между контейнерами и правами доступа.
- $CV \subseteq C \times V$ – отношение, задающее соответствие между контейнерами и атрибутами.

– $UC \subseteq U \times C$ – отношение, задающее соответствие между субъектами и контейнерами.

– $containers: U \rightarrow C^2$ – функция, ставящая в соответствие субъекту u_j множество контейнеров $containers(u_j) \subseteq \left\{ c \mid (\exists c' \geq c) \text{ и } ((u_j, c') \in UC) \right\}$.

– $perm: U \rightarrow 2^P$ – функция, измененная по сравнению с алгебраической записью первого уровня:

$$perm(u_j) \subseteq \bigcup_{c \in containers(u_j)} \left\{ p \mid (\exists p' \geq p) \text{ и } ((c, p') \in CP) \right\} \cup \left\{ p \mid (\exists p' \geq p) \text{ и } ((u_j, p') \in UP) \right\}$$

– $attr: U \rightarrow 2^V$ – функция, измененная по сравнению с алгебраической записью первого уровня:

$$attr(u_j) \subseteq \bigcup_{c \in containers(u_j)} \left\{ v \mid (\exists v' \geq v) \text{ и } ((c, v') \in CV) \right\} \cup \left\{ v \mid (\exists v' \geq v) \text{ и } ((u_j, v') \in UC) \right\} \blacksquare$$

3.2.3 Роли

Современный подход к управлению доступом состоит в использовании ролей [23], которые соответствуют выполняемым обязанностям в организации. Права доступа и атрибуты назначаются ролям, а роли распределяются между пользователями (рис. 3.10.). Использование ролей не запрещает назначения прав доступа и атрибутов напрямую пользователям и контейнерам, эти механизмы допускают совместное применение.

Основное отличие от контейнеров – это динамическая природа ролей. Права доступа контейнера всегда доступны пользователю. А чтобы использовать права доступа роли, ее необходимо активировать в сессии.

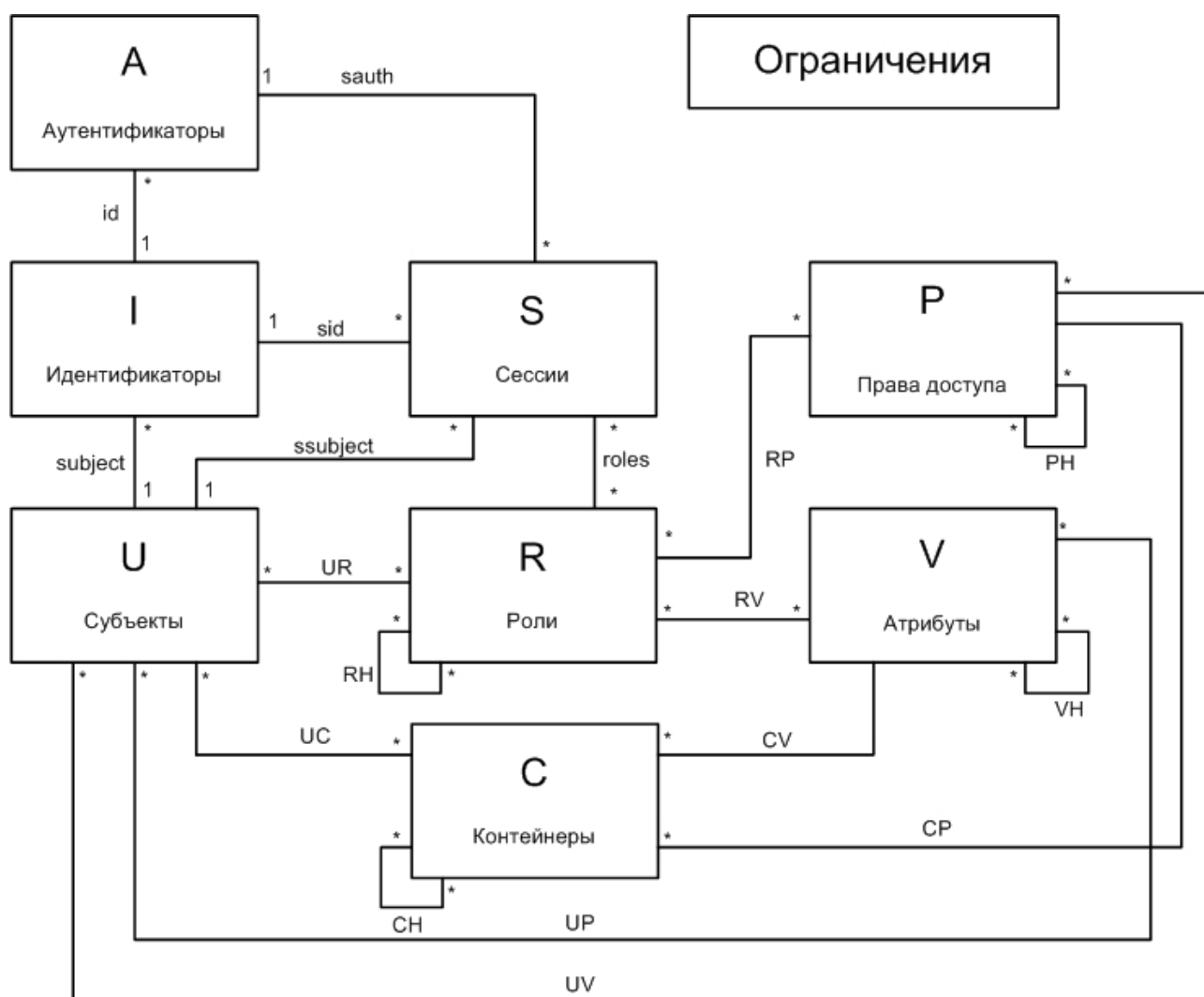


Рис 3.10. Алгебраическая запись правил разграничения доступа. Уровень 2.

Формальное определение второго уровня алгебраической записи правил разграничения доступа к сервисам приведено ниже.

Определение 3. Алгебраическая запись правил разграничения доступа к сервисам уровня 2 включает (рис. 3.10):

- $U, I, A, P, V, S, C, user, K(i_u), id, UP, UV, UC, CP, CV, suser, sid, sauth, PH, VH, CH, containers, perms, attrs$ и ограничения, те же самые, что и алгебраической записи уровня 1.
- Множество ролей R .
- $UR \subseteq U \times R$ отношение, задающее соответствие между субъектами и ролями.
- $RP \subseteq R \times P$ – отношение, задающее соответствие между ролями и правами доступа.

– $RC \subseteq R \times C$ – отношение, задающее соответствие между ролями и атрибутами.

– $RH \subseteq R \times R$ – частичный порядок на множестве ролей R , называемый иерархией ролей и обозначаемый \geq .

– $roles : S \rightarrow 2^R$ – функция, ставящая в соответствие сессии s_j множество ролей

$roles(s_j) \subseteq \left\{ r \mid (\exists r' \geq r) \text{ и } (users(s_j, r') \in UR) \right\}$ и субъект в сессии s_j получает

дополнительные права $\bigcup_{r \in roles(s_j)} \{ p \mid (\exists p' \geq p) \text{ и } (\exists r' \geq r) \text{ и } (ur', p') \in RP) \}$ и до-

полнительные атрибуты $\bigcup_{r \in roles(s_j)} \{ v \mid (\exists v' \geq v) \text{ и } (\exists r' \geq r) \text{ и } (ur', v') \in RV) \}$ ■

3.2.4 Делегирование полномочий управления доступом к сервисам

В приведенных выше определениях считается, что управление доступом ведется единым администратором. Такой подход неприемлем для крупных организаций. Стандартным методом распределения обязанностей управления между несколькими администраторами является делегирование полномочий управления доступом к сервисам. Для поддержки делегирования вводятся множества административных прав доступа (создание пользователей, назначением им прав доступа и атрибутов, ролей и т.д.) и административных ролей. Административные роли образуют иерархию, в вершине которой один глобальный администратор, наделенный всеми административными правами. Ниже по иерархии расположены администраторы с более узким набором административных прав. Такие администраторы могут быть ответственны за подмножество субъектов (например, сотрудников одной организации), за распределение прав доступа к определенному типу сервиса (Интернет, электронная почта, вычислительный кластер и т.п.), или назначение определенных ролей. В рамках своей «зоны ответственности», администраторы могут в свою очередь делегировать часть своих полномочий другим администраторам.

Определение 4. Алгебраическая запись политик управления доступом к сервисам уровня 2 включает (рис. 3.11):

– $U, I, A, P, V, S, C, R, user, K(i_u), id, UP, UV, UC, UR, CP, CV, RP, RV, suser, sid, sauth, PH, VH, CH, RH, containers, roles, perms, attrs$ и ограничения, и в те же самые, что и алгебраической записи уровня 2.

– AR – множество административных ролей.

– AP – множество административных прав доступа.

– $UAR \subseteq U \times AR$ – отношение, задающее соответствие между субъектами доступа и административными ролями.

– $ARP \subseteq AR \times AP$ – отношение, задающее соответствие между административными ролями и административными правами доступа.

– $ARH \subseteq AR \times AR$ – частичный порядок на множестве административных ролей AR , называемый иерархией административных ролей и обозначаемый \geq .

– $APH \subseteq AP \times AP$ – частичный порядок на множестве административных прав доступа AP , называемый иерархией административных прав доступа и обозначаемый \geq .

– $aroles: S \rightarrow 2^{AR}$ – функция, ставящая в соответствии сессии s_j множество ад-

министративных ролей $aroles(s_j) \subseteq \left\{ r \mid (\exists r' \geq r) u) u (us(s_j, r') \in UAR) \right\}$ и

субъект доступа в сессии s_j получает административные права

$\bigcup_{r \in aroles(s_j)} \{ p \mid (\exists p' \geq p) u) u \exists r' \geq r) u) (r', p') \in ARP) \}$ ■

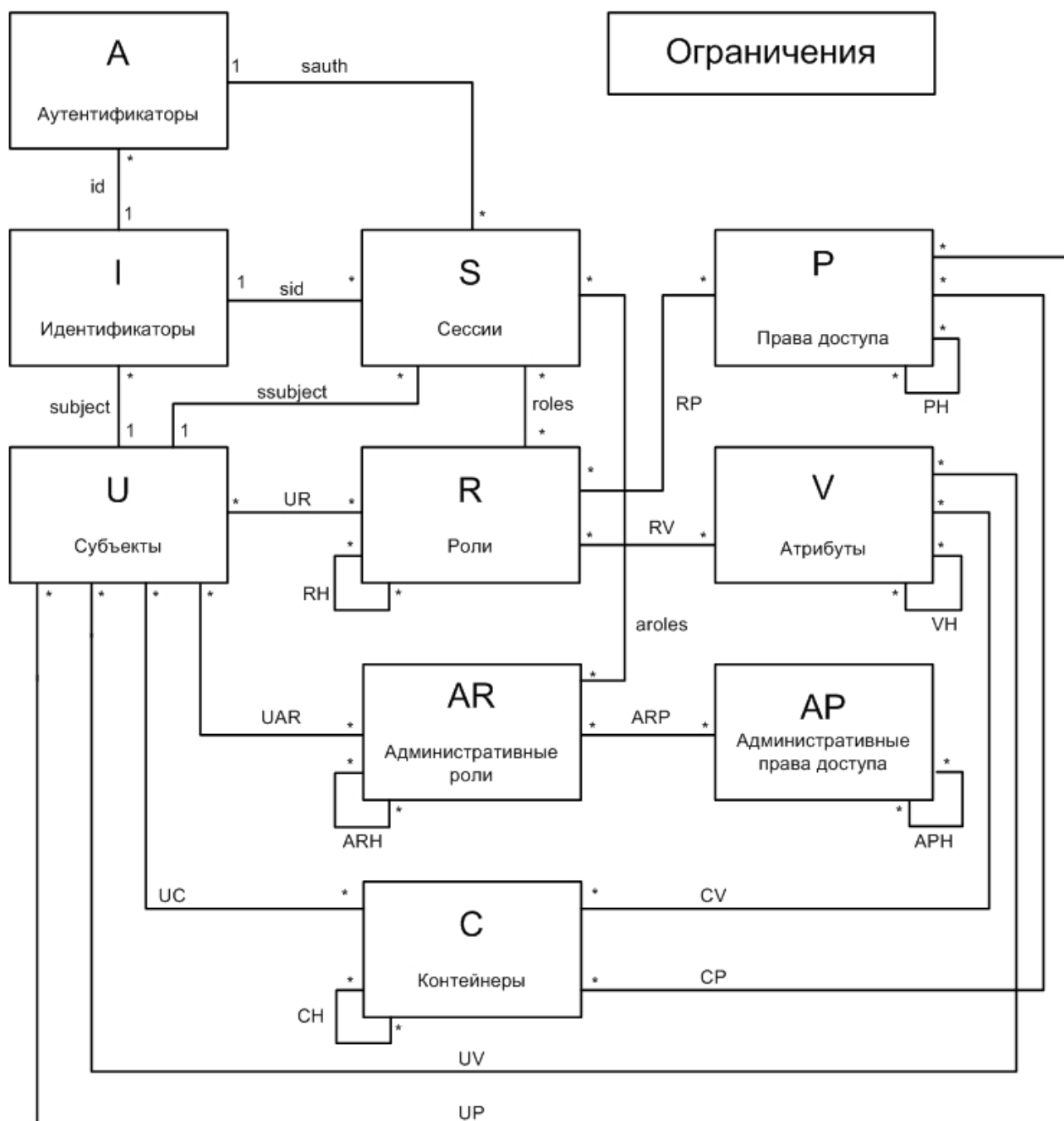


Рис 3.11. Алгебраическая запись правил разграничения доступа. Уровень 3.

3.2.5 Применение методов авторизации в алгебраической записи

Описанная алгебраическая запись допускает применение различных методов авторизации, облегчающих процесс управления, вместо назначения прав каждому конкретному пользователю. Поддерживается три метода авторизации: дискреционный в виде списков возможностей (Capability Lists), управление на основе ролей [23] и управление на основе атрибутов [24].

Дискреционный метод управления доступом поддерживается уже в базовом уровне алгебраической записи. Используется вариант дискреционного управления на основе списков возможностей, при котором права доступа хранятся совместно с субъектами доступа. Практическая реализация в виде комплекса программ по управлению доступом позволяет также использовать списки контроля доступа (ACL), что требует специальной поддержки со стороны сервиса.

На первом уровне интеграции вводятся контейнеры, позволяющие назначать права доступа с помощью дискреционного подхода не каждому отдельному субъекту, а заданному списку субъектов. Это позволяет упростить процесс выделения одинаковых прав доступа большому количеству субъектов.

Управление на основе ролей включено в алгебраическую запись в явном виде, начиная с уровня 2.

Управление доступом на основе атрибутов позволяет сервису принимать решение о разрешении или запрещении доступа на основе атрибутов субъекта доступа. Алгебраическая запись содержит не только множество прав доступа субъектов P , но и множество атрибутов V . При этом никаких ограничений на синтаксис и семантику атрибутов не накладывается. Это делает возможным применения атрибутов для управления доступом к сервисам. Логику принятия решений о разрешении или запрещении доступа на основе атрибутов должен обеспечить сам сервис.

3.3 Выводы по главе

Разработана система моделей управления доступом к сервисам, состоящая из онтологии управления доступом и алгебраической записи правил разграничения доступа к сервисам. Онтология задает семантику базовых понятия и операций предметной области управления доступом к сервисам. Алгебраическая запись правил разграничения доступа позволяет описывать правила разграничения доступа в формальном виде с использованием разных методов управления доступом: дискреционного, ролевого, атрибутного.

Предложенная система моделей используется в качестве унифицированной модели управления доступом в методике семантической интеграции, разработанной во второй главе, и служит основой для интеграции механизмов управления доступом к сервисам разных типов.

4 РЕАЛИЗАЦИЯ И ПРИМЕНЕНИЕ РАЗРАБОТАННЫХ МЕТОДОВ И ТЕХНОЛОГИЙ

4.1 Комплекс программ по управлению доступом к сервисам

На основе предложенного метода семантической интеграции управления доступом разработан комплекс программ по управлению доступом к сервисам разных типов.

4.1.1 Функции комплекса программ

Комплекс программ по управлению доступом к сервисам предоставляет следующие функции:

- единый репозиторий правил разграничения доступа к сервисам разных типов;
- идентификация субъектов доступа для сервисов разных типов.
- аутентификация субъектов доступа для сервисов разных типов;
- авторизация субъектов доступа для сервисов разных типов;
- однократная регистрация субъектов доступа для сервисов разных типов;
- учет использования сервисов;
- единая консоль управления доступом к сервисам разных типов;
- делегирование полномочий управления доступом к сервисам;
- интеграция с внешними информационными системами.

4.1.2 Уровни интеграции

Определено три уровня интеграции управления доступом к сервисам:

- **Нулевой уровень** – сервис использует данные о субъектах доступа из комплекса программ. Уровень применяется для интеграции с внешними информационными системами, которым требуется информация о субъектах доступа. Примером может быть система статистики использования сервисов.

- **Первый уровень** – сервис использует данные о субъектах и механизмы идентификации и аутентификации комплекса программ. Права доступа описываются, хранятся и анализируются средствами сервиса. Характерен для сервисов, использующих ACL, связанные с объектами доступа. Типичным примером являются Unix-системы, где права доступа задаются для каждого файла и каталога и хранятся в файловой системе. Идентификация и аутентификация пользователей Unix выполняется подключаемыми модулями аутентификации (Pluggable Authentication Modules, PAM), которые можно настроить на работу с комплексом программ по управлению доступом к сервисам.

- **Второй уровень** – полная интеграция, сервис использует данные о субъектах, механизмы идентификации, аутентификации и авторизации комплекса программ по управлению доступом. Примером использования интеграции второго уровня может служить сервер RADIUS, работающий в режиме прокси. В этом режиме сервер RADIUS принимает запросы от сервисов и передает их комплексу программ по управлению доступом к сервисам. Комплекс программ проводит идентификацию, аутентификацию и авторизацию субъекта доступа, выполняет поиск конфигурационной информации и передает все необходимые данные серверу RADIUS, который пересылает их сервису.

4.1.3 Логическая архитектура

Логическая архитектура комплекса программ по управлению доступом к сервисам показана на рис. 4.1. и состоит из трех уровней:

- **Уровень технических служб** обеспечивает взаимодействие с репозиториями правил разграничения доступа, обеспечивает создание консолидированного репозитория правил разграничения доступа.

- **Уровень приложений** реализует прикладную логику управления доступом: идентификацию, аутентификацию, авторизацию, управление сессиями и т.д.

- **Уровень представления** обеспечивает взаимодействие комплекса с внешним миром: сервисами, администраторами, пользователями сервисов и

внешними информационными системами.

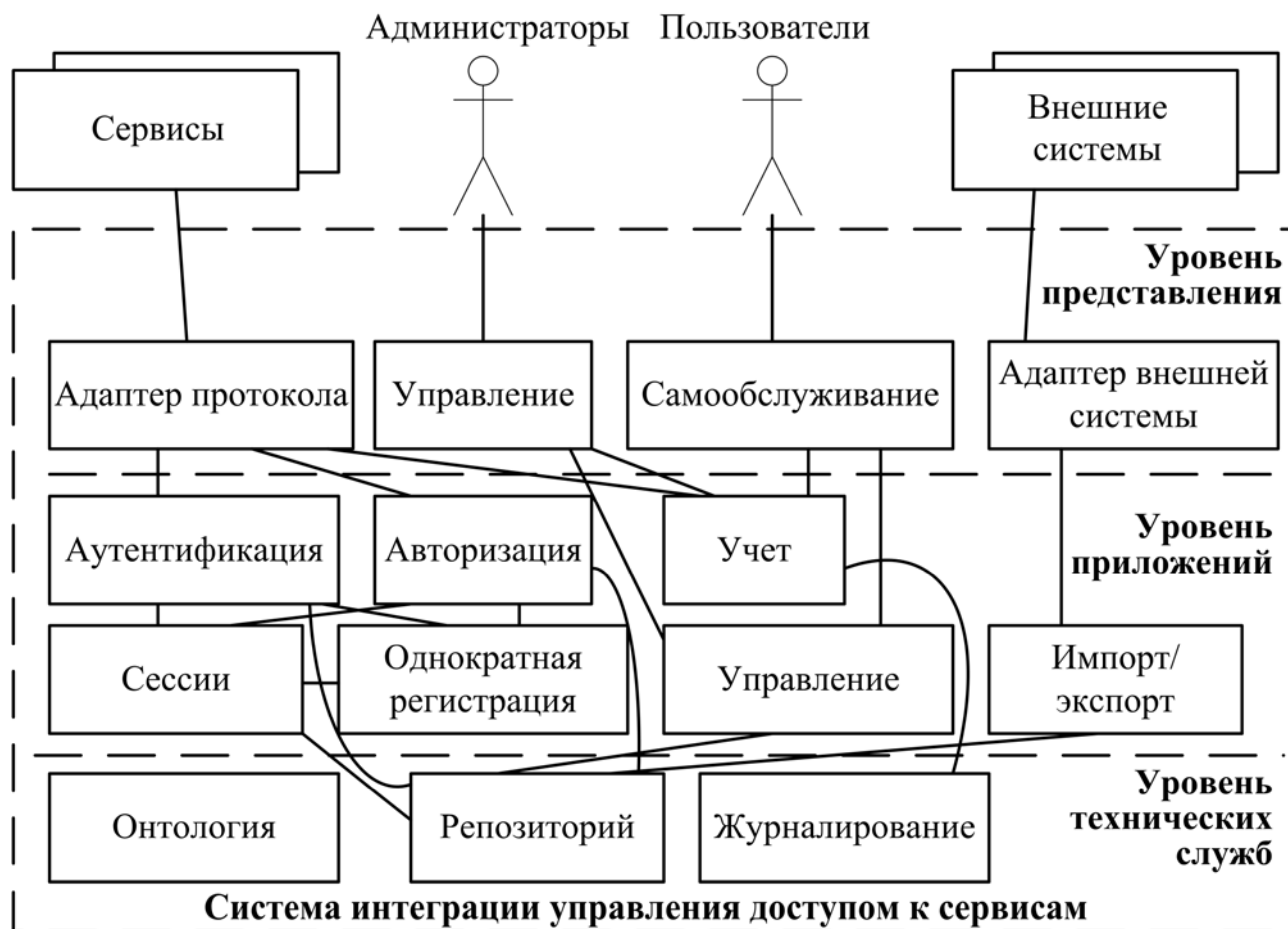


Рис 4.1. Логическая архитектура комплекса программ по интеграции управления доступом к сервисам

4.1.3.1 Уровень технических служб

Уровень технических служб состоит из следующих компонентов:

- **Онтология** содержит машинное представление семантической модели управления доступом к сервисам.
- **Репозиторий** правил разграничения доступа хранит правила разграничения доступа в общем формате и обеспечивает взаимодействие с репозиториями правил разграничения доступа конкретных сервисов.
- Модуль **журналирования** обеспечивает регистрацию всех событий системы управления доступом.

4.1.3.2 Уровень приложений

Уровень приложений состоит из следующих компонентов:

- Модуль **аутентификации** обеспечивает идентификацию и аутентификацию субъектов доступа.
- Модуль **авторизации** обеспечивает проверку прав доступа субъектов доступа к объектам доступа.
- Модуль **учета** ведет учет использования сервисов субъектами доступа.
- Модуль **сессий** отвечает за создание и отслеживание сессий субъектов доступа для реализации однократной регистрации.
- Модуль **однократной регистрации** обеспечивает однократную регистрацию субъектов доступа.
- Модуль **управления** отвечает за управления правилами разграничения доступа в репозитории правил разграничения доступа.
- Модуль **импорта/экспорта** обеспечивает обмен данными с внешними информационными системами.

4.1.3.3 Уровень представления

Уровень представления состоит из следующих компонентов:

- **Адаптер протокола** обеспечивает подключение сервисов к системе управления доступом по разным протоколам и с разной структурной организацией данных. Обеспечивает первый и второй уровень интеграции управления доступом.
- Система **управления** позволяет администраторам управлять доступом к сервисам разных типов через единый интерфейс.
- Система **самообслуживания** позволяет пользователям выполнять некоторые действия по управления доступом самостоятельно (поддержание в актуальном состоянии персональной и контактной информации, смена забытого пароля, простые настройки конфигурации сервисов).
- **Адаптеры внешних систем** обеспечивают подключение различных

информационных систем к системе управления доступом. Обеспечивает нулевой уровень интеграции.

4.1.4 Взаимодействие сервисов с комплексом программ

Диаграмма последовательности управления доступом к сервисам при использовании комплекса программ показана на рис. 4.2 (основной успешный сценарий, второй уровень интеграции).



Рис 4.2. Диаграмма последовательности управления доступом к сервису

Взаимодействующие объекты на диаграмме: субъект доступа, сервис и комплекс программ по управлению доступом к сервисам (система). Субъект выполняет запрос на доступ к сервису. Сервис пересылает полученный запрос к комплексу программ, который выполняет идентификацию и аутентификацию субъекта и создает сессию, в которой запоминаются данные о субъекте. Затем выполняется поиск прав доступа субъекта и их анализ, на основе которого комплекс программ принимает решение о разрешении или запрещении доступа

(авторизация). Комплекс программ сообщает о своем решении сервису, который выполняет указанное действие.

При появлении второго запроса субъекта на доступ к сервису, идентификация и аутентификация не проводятся повторно, вместо этого данные о субъекте извлекаются из созданной ранее сессии. После получения данных о субъекте из сессии авторизация выполняется обычным образом.

4.1.5 Архитектура развертывания

После внедрения комплекс программ по управлению доступом к сервисам становится единственной точкой хранения правил разграничения доступа к сервисам. В случае остановки работы комплекса становится невозможной проверка прав доступа к сервисам, что ведет к отказу в обслуживании пользователей сервисами. В связи с этим необходимо обеспечить надежную работу комплекса.

В целях повышения надежности работы комплекса, разработана архитектура развертывания, отражающая распределение компонентов комплекса программ по управлению доступом по физическим серверам. Архитектура развертывания показана на рис. 4.3.

Причины остановки аппаратных и программных систем исследованы в работах [80, 81, 82, 83, 84, 85, 86]. Для повышения надежности предлагается использовать резервирование [87, 88]. Выбран метод поэлементного резервирования, т.к. он обеспечивает большую надежность, чем при общем резервировании [89]. С точки зрения анализа надежности система состоит из трех элементов [90], соответствующих уровням логической архитектуры: технических служб, приложений и представления). Компоненты, реализующие каждый уровень логической архитектуры, размещаются на отдельных серверах. Каждый сервер продублирован в соответствии с поэлементным подходом к резервированию.

Предлагается использовать постоянное резервирование, при котором серверы работают одновременно в режиме балансировки нагрузки. Хотя надежность при резервировании замещением выше, чем при постоянном резервиро-

вании [87], производительность работы при резервировании замещением ниже, т.к. один сервер простаивает.



Рис 4.3. Архитектура развертывания комплекса программ по управлению доступом к сервисам

4.1.6 Реализация

4.1.6.1 Репозиторий правил разграничения доступа

При реализации комплекса программ в качестве репозитория правил разграничения доступа был выбран каталог LDAP [36]. Каталоги сейчас являются фактическим стандартом хранения правил разграничения доступа. Широкую популярность LDAP обеспечивает наличие стандартных схем данных для описания пользователей [37, 38, 39]. Существующие стандартные схемы могут быть расширены для управления доступом к широкому кругу сервисов.

Для использования каталога LDAP в качестве репозитория правил раз-

граничения доступа, было предложено отображение формальной алгебраической записи правил разграничения доступа в формат LDAP. Подробно отображение описано в Приложении 1.

При реализации комплекса программ в качестве каталога LDAP был выбран Sun Java Directory Server, т.к. это каталог промышленного уровня, распространяемый бесплатно.

Протокол LDAP также выбран в качестве внутреннего протокола идентификации и аутентификации.

4.1.6.2 Адаптеры сервисов

Реализован ряд адаптеров протоколов идентификации:

- Адаптер RADIUS на основе FreeRADIUS;
- Адаптер J2EE, SAML и федеративной идентификации на основе OpenSSO.
- Адаптер для сетей Windows на основе Samba.

4.1.6.3 Консоль управления доступом к сервисам

Консоль управления доступом к сервисам реализована с использованием программного продукта Sun Java System Delegated Administrator [91]. Данный программный продукт позволяет управлять учетными записями в каталоге Sun Java System Directory Server. Delegated Administrator предоставляет два интерфейса: Web и командную строку. Недостатком является небольшое число сервисов, управлять доступом к которым можно с помощью Delegated Administrator – это электронная почта и органайзер. Достоинством является встроенная возможность расширения функциональности Delegated Administrator по управлению доступом к другим сервисам.

Delegated Administrator использует механизм сервисных пакетов – готовых наборов атрибутов управления доступом, которые можно быстро и согласовано назначить субъекту доступа. В стандартной конфигурации существуют пакеты только для электронной почты и для органайзера, но допускается создавать собственные сервисные пакеты. Сервисные пакеты делают возможным

применение Delegated Administrator для управления доступом к широкому кругу сервисов. Для этого нужно создать сервисные пакеты с набором атрибутов, необходимых для каждого типа сервисов. При этом используются атрибуты LDAP, преобразование в формат, понятный сервису, выполняется адаптером протокола управления доступом. Пример сервисного пакета управления доступом к межсетевому экрану Mikrotik [92]:

```
inetCOS: Mikrotik User  
objectClass: mikrotikprofile  
Mikrotik-Rate-Limit: 32k
```

Атрибут «inetCOS» является служебным атрибутом Delegated Administrator, содержащим имя сервисного пакета [91]. Класс «mikrotikprofile» указывает, что объект является пользователем межсетевого экрана Mikrotik. Атрибут «Mikrotik-Rate-Limit» задает полосу пропускания, доступную пользователю.

Delegated Administrator, как следует из названия, включает средства распределения полномочий управления между администраторами путем делегирования. Возможности Delegated Administrator уже, чем в модели интегрированной инфраструктуры управления доступом: строится иерархия администраторов, соответствующая структуре дерева каталога LDAP. На вершине находится администратор верхнего уровня, имеющий все полномочия управления. Затем следуют администраторы организаций, расположенных на первом уровне дерева каталога. Эти администраторы управляют доступом в рамках своей организации и организаций, расположенных ниже ее в дереве каталогов. Возможностей делегирования, предоставляемых Delegated Administrator достаточно для эксплуатации системы в небольших и средних организациях. В дальнейшем планируется реализация полнофункционального делегирования в соответствии с предложенной моделью. Пример окна Web-интерфейса системы управления доступом показан на рис. 4.4.

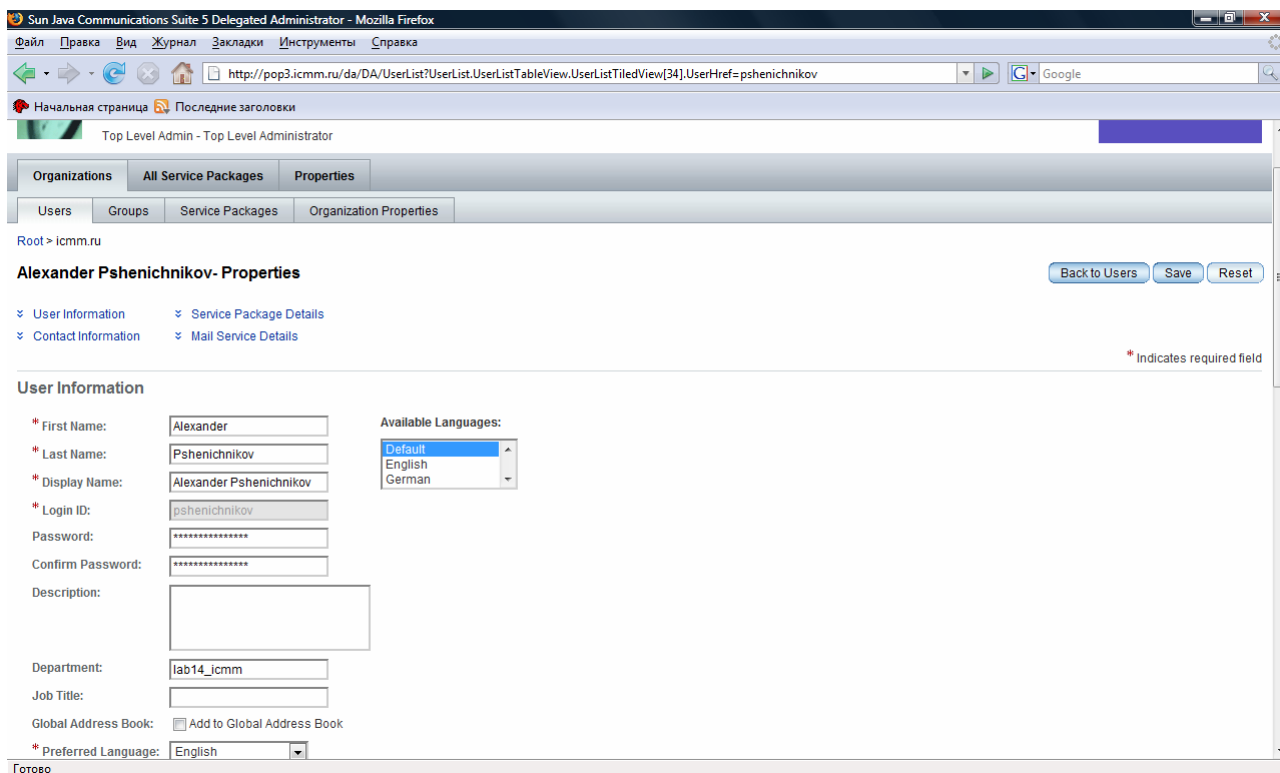


Рис 4.4. Пример окна Web-интерфейса системы управления доступом

Пример интерфейса командной строки показан на рис. 4.5. В примере создается пользователь Ivan Ivanov, затем ему назначаются роль администратора организации icmm.ru

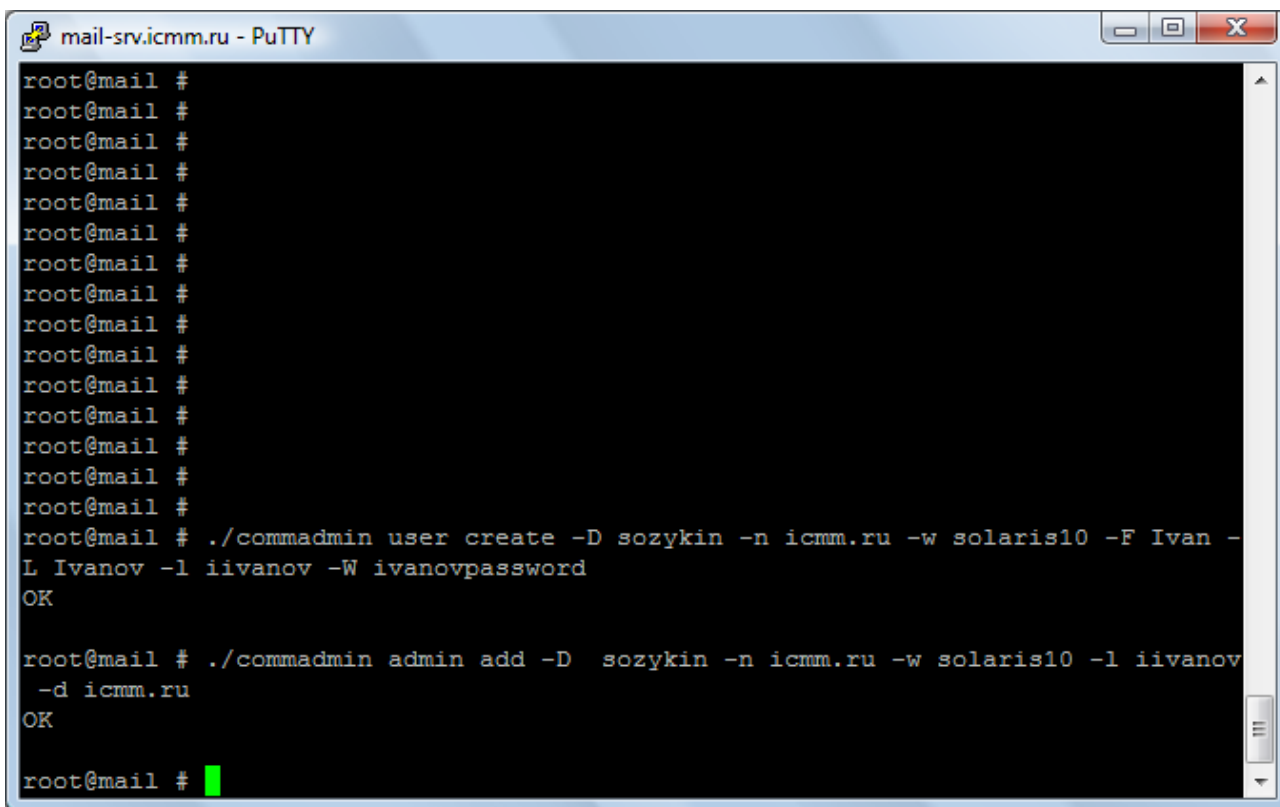


Рис 4.5. Пример интерфейса командной строки системы управления доступом

4.2 Практическое применение

4.2.1 Сеть Пермского научного центра

Комплекс программ применяется для управления доступом к сервисам корпоративной сети Пермского Научного центра Уральского отделения Российской Академии наук (ПНЦ), которая объединяет четыре академических института и Президиум. Схема корпоративной сети ПНЦ показана на рис 4.6. Организации территориально распределены по городу Перми на значительное расстояние. Протяженность оптической инфраструктуры сети ПНЦ превышает 32 км. В таких условиях управления доступом к сервисам сети из всех институтов является сложной задачей. Для ее упрощения в сети ПНЦ был внедрен комплекс программ по управлению доступом к сервисам.



Рис 4.6. Схема корпоративной сети ПНЦ УрО РАН

4.2.2 Сервисы сети Пермского научного центра

Комплекс программ применен для управления доступом к сетевым, информационным и вычислительным сервисам сети ПНЦ, приведенным в табл. 4.1.

Таблица 4.1.

Сервисы сети ПНЦ

№	Сервис	Протокол управления доступом
Сетевые сервисы		
1	Доступ в Интернет	RADIUS
2	Точки беспроводного доступа WiFi	RADIUS
3	Модемные пулы аналоговый и ISDN	RADIUS
4	Электронная почта	LDAP
5	Файловые серверы Windows и UNIX (Solaris)	LDAP
Информационные сервисы		
6	Порталы ИМСС и ПНЦ	J2EE
7	Система управления документами	LDAP
8	Web-сервер (домашние страницы сотрудников)	LDAP
Вычислительные сервисы		
9	Кластер MBC-1000/16	LDAP
10	Серверы Unix (Solaris)	LDAP

4.2.3 Схема реализации системы управления доступом к сервисам

На основе разработанного комплекса программ в сети ПНЦ реализована система управления доступом к сервисам. Схема реализации показана на рис. 4.7.

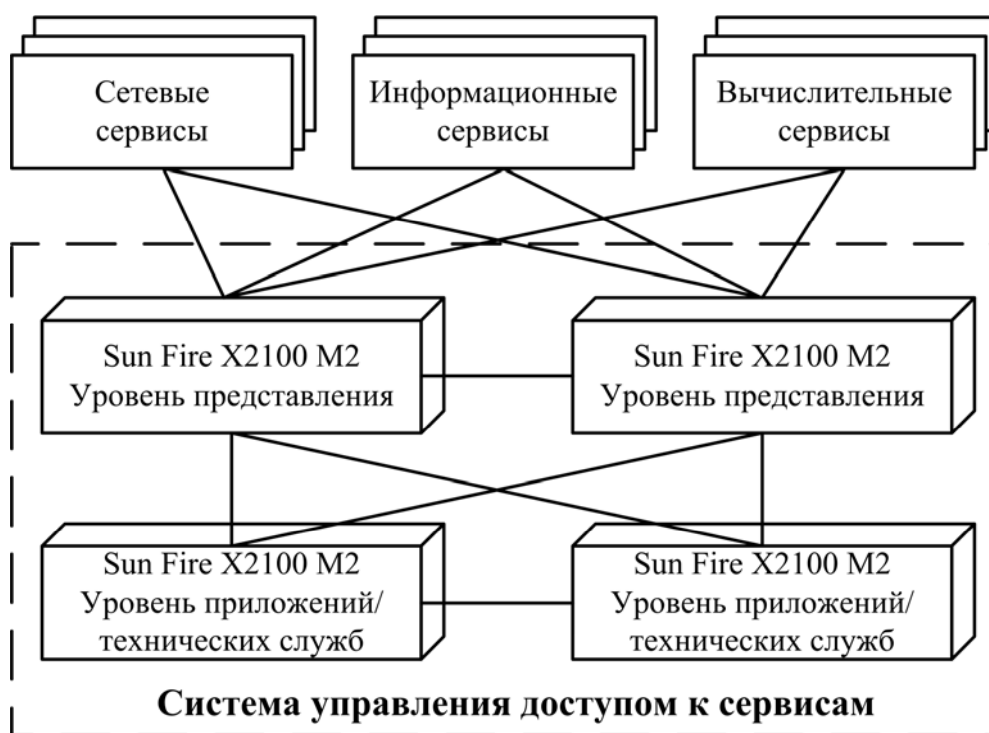


Рис 4.7. Схема реализации системы управления доступом к сервисам в сети ПНЦ

Система развернута на четырех серверах Sun Fire X2100 M2, работающих под управлением ОС Solaris 10. При реализации уровни технических служб и приложений совмещены в одном сервере. Серверы работают в режиме балансировки нагрузки.

4.2.4 Интеграция с системой статистики использования сервисов

Выполнена интеграция системы управления доступом к сервисам с системой статистики использования сервисов. Интеграция позволила строить более информативные отчеты по статистике использования сервисов. До интеграции система статистики работала с идентификаторами пользователей. Часто по идентификатору трудно было определить реального пользователя. Система управления доступом к сервисам кроме идентификаторов содержит подробную информацию о пользователях, включая организационную принадлежность. Интеграция системы статистики с системой управления доступа позволила строить отчеты не по идентификаторам, а по реальным именам пользователей с возможностью группировок по организациям и подразделениям. Пример подобного отчета приведен на рис. 4.8.

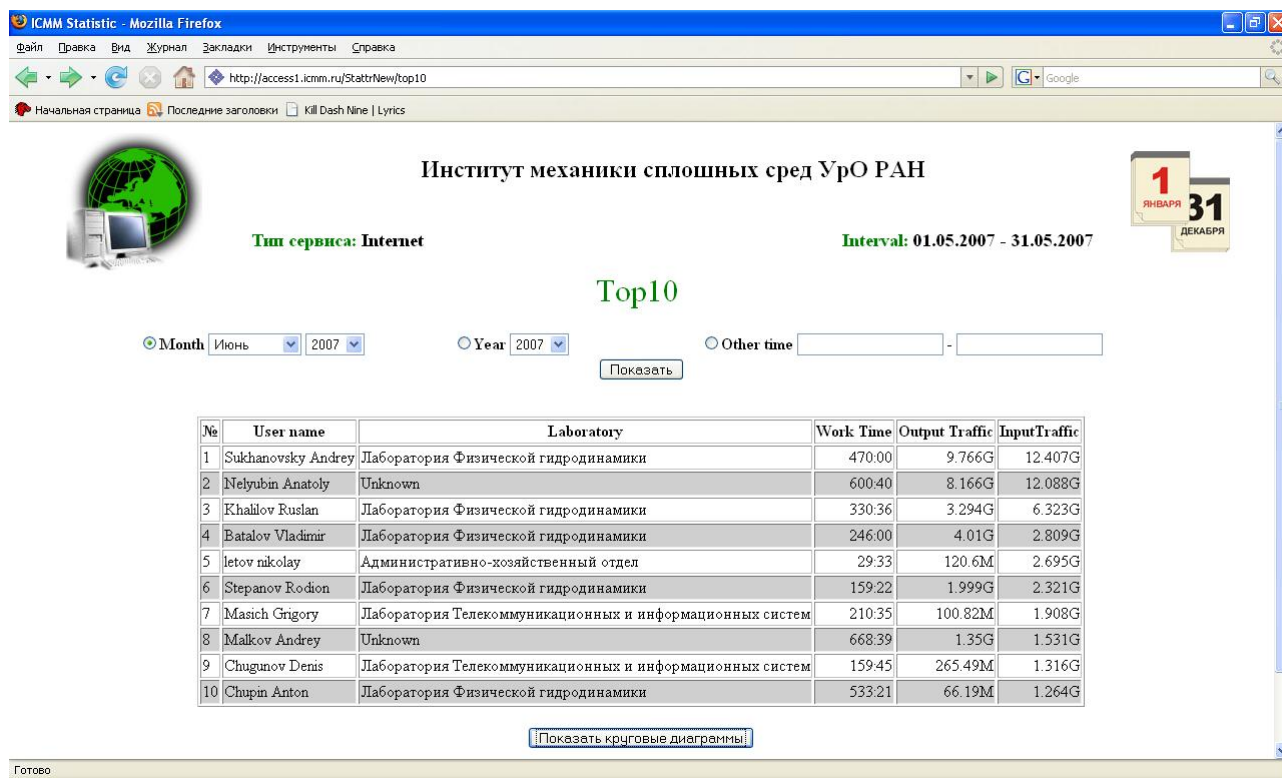


Рис 4.8. Пример отчета системы статистики использования сервисов

Предоставляется статистика об использовании Интернет и модемного пула сотрудниками ПНЦ с группировкой по организациям и подразделениям. Для интеграции реализован адаптер JDBC, позволяющий записывать данные из каталога LDAP в СУБД Oracle, которая используется системой статистики. Контроль прав доступа при обращении к Web-отчетам осуществляется с применением OpenSSO.

4.3 Оценка эффективности использования системы семантической интеграции управления доступом к сервисам

В данном разделе выполнен анализ эффективности использования семантической интеграции управления доступом к сервисам. Для этой цели был проведен ряд экспериментов с системой, внедренной в сети ПНЦ, с целью определить, как внедрение семантической интеграции управления доступом повышает удобство работы с сервисами и снижает затраты на управление.

4.3.1 Удобство работы с сервисами

С точки зрения управления доступом, основной причиной неудобства работы пользователей с сервисами является большое количество учетных записей

[109]. Наблюдается так называемая «усталость от паролей», вызванная большим количеством учетных записей, требующих ввода паролей [110]. Данная проблема актуальна именно для пользователей, которые являются людьми, т.к. запоминание идентификаторов, сложных паролей и их многократный ввод программными и аппаратными системами не представляют проблем [111, 112].

В сети ПНЦ до внедрения системы управления доступом к сервисам каждому пользователю было выделено минимум 10 учетных записей, по количеству сервисов сети (табл. 4.1). Некоторые сотрудники ПНЦ использовали по несколько учетных записей одного сервиса, у них общее количество идентификаторов и паролей превышало 10 штук.

После внедрения системы управления доступом к сервисам количество учетных записей сократилось с 10 до одной. Это позволило устранить проблему «усталости от паролей». Реализованная для некоторых сервисов однократная регистрация позволила сократить количество вводов паролей.

Сокращение количества идентификаторов также привело к уменьшению количества запросов в службу поддержки по восстановлению забытых паролей и консультаций по порядку доступа к сервисам на 60%.

4.3.2 Сокращение затрат на управления доступом

До внедрения системы управления доступом к сервисам, использующей семантическую интеграцию, управления доступом к сервисам сети ПНЦ велось через собственные консоли управления каждого сервиса. Таких консолей было 10 штук, по количеству сервисов (табл. 4.1). После внедрения системы администраторам была предоставлена единая консоль управления доступом к сервисам всех типов в сети ПНЦ.

Проведен сравнительный анализ временных характеристик процесса управления доступом к сервисам с использованием системы интегрированного управления и без нее. Результаты сравнительного анализа представлены в табл. 4.2.

Таблица 4.2.

Результаты сравнительного анализа временных затрат в ходе управления доступом к сервисам

Операция	С помощью традиционных систем управления доступом, мин.	С помощью системы интегрированного управления, мин.
Создание учетной записи	15-30	3-5
Изменение учетной записи	5-10	2-3
Удаление учетной записи	15-30	3-5
Назначение прав доступа	10-15	4-10
Изменение прав доступа	5-10	1-2

Время, сэкономленное при управлении доступом с использованием системы интегрированного управления, может составлять до 80% от обычного времени работы, с применением существующих систем управления доступом.

Интеграция управления доступом к сервисам разных типов позволила снизить требования к администраторам за счет предоставления единой интуитивной понятной консоли управления с Web-интерфейсом.

4.4 Выводы по главе

Предложенные в работе модели, методы и технологии реализованы в виде комплекса программ по управлению доступом к сервисам разных типов.

Комплекс программ применен для создания системы управления доступом к сервисам сети ПНЦ. Внедрение системы семантической интеграции управления доступом позволило повысить эффективность управления доступом к сервисам за счет повышения удобства работы с сервисами, сокращения временных затрат на управления доступом и снижения требований к администраторам.

ЗАКЛЮЧЕНИЕ

В данной диссертационной работе получены следующие научные и практические результаты:

1. Выполнено исследование подходов к управлению доступом к сервисам. Показано, что причиной низкой эффективности процесса управления доступом является многообразие методов и технологий управления доступом. Возможности существующих систем интеграции управления доступом ограничены из-за использования синтаксического или структурного подходов к интеграции. Для расширения возможностей существующих систем предлагается применить семантическую интеграцию.

2. Предложена методика интеграции управления доступом к сервисам разных типов на основе семантического подхода. Методика обеспечивает повышение интероперабельности: значительно расширяет круг поддерживаемых сервисов и методов управления доступом.

3. Предложена онтология управления доступом к сервисам. Онтология задает семантику базовых понятий и операции предметной области управления доступом к сервисам и служит основой для семантической интеграции механизмов управления доступом к сервисам разных типов, использующим различные модели, методы и технологии.

4. Разработана алгебраическая запись правил разграничения доступа к сервисам, позволяющая описывать правила разграничения доступа в формальном виде с использованием различных методов управления доступом.

5. Реализован комплекс программ управления доступом к сервисам на основе разработанных технологий.

6. Комплекс программ успешно применен для интеграции управления доступом к сервисам Пермского научного центра.

7. Исследование эффективности использования системы показало, что семантическая интеграция управления доступом повышает удобство использо-

вания сервисов корпоративной сети, сокращает временные затраты на управление доступом и снижает требования к администраторам.

Основные результаты работы докладывались и обсуждались на следующих научных конференциях и семинарах:

- Всероссийская научная конференция «Научный сервис в сети Интернет», Новороссийск, 2002.
- 34-я Региональная молодежная конференция «Проблемы теоретической и прикладной математики», Екатеринбург, 2003.
- 35-я Региональная молодежная конференция «Проблемы теоретической и прикладной математики», Екатеринбург, 2004.
- XI конференция представителей региональных научно-образовательных сетей «RELARN-2004», Самара, 2004.
- XIII конференция представителей региональных научно-образовательных сетей «RELARN-2006», Барнаул, 2006.
- Всероссийская научная конференция «Научный сервис в сети Интернет: технологии параллельного программирования», Новороссийск, 2006.
- XIV конференция представителей региональных научно-образовательных сетей «RELARN-2007», Нижний Новгород, 2007.
- XIV Всероссийская научно-методическая конференция «Телематика'2007», Санкт-Петербург, 2007.
- Всероссийская научная конференция «Научный сервис в сети Интернет: многоядерный компьютерный мир. 15 лет РФФИ», Новороссийск, 2007.
- International Conference: Computational and Informational Technologies in Science, Engineering and Education (CTMM-2008) Almaty, Kazakhstan, 2008

По теме диссертации опубликовано 17 научных работ.

Публикации в рецензируемых журналах из списка ВАК:

1. *Созыкин А.В.* Модели и методы создания интегрированной инфраструктуры управления доступом к сервисам // Системы управления и информационные технологии, 2007, №4.1(30). - С. 191-195.

2. Созыкин А.В., Масич Г.Ф., Бездушный А.Н., Бобров А.В., Босов А.В., Масич А.Г. Онтология управления доступом к сервисам // Научные технологии, №11, 2008. – С.34-43.

Публикации в других изданиях:

1. Бездушный А.Н., Масич А.Г., Масич Г.Ф., Созыкин А.В., Серебряков В.А. Интеграция сервисов управления объектами сети с информационными ресурсами посредством службы каталогов LDAP // Труды Всероссийской научной конференции “Научный сервис в сети Интернет” (23-28 сентября 2002г., Новороссийск). – М.: Изд-во МГУ, 2002. - С.119-122

2. Алексеев А.Н., Масич А.Г., Масич Г.Ф., Созыкин А.В. Использование метакаталогов для создания справочной системы научного института. // Тез. докл. 13 Зимней школы по механике сплошных сред, Пермь, 2003, С.15

3. Алексеев А.Н., Масич А.Г., Масич Г.Ф., Созыкин А.В. О некоторых аспектах разработки метакаталога для справочной системы научного института // Труды 34-й Региональной молодежной конференции «Проблемы теоретической и прикладной математики», Екатеринбург, 2003.

4. Масич А.Г., Масич Г.Ф., Созыкин А.В. Использование каталога LDAP для управления данными о пользователях сервисами корпоративной сети научного центра РАН // Труды 35-й Региональной молодежной конференции «Проблемы теоретической и прикладной математики», Екатеринбург, УрО РАН, 2004,- с.323-327

5. Масич А.Г., Масич Г.Ф., Созыкин А.В. Аспекты развития и управления в корпоративной сети Пермского научного центра УрО РАН // Тез. докл. XI конференции представителей региональных научно-образовательных сетей «RELARN-2004», Самара, 2004, - с. 51-55

6. Созыкин А.В., Масич Г.Ф., Масич А.Г., Бездушный А.Н. Вопросы интеграции управления идентификацией пользователей сетевых, вычислительных и информационных сервисов. // Журнал «Электронные библиотеки», том 7, выпуск 2. М.: Институт развития информационного общества, 2004.

7. Масич А.Г., Масич Г.Ф., Созыкин А.В. Организация распределенного

каталога корпоративной сети. // Информационные управляющие системы: Сборник научных трудов ПГТУ, - Пермь, 2004. - С. 279-283

8. Масич Г.Ф., Алексеев А.Н., Бобров А.В., Созыкин А.В., Чугунов Д.П. Использование технологии ИСИР при построении корпоративного портала // Информационные и математические технологии в науке, технике и образовании: Труды X Байкальской Всероссийской конференции «Информационные и математические технологии в науке, технике и образовании». Часть I. - Иркутск: ИСЭМ СО РАН, 2005. - С. 12-18.

9. Созыкин А.В., Масич Г.Ф., Масич А.Г., Бобров А.В. Архитектура консолидированного хранилища данных о пользователях и сервисах корпоративной сети. // Материалы XIII конференций представителей региональных научно-образовательных сетей «RELARN-2006». Сборник тезисов докладов – Барнаул: Изд-во АлтГТУ, 2006 – С.69-73.

10. Масич Г.Ф., Созыкин А.В., Бобров А.В. Модель системы управления доступом к сервисам корпоративной сети // Научный сервис в сети Интернет: технологии параллельного программирования: Труды Всероссийской научной конференции – М.: Изд-во МГУ, 2006. – С.221-223.

11. Созыкин А.В., Масич Г.Ф. Использование централизованного управления идентификацией пользователей в Пермском научном центре УрО РАН. // Материалы XIV конференции представителей региональных научно-образовательных сетей «RELARN-2007». Сборник тезисов докладов – Нижний Новгород, 2007 – С.42-48.

12. Масич Г.Ф., Созыкин А.В., Бобров А.В. Использование системы РАМ (Pluggable Authentication Modules) для реализации однократной регистрации пользователей UNIX-серверов. // Труды XIV Всероссийской научно-методической конференции Телематика'2007 – СПб.: Редакционно-издательский отдел СПбГИТМО, 2007. – С.385-387.

13. Созыкин А.В., Масич Г.Ф., Бобров А.В. Интеграция управления идентификацией пользователей научных сервисов // Научный сервис в сети Интернет: многоядерный компьютерный мир:15 лет РФФИ: Труды Всероссийской

научной конференции (24-29 сентября 2007 г., г. Новороссийск) – М.: изд-во МГУ, 2007. - С. 323-328.

14. *Созыкин А.В. Масич Г.Ф. Бобров А.В.* Формальная модель управления доступом к сервисам // Журнал «Информационные технологии моделирования и управления» - Воронеж: изд-во "Научная книга", ISSN 1813-9744, 2007, № 7. – С. 841-849.

15. *Sozykin A.V., Masich G.F.* Integrated access control infrastructure for network of Perm Research Center of the UrB of RAS // International Conference: Computational and Informational Technologies in Science, Engineering and Education (CTMM-2008) Almaty, Kazakhstan, September 10 — September 14, 2008. – http://www.nsc.ru/ws/show_abstract.dhtml?en+186+13669

Личный вклад автора в работах с соавторами заключается в разработке методики семантической интеграции управления доступом к сервисам разных типов, создании онтологии управления доступом к сервисам, разработке алгебраического представления правил разграничения доступа, описания реализации предложенных технологий в виде комплекса программ.

Работа поддержана грантами РФФИ:

– «Корпоративный портал научного института на основе интеграции информационно-справочных сервисов и сервисов сетевого управления» № 03-07-90140-в (2003-2005).

– «Телекоммуникационные ресурсы ПНЦ УрО РАН» № 04-07-96003-Урал (2004-2006).

Работа поддержана программой Президиума РАН «Информатизация»:

– Разработка пилотного проекта системы «Научный институт РАН» в ИМСС УрО РАН (2005).

– Развитие и внедрение системы «Научный институт РАН» в ИМСС и Президиуме ПНЦ УрО РАН (2006).

В заключении автору хотелось бы поблагодарить всех сотрудников Лаборатории телекоммуникационных и информационных систем Института механики сплошных сред УрО РАН, сотрудников Вычислительного Центра

им. А.А. Дородницына РАН: к.ф.-м.н. А.Н. Бездушного, к.т.н. А.К. Нестеренко, к.т.н. Т.М. Сысоева и сотрудников Института математики и механики УрО РАН д.т.н. Ю.И. Кузякина, к.т.н. И.А. Хохлова, к.т.н. М.Л. Гольдштейна. Особо хочу поблагодарить моего научного руководителя к.т.н. Г.Ф. Масича.

ПРИЛОЖЕНИЕ 1. ОТОБРАЖЕНИЕ ПОНЯТИЙ ФОРМАЛЬНОЙ АЛГЕБРАИЧЕСКОЙ ЗАПИСИ ПРАВИЛ РАЗГРАНИЧЕНИЯ ДОС- ТУПА В LDAP

Таблица П1-1.

Отображение понятий формальной алгебраической записи правил разграниче-
ния доступа в LDAP

Понятие формального языка описания правил разграничения доступа	Понятие LDAP
Субъект	Классы объектов person, organizationalPerson, inetOrgPerson
Идентификатор	Атрибут uid
Аутентификатор	Атрибут userPassword
Контейнер	Классы объектов organization (контей- нер-организация), organizationalUnit (контейнер-подразделение), groupOfNames, groupOfUniqueNames (контейнеры-группы).
Роль	Класс объектов organizationalRole
Права доступа	Атрибуты LDAP, специфичные для сервисов разных типов
Атрибуты	Атрибуты sn, gn, cn, email, address, roomNumber, telephoneNumber и дру- гие специфичные для сервисов разных типов
Административные роли	Класс объектов organizationalRole
Административные права доступа	ACL, атрибуты LDAP, специфичные для сервисов разных типов

ПРИЛОЖЕНИЕ 2. АКТЫ ВНЕДРЕНИЯ

УТВЕРЖДАЮ

Председатель ПНЦ УрО РАН

Академик

 Матвеев В.П.

« 24 » 2008 г.



АКТ

внедрения результатов диссертационной работы

Созыкина Андрея Владимировича

«Семантическая интеграция управления доступом к сервисам»,

представленной на соискание ученой степени кандидата технических наук
по специальности 05.13.11 – Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

Комиссия при Президиуме ПНЦ УрО РАН в составе:

председатель: Роговой А.А., зам. директора по науке ИМСС УрО РАН,


члены: Шмагель К.В., зам. директора по науке ИЭГМ УрО РАН;

Приходченко В.П., гл. уч. секретарь ПНЦ УрО РАН,


составили настоящий акт о том, что результаты диссертационной работы Созыкина А.В. «Семантическая интеграция управления доступом к сервисам» использованы в Институте механики сплошных сред УрО РАН, Институте экологии и генетике микроорганизмов УрО РАН, Горном институте УрО РАН, Институте технической химии УрО РАН и Президиуме Пермского научного центра УрО РАН.

С помощью предложенной в работе методики и на основе разработанного комплекса программ, диссертантом создана система управления доступом к сервисам корпоративной сети ПНЦ УрО РАН. На систему возлагается управление доступом сотрудников вышеперечисленных научных институтов и Президиума ПНЦ УрО РАН к информационным, сетевым и вычислительным сервисам сети ПНЦ УрО РАН.

Использование указанных результатов позволило сделать более удобной работу сотрудников институтов с сервисами сети ПНЦ УрО РАН, сократить временные затраты на управления доступом, снизить требования к администраторам.

Председатель комиссии:  Роговой А.А.

Члены комиссии:  Шмагель К.В.

 Приходченко В.П.



Директор ИМСС УрО РАН

Академик РАН

Матвеев В.П.

« 19 » 11 2008 г.

АКТ

внедрения результатов диссертационной работы

Созыкина Андрея Владимировича

«Семантическая интеграция управления доступом к сервисам»,
представленной на соискание ученой степени кандидата технических наук
по специальности 05.13.11 – математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

Комиссия при ИМСС УрО РАН в составе:

председатель: Роговой А.А., зам. директора по науке

члены: Шайдуров В.Г., зам. директора по вычислительной технике
Чугунов Д.П., инженер-исследователь,

составили настоящий акт о том, что результаты диссертационной работы
Созыкина А.В. «Семантическая интеграция управления доступом к сервисам»
использовались:

– При выполнении НИР по темам: № ГР 01.200.1 18927 "Развитие
вычислительных, информационных и телекоммуникационных ресурсов Пермского
фрагмента РИВС УрО РАН", (2002 – 2004); № ГР 01.20.0 500084 "ИВТР
корпоративной сети Пермского научного центра УрО РАН", (2005-2007).

– В рамках проектов, поддержанных грантами РФФИ № 03-07-90140-в
(2003-2005) и № 04-07-96003-Урал (2004-2006).

Председатель комиссии: _____ Роговой А.А.

Члены комиссии: _____ Шайдуров В.Г.

_____ Чугунов Д.П.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. De Capitani di Vimercati S., Paraboschi S, Samarati P. Access control: principles and solutions. Software Practice and Experience, № 33, P. 397-421, 2003.
2. Lampson B., Abadi M., Burrows M., Wobber E. Authentication in distributed systems: Theory and practice // ACM Transactions on Computer Systems, № 10(4), P. 265-310, 1992.
3. Lampson B. Computer security in the real world. Annual Computer Security Applications Conference, 2000.
4. Abadi M., Burrows M., Lampson B., Plotkin G. A calculus for access control in distributed systems. // ACM Transactions on Programming Languages and Systems (TOPLAS), volume 15, P. 706-734. ACM Press, Sep 1993.
5. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. / Под ред. Шаньгина В.Ф. – 2-е изд., перераб. и доп. – М.: Радио и связь, 2001. 376 с.
6. Белкин П.Ю., Михайлский О.О., Першаков А.С. Программно-аппаратные средства обеспечения информационной безопасности. Защита программ и данных: Учеб. пос. для вузов. – М.: Радио и связь. – 1999. – 168 с.
7. Rivest R. The MD5 Message-Digest Algorithm. // Network Working Group. - RFC 1321.
8. U.S. Department of Commerce. National Bureau of Standards. Secure Hash Standard. 1995.
9. Мафтик С. Механизмы защиты в сетях ЭВМ: Пер. с англ. – М.: Мир, 1993. – 216 с.
10. Schneier B. Applied Cryptography. – John Wiley & Sons, Inc., 1996. – 758 p.
11. Feige U., Fiat A., Shamir A. Zero Knowledge Proofs of Identity. // Journal of Cryptology v.1, n.2, 1988, P. 77-94.

12. Needham R.M., Schroeder M.D. Using Encryption for Authentication in Large Networks of Computers. // Communication of the ACM, V.21, N12, December 1978.
13. Neuman B.C., Ts'o T. Kerberos: An Authentication Service for Computer Networks // IEEE Communications, 32(9):33-38. September 1994.
14. National Institute of Standards and Technology. Public Key Infrastructure Technology. ITL Bulletin, July 1997.
<http://www.nist.gov/itl/lab/bulletns/archives/july97bull.htm>
15. Adams C., Farrell S. "Internet X.509 Public Key Infrastructure: Certificate Management Protocols" // RFC 2510, 1999
16. Housley R., Ford W., Polk W., Solo D., "Internet X.509 Public Key Infrastructure: Certificate and CRL Profile" // RFC 3280, 2002.
17. Trado J., Alagappan K. SPX: Global authentication using public key certificates. // IEEE Symposium on Security and Privacy (Oakland, Calif., May 1991), P. 232-244.
18. Guillou L.C., Quisquater J.J. A Practical Zero-Knowledge Protocol Fitted to Security Microprocessor Minimizing Both Transmission and Memory. // Advanced in Cryptology EUROCRYPT'88 Proceedings, Springer-Verlag, 1988, P. 123-128.
19. Department of Defense. Trusted Computer Security Evaluation Criteria, DOD 5200.28-STD., 1985.
20. Lampson B. Protection. // ACM Operating Systems Rev. 8, 1 (Jan. 1974), P. 18-24.
21. Levy, Henry M., Capability-Based Computer Systems. // Digital Equipment Corporation, 1984. ISBN 0-932376-22-3.
22. Bell D., La Padula L. Secure computer systems: unified exposition and Multics interpretation. // Technical Report MTR-2997, MITRE Corporation, 1975.
23. Sandhu R. S., Coynek E.J., Feinsteink H.L., Youmank C.E. Role-Based Access Control Models. IEEE Computer, Volume 29, Number 2, February 1996, P. 38-47.

24. Yuan E., Tong J. Attribute Based Access Control (ABAC) for Web Services. // 3rd International Conference on Web Services (ICWS 2005), Orlando, USA, July 2005, P. 561-569.
25. Obrst L. Ontologies for semantically interoperable systems. // Conference on Information and Knowledge Management. New Orleans, LA, USA, 2003. P. 366 – 369. ISBN:1-58113-723-0.
26. OWL Web Ontology Language Overview. // <http://www.w3.org/TR/owl-features>.
27. The Protege Project. // <http://protege.stanford.edu>
28. Ontolingua. // <http://www.ksl.stanford.edu/software/ontolingua>
29. Chimaera // <http://ksl.stanford.edu/software/chimaera>
30. Laat C., Gross G., Gommans L., Vollbrecht J., Spence D. Generic AAA Architecture. // RFC 2903, 2000.
31. Rigney C., Willens S., Rubens A., Simpson W. Remote Authentication Dial In User Service (RADIUS). // RFC 2865, 2000.
32. Finseth C. An Access Control Protocol, Sometimes Called TACACS. // RFC 1492, 1993.
33. Calhoun P., Loughney J., Guttman E., Zorn G., Arkko J. Diameter Base Protocol. // RFC 3588, 2003.
34. Eduroam project // <http://www.eduroam.org/>
35. Solaris System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP). // Part No: 816-4556-10. <http://docs.sun.com/app/docs/doc/816-4556>
36. Wahl M., Howes T., Kille S. Lightweight Directory Access Protocol (v3). // RFC 2251, 1997.
37. Wahl M. A Summary of the X.500(96) User Schema for use with LDAPv3. // RFC 2256, 1997.
38. Zeilenga K., COSINE LDAP/X.500 Schema. // RFC 4524, 2006.
39. Howard L. An Approach for Using LDAP as a Network Information Service. // RFC 2307, 1998.

40. Kohl J., Neuman C. The Kerberos Network Authentication Service (V5). // RFC 1510, 1993.
41. The A-Select Authentication System. // <http://a-select.surfnet.nl/>
42. Демченко Ю.В. Федеративный доступ к ресурсам научных и университетских сетей. // Тез. докл. конференции представителей региональных научно-образовательных сетей «RELARN-2007», 2007.
43. Security Assertion Markup Language (SAML) // <http://www.oasis-open.org/committees/security/>
44. OpenSAML Project // <http://www.opensaml.org>
45. Windows Live ID // <https://accountservices.passport.net/ppnetworkhome.srf>
46. Маквитти Л. Федеративное управление идентификацией пользователей. // Сети и системы связи, № 13, 2003.
47. Liberty Alliance Project // <http://www.projectliberty.org>
48. WS-Security // <http://www.oasis-open.org/committees/wss/>
49. Valkenburg P., Stals B, Vooren T. Federated Identity Management in Higher Education. // http://aaa.surfnet.nl/info/en/artikel_content.jsp?objectnumber=182026
50. Shibboleth Project // <http://shibboleth.internet2.edu/>
51. Tivoli Identity Manager // <http://www-306.ibm.com/software/tivoli/products/identity-mgr/>
52. Sun Identity Manager // http://www.sun.com/software/products/identity_mgr/index.xml
53. Oracle Identity Manager. // http://www.oracle.com/technology/products/id_mgmt/index.html
54. Microsoft Identity Intergation Server // [http://technet.microsoft.com/ru-ru/miis/default\(en-us\).aspx](http://technet.microsoft.com/ru-ru/miis/default(en-us).aspx)
55. Novell Identity Manager // <http://www.novell.com/products/identitymanager>
56. Perkins E., Witty R.J. Magic Quadrant for User Provisioning, 2H07 Gartner, 23 August 2007, Note G00150475.

57. The Open Web SSO project // <https://opensso.dev.java.net/>
58. Gray N. ACLs in OWL: practical reasoning about access. // 3rd European Semantic Web Conference, Budva, Montenegro, 11th–14th June, 2006.
59. Agarwal S., Sprick B., Wortmann S. Credential Based Access Control for Semantic Web Services. // AAAI Spring Symposium, 2004.
60. Patterson R.S., Miller J.A., Cardoso J., Davis M. Security and Authorization Issues in HL7 Electronic Health Records: A Semantic Web Services Based Approach. // http://lsdis.cs.uga.edu/~rsp/patterson_richard_s_200612.pdf
61. Priebe T., Dobmeier W., Schlager C., Kamprath N. Supporting Attribute-based Access Control in Authorization and Authentication Infrastructures with Ontologies. // Journal of software, Vol. 2, No. 1, 2007.
62. Li H., Zhang X., Wu H., Qu Y. Design and Application of Rule Based Access Control Policies. // ISWC Workshop on Semantic Web and Policy, 2005, P. 34-41.
63. Toninelli A., Kagal L., Bradshaw J.M., Montanari R. Rule-based and Ontology-based Policies: Toward a Hybrid Approach to Control Agents in Pervasive Environments. // Proc. of the Semantic Web and Policy Workshop (SWPW), in conj. with ISWC 2005, Galway, Ireland, Nov. 7, 2005.
64. Toninelli A., Montanari B., Kagal L., Lassila O. A Semantic Context-Aware Access Control Framework for Secure Collaborations in Pervasive Computing Environments. // International Semantic Web Conference, 2006, P. 473-486.
65. Patterson R.S., Miller J.A. Expressing Authorization in Semantic Web Services. // IEEE International Conference Granular Computing, 2006, P. 792- 795.
66. Tomasek M., Furdik K. Service-based architecture of Access-eGov system. // 1st Workshop on Intelligent and Knowledge oriented Technologies, 2006, P. 29-32.
67. Su L., Chadwick D.W., Basden A., Cunningham J.A. Automated Decomposition of Access Control Policies. // Sixth IEEE International Workshop on Policies for Distributed Systems and Networks, 2005, P. 3-13.
68. Colomb R.M. Use of Upper Ontologies for Interoperation of Information Systems. // Technical Report 20/02 ISIB-CNR. Padova, Italy, November, 2002.

69. Cyc Knowledge Server // <http://www.opencyc.org/>
70. Niles I., Pease A. Towards a standard upper ontology. // Conf. Formal Ontology In Information Systems, Ogunquit, Maine, October 17–19, 2001, P. 2–9.
71. Degen W., Heller B., Herre H., Smith B. GOL – a general ontological language. // Int. Conf. Formal Ontology In Information Systems Ogunquit, Maine, October 17–19, 2001, P. 34–46.
72. Gangemi A., Guarino N., Masolo C., Oltramari A., Schneider L. Sweetening ontologies with DOLCE. // 13th Int. Conf. Knowledge Engineering and Knowledge Management. Ontologies and the SemanticWeb, Siguenza, Spain, October 1–4, 2002, P. 166–181.
73. Weber R. Ontological Foundations of Information Systems // Monograph №4. /Australia: Melbourne, Vic., Coopers & Lybrand and the Accounting Association of Australia and New Zealand. – 1997.
74. Wand Y., Weber R. An ontological model of an information system // IEEE Transactions on Software Engineering Journal. – 1990. – 16(11). – P. 1281–1291.
75. ISO/IEC 2382-8:1998 Security.
76. ISO/IEC 27000 Information technology - Security techniques - Information security management systems - Overview and vocabulary.
77. ГОСТ Р 50739-95. Средства вычислительной техники. Защита от несанкционированного доступа к информации.
78. Гостехкомиссия России. Руководящий документ: Защита от несанкционированного доступа к информации. Термины и определения. – М.: ГТК РФ, 1992. – 13 с.
79. Горбатов В.А., Горбатов А.В., Горбатова М.В. Дискретная математика. – М.: АСТ: Астрель, 2006. – 447 с.
80. Gray J.. Why do computers stop and what can be done about it. // Proc. of the 5th Symp. on Reliability in Distributed Software and Database Systems, 1986.
81. Nurmi D., Brevik J., and Wolski R. Modeling machine availability in enterprise and wide-area distributed computing environments. // Euro-Par'05, 2005.

82. Sahoo R. K., Sivasubramaniam A., Squillante M. S., Zhang Y. Failure data analysis of a large-scale heterogeneous server environment. // Proc. of DSN'04, 2004.
83. Oppenheimer D. L., Ganapathi A., Patterson D. A. Why do internet services fail, and what can be done about it? // USENIX Symp. on Internet Technologies and Systems, 2003.
84. Heath T., Martin R. P., Nguyen T. D. Improving cluster availability using workstation validation. // Proc. of ACM SIGMETRICS, 2002.
85. Schroeder B., Gibson G.A. A large-scale study of failures in high-performance computing systems. // International Conference on Dependable Systems and Networks. Philadelphia, PA, USA, June 25-28, 2006.
86. Lin T.Y., Siewiorek D.P. Error Log Analysis: Statistical Modeling and Heuristic Trend Analysis. // IEEE Transactions on reliability, VOL. 39, NO. 4, 1990. P. 419-432.
87. Половко А.М., Гуров С.В. Основы теории надежности. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2006. – 704 с.
88. Половко А.М. Принципы построения абсолютно надежных технических устройств. О-во «Знание», РСФСР. – Л., 1993.
89. Смолицкий Х.Л., Чукреев П.А. О сравнении надежности систем при поэлементном и общем резервировании. – Изв.АН СССР, ОТН, Энергетика и автоматика, № 3, 1959.
90. Б. Гнеденко, Ю. Беляев, А. Соловьев «Математические методы в теории надежности». М.: Наука, 1965, 524 с.
91. Sun Java System Delegated Administrator Administration Guide. // Sun Microsystems, Part No: 819–4438–10, March 2007.
92. Mikrotik Router OS // <http://www.mikrotik.com>.
93. Miller M.S., Yee K.-P., Shapiro J. Capability Myths Demolished. Technical Report SRL2003-02, Systems Research Laboratory, Johns Hopkins University. 2003.

94. Gruber, T.R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. Technical Report KSL 93-04 Knowledge Systems Laboratory Stanford University. 1993.
95. Damiani E., De Capitani di Vimercati S., Fugazza C., Samarati P. Semantics-aware Privacy and Access Control: Motivation and Preliminary Results.
96. Ларман К. Применение UML и шаблонов проектирования. 2-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 624 с.
97. Jordan C.S. Guide to Understanding Discretionary Access Control in Trusted Systems. DIANE Publishing. 1987. ISBN 0788122347.
98. Messaoud B. Access Control Systems / Security, Identity Management and Trust Models. Springer. 2006. 261 p. ISBN: 0387004459.
99. Zhang N., Ryan M., Guelev D.P. Synthesising verified access control systems through model checking. // Journal of Computer Security, Vol. 16, No 1 / 2008. P. 1-61.
100. Zhang N., Ryan M., Guelev D.P. Synthesising verified access control systems in XACML. // 2004 ACM Workshop on Formal Methods in Security Engineering, Washington DC, USA, Oct 2004, P. 56-65.
101. Zhang N., Ryan M., Guelev D.P. Evaluating access control policies through model-checking. // 8th Information Security Conference, Singapore, Sep 2005.
102. Gong L., Needham R., Yahalom R. Reasoning about Belief in Cryptographic Procols. // IEEE 1990 Symposium on Security and Privacy, Oakland, California, May 1990, P. 234-248.
103. Fisler K., Krishnamurthi Sh., Meyerovich L.A., Tschantz M.C. Verification and change-impact analysis of access-control policies. // ICSE'05, St. Louis, Missouri, USA, May 2005.
104. Guelev D.P., Ryan M., Schobbens P-Y. Model-checking access control policies. // 7th Information Security Conference (ISC'04), Springer-Verlag, 2004.
105. OASIS Standard, 1 Feb 2005. eXtensible Access Control Markup Language (XACML) Version 2.0.

106. Adams A., Sasse M.A., Users Are Not the Enemy: Why Users Compromise Security Mechanisms and How to Take Remedial Measures. // Comm. ACM, vol. 42, no. 12, 1999.
107. Dhamija R., Tygar J.D., Hearst M. Why Phishing Works. // Human Factors in Computing Systems (CHI 06), ACM Press, 2006.
108. Norman D.A. Design Rules Based on Analyses of Human Error. // Comm. ACM, vol. 26, no. 4, 1983.
109. Gross B.M., Churchill E.F. Addressing Constraints: Multiple Usernames Task Spillage and Notions of Identity. // Human Factors in Computing Systems: Extended Abstracts (CHI 07), ACM Press, 2007.
110. Р. Дхамиджа, Л. Дюссо. Семь проблем управления идентификацией // «Открытые системы», №5, 2008. <http://www.osp.ru/os/2008/05/5198740/>
111. Florencio D., Herley C. A Large Scale Study of Web Password Habits. // Int'l Word Wide Web Conf. (WWW 07), ACM Press, 2007
112. Dhamija R., Perrig A., Deja Vu: A User Study Using Images for Authentication. // 9th Usenix Security Symp., Usenix Assoc., 2000.

СПИСОК ИЛЛЮСТРАЦИЙ

Рис 1.1. Модель управления доступом Лампсона [2]	10
Рис 1.2. Основные компоненты инфраструктуры открытого ключа	18
Рис 1.3. Списки контроля доступа и возможностей	23
Рис 1.4. Мандатное управление доступом	24
Рис 1.5. Модель ролевого управления доступом [23]	26
Рис 1.6. Типы интеграции и уровни интероперабельности [25]	29
Рис 1.7. Пример семантической интеграции на основе онтологии.....	30
Рис 2.1. Схема прототипа	44
Рис 2.2. Схема прототипа и предлагаемого решения (новые блоки закрашены, развитые отмечены уголком)	47
Рис 2.3. Структурная схема системы семантической интеграции управления доступом к сервисам	49
Рис 3.1. Структура систем управления доступом к сервисам. Объекты.	51
Рис 3.2. Структура систем управления доступом к сервисам. Свойства.	52
Рис 3.3. Состояния управления доступом к сервисам	54
Рис 3.4. События управления доступом к сервисам	54
Рис 3.5. Операции управления доступом к сервисам	55
Рис 3.6. Диаграмма состояний системы управления доступом к сервисам	57
Рис 3.7. Базовый уровень алгебраической записи правил разграничения доступа	59
Рис 3.8. Пример иерархии прав доступа	60
Рис 3.9. Алгебраическая запись правил разграничения доступа. Уровень 1.....	63
Рис 3.10. Алгебраическая запись правил разграничения доступа. Уровень 2....	65
Рис 3.11. Алгебраическая запись правил разграничения доступа. Уровень 3....	68
Рис 4.1. Логическая архитектура комплекса программ по интеграции управления доступом к сервисам	73
Рис 4.2. Диаграмма последовательности управления доступом к сервису	75

Рис 4.3. Архитектура развертывания комплекса программ по управлению доступом к сервисам	77
Рис 4.4. Пример окна Web-интерфейса системы управления доступом	80
Рис 4.5. Пример интерфейса командной строки системы управления доступом	80
Рис 4.6. Схема корпоративной сети ПНЦ УрО РАН	81
Рис 4.7. Схема реализации системы управления доступом к сервисам в сети ПНЦ	83
Рис 4.8. Пример отчета системы статистики использования сервисов	84

СПИСОК ТАБЛИЦ

Таблица 1.1. Пример матрицы доступа.....	22
Таблица 1.2. Оценка существующих систем интеграции управления доступом к сервисам	36
Таблица 1.3. Оценка онтологий в предметной области управления доступом .	38
Таблица 1.4. Основные понятия онтологии BWW	39
Таблица 1.5. Термины и определения Руководящего документа Гостехкомиссии России [78]	40
Таблица 4.1. Сервисы сети ПНЦ	82
Таблица 4.2. Результаты сравнительного анализа временных затрат в ходе управления доступом к сервисам	86
Таблица П1-1. Отображение понятий формальной алгебраической записи пра- вил разграничения доступа в LDAP.....	93