1 Название проекта/ Номер годового отчета

Проект 1994Р: Формальные методы защиты информации в компьютерных сетях

Задача 2: Разработка математических основ, архитектуры и принципов реализации компонент многоагентной системы обучения обнаружению атак на компьютерные сети.

Отчет №2

2. Головной институт

Санкт-Петербургский институт информатики и автоматизации Российской академии наук

3. Институты-участники

Нет

4. Руководитель , номер телефона, факса, адрес электронной почты

Котенко Игорь Витальевич, (812)-323-3570, (812)-328-0685, ivkote@iias.spb.su

5. Дата начала осуществления, продолжительность проекта

1 декабря 2000, 36 месяцев

6. Краткое описание плана работ: цель, предполагаемые результаты, научно-технический подход

Краткий план работ

В-1. Разработка онтологии задач обучения, распределение задач обучения между типовыми агентами обучения	1-3 кварталы
Промежуточный отчет # 1, представляющий результаты исследований по задаче В-1	3 квартал
В-2. Разработка архитектуры многоагентной обучающей системы и математических методов, реализующих функциональности типовых обучающих агентов	4-6 кварталы
Представление статьи в международный журнал	5 квартал
Промежуточный отчет # 2, представляющий результаты исследований по задаче В-2	6 квартал
В-3. Разработка протокола взаимодействия интеллектуальных обучающих агентов (протокола переговоров) для обобщения решений отдельных агентов в соответствии с процедурой мета-классификации и разработка архитектуры многоагентной обучающей системы в целом	7-8 кварталы
В-4. Разработка объектно-ориентированного проекта многоагентной системы обучения обнаружению атак	6-8 кварталы
В-5. Разработка программного прототипа многоагентной системы обучения обнаружению атак, реализующей основные теоретические решения	9-11 кварталы
Промежуточный отчет #3, описывающий результаты решения задачи В-4 и частично разработанные программные компоненты многоагентной системы обучения обнаружению атак	10 квартал
В-6. Оценка свойств, достоинств и недостатков разработанной архитектуры и математических методов, реализованных в компонентах прототипа многоагентной системы обучения обнаружению вторжений в компьютерную сеть	12 квартал
Итоговый отчет, описывающий результаты моделирования многоагентной системы обучения обнаружению вторжений и итоговое заключение по задаче 2 в целом	12 квартал

Примечание: Строки таблицы, показанные серым цветом, отвечают исследованиям, запланированным на второй год работы.

Цель проекта

Целями исследований по задаче 2 проекта являются разработка математических основ, многоагентной архитектуры и принципов реализации системы обучения обнаружению атак, функционирующей параллельно с системой защиты компьютерной сети.

Ожидаемые результаты

1. Онтология задач обучения обнаружению вторжений;
2. Распределение задач обучения между типовыми агентами обучения и архитектура их взаимодействия в рамках многоагентной системы обучения;
3. Математические методы и алгоритмы реализации функций типовых агентов обучения различных классов, а также других компонент многоагентной системы обучения, обеспечивающих взаимодействие агентов. Программная реализация компонент многоагентной системы обучения с использованием современных стандартных сред программирования *Visual C++, JAVA 2, SQL Server, XML* и др.
4. Результаты исследований программных компонент многоагентной системы обучения обнаружению атак на компьютерные сети с оценкой преимуществ и недостатков разработанной архитектуры, а также математических методов, реализованных в компонентах программной системы.

Научно-технический подход

Ключевым аспектом этой задачи является выбор адекватных методов обучения среди существующих, и разработка специализированных методов и алгоритмов, которые могли бы обеспечить обучение на основе прецедентов. Прецеденты, специфицирующие вторжения, являются, как правило, упорядоченными последовательностями данных регистрации различной длины, задаваемыми в терминах, возможно, повторяющихся символов. Эти символы соответствуют предобработанным сообщениям входного трафика, поступающего на порт хостов компьютерной сети. В случае распределенной атаки, так же как и в случае нормальных распределенных действий пользователей, ситуацию на сети задает множество таких последовательностей. По этой причине задача обнаружения знаний в данных для обнаружения вторжений является более сложной и менее изученной по сравнению с традиционными задачами обучения.

Методы обучения включают в себя три класса методов. Первый класс методов строится на основе модели атаки а терминах формального контекстно-свободного языка. В этом случае задача обучения может быть сведена к задаче восстановления грамматики на основе прецедентов. Второй класс методов базируется на статистических свойствах прецедентов, определяющих нормальные и аномальные действия пользователей, осуществляющих доступ к ресурсам сети. Третий класс методов ориентирован на решение задач извлечения правил из прецедентов, определенных в терминах высокоуровневых понятий, например, паттернов.

В основу архитектурных решений положена технология многоагентных систем. Обоснование и разработка конкретной архитектуры будет выполняться на основе декомпозиции общей задачи обучения на множество подзадач в соответствии с онтологией атак, и распределением этих подзадач среди типовых программных агентов обучения, каждый из которых будет использоваться для клонирования ряда специализированных агентов. Каждый специализированный агент при этом настроен на обнаружение частного класса зависимостей (паттернов, логических правила, определенных над паттернами и др.) из данных фиксированного формата (последовательности событий, множества паттернов, подмножества правил и др.).

В математическом описании процедур взаимодействия агентов обучения обнаружению атак, в особенности, распределенных атак, будет использоваться идея мета-классификации, реализуемой на основе многоуровневого обучения, которая предлагает перспективный подход к объединению знаний, полученных из различных источников.

7. Ход выполнения технических работ за первый год (для годовых отчетов за второй год)

Ход выполнения работ за первый год полностью соответствовал плану работ как по содержанию, так и по срокам завершения предусмотренных этапов работ.

Основные достижения за первый год

Основные достижения за первый год были связаны с решением запланированных задач. Эти задачи и полученные по ним результаты перечисляются ниже.

На первый год исследований была запланирована задача В-1 “Разработка онтологии задач обучения, распределение задач обучения между типовыми агентами обучения”. Она включала в себя решение следующих частных подзадач:

1. Анализ структуры данных регистрации.
2. Разработка многоуровневой онтологии задач обучения.
3. Разработка концептуальных моделей типовых агентов многоагентной обучающей системы.

Кроме того, план работ предполагал также проведение частичных исследований по задаче В-2 "Разработка архитектуры многоагентной обучающей системы и математических методов, реализующих функциональности типовых обучающих агентов", полное решение которой было запланировано к концу 6 квартала.

Основные результаты, полученные в рамках вышеназванных задач в течение первого года исследований, таковы.

1.Анализ структуры и особенностей данных, используемых для обучения

Как правило, информация, получаемая из одного источника, не содержит достаточно свидетельств, позволяющих уверенно и своевременно обнаруживать атаки и факты нарушения политики безопасности. Для построения эффективной системы обнаружения вторжений и обучающей системы необходимо использовать взаимосвязанный комплекс данных регистрации, полученных от разнообразных источников и представляющих данные на различных уровнях обобщения (на сетевом уровне, уровне операционной системы, на уровне приложений и на уровне дополнительных источников). Обращение к нескольким источникам информации может значительно повысить достоверность решений, связанных с обнаружением атак и защитой компьютерной сети.

Концептуальный анализ любой проблемы обучения включает, прежде всего, анализ источников знаний. В рассматриваемой задаче главными источниками данных являются экспериментальные данные, описывающие деятельность пользователей и "историю" вторжений, которые совместно составляют данные регистрации, соответствующие случаям, интерпретируемым как "нормальная деятельность", "подозрительная деятельность" и "атака". Поскольку в проекте система обнаружения вторжений рассматривается как мультисенсорная система объединения знаний, полученных из различных источников, то соответствующая задача обучения является задачей распределенного обнаружения знаний, которая реализуется на основе многоагентной архитектуры.

Известные алгоритмы обучения обнаружению атак являются вычислительно сложными. Существенного снижения сложности можно достигнуть за счет предварительного анализа информативности данных и выявления потенциально наиболее представительных атрибутов и признаков деятельности субъектов (внутренних и внешних по отношению к защищаемой системе), которые проявляются в данных регистрации. В качестве таких признаков целесообразно выделять повторы определенных событий и их комбинации, неправильные команды и команды, неадекватные текущей ситуации, признаки, свидетельствующие об использовании известных уязвимостей, о неадекватности параметров и содержания сетевого трафика, заметные отклонения значений атрибутов, характеризующие профиль работы субъектов системы (например, время и дата работы, адрес субъекта, используемые субъектами сервисы, характеристики системных ресурсов, в том числе, данные о загрузке центрального процессора, обращении к оперативной и дисковой памяти, к файлам, телекоммуникационным портам и т. д.) и необъяснимые проблемы (например, выход из строя маршрутизатора, перезагрузка сервера, невозможность запуска системного сервиса и др.). Вовлечение экспертов на данном этапе обучения может существенно сократить перебор паттернов и размерность данных, используемых для обучения.

Основные особенности и трудности рассматриваемой задачи обучения обусловлены распределенным характером и наличием зависимостей временного характера. Такие данные в проекте представляются в терминах обобщенной модели временных последовательностей событий, которая определяет структуру паттернов подлежащих обнаружению в процессе обучения.

2. Разработка многоуровневой онтологии задач обучения.

Непротиворечивая и согласованная работа крупномасштабной распределенной системы, основанной на знаниях, может быть обеспечена только в том случае, если распределенные сущности, составляющие систему, в состоянии понимать друг друга. В соответствии с современными представлениями, такая работа может быть организована наилучшим образом, только с помощью подхода, базирующегося на использовании онтологий. Именно онтология в состоянии обеспечить совместную непротиворечивость локальных баз знаний, целостность знаний и однозначную и корректную интерпретацию терминов, составляющих язык обмена сообщениями между сущностями распределенной системы.

Многоуровневая онтология задачи обучения обнаружению вторжений объединяет в единую взаимосвязанную систему комплекс базовых понятий, формирующих верхние уровни модели знаний, с которыми манипулируют компоненты разрабатываемой системы. Эта онтология охватывает понятия из *проблемной* онтологии "Data fusion" и "Data fusion learning", а также из онтологии *предметной* области "Intrusion detection" и "Intrusion detection learning". Разработанная онтология служит базисом для построения верхнего уровня представления распределенных знаний, которые являются общими ("*shared knowledge*") для компонент системы обучения обнаружению вторжений. Этот уровень знаний позволяет, с одной стороны, обеспечить целостность распределенной базы знаний, а с другой стороны, "взаимопонимание агентов" при обмене сообщениями.

3. Разработка концептуальных моделей типовых агентов многоагентной обучающей системы

Обучаемая система обнаружению вторжений, рассматривается как мультисенсорная система объединения данных, полученных из различных источников. Эта система формирует решения на основе многоуровневой модели обработки входных данных (входного трафика сети и данных аудита). Применительно к такому взгляду на обучаемую систему разработана концептуальная модель многоагентной системы обучения.

В процессе исследований проанализированы структуры данных обучения и выбрано множество адекватных методов (алгоритмов) обучения, которые позволяют справиться с рассматриваемой задачей обучения. Это множество включает в себя как некоторые из известных методов (например, некоторые из множества *ID3*, *C4.5*, *AQ*, *CN2*, *бустинг*, а также методологию мета-классификации), так и методы, которые разработаны или разрабатываются исполнителями данного проекта (например, метод визуального аналитического обнаружения закономерностей – –*VAM*-метод, классификации, алгоритм *GK2*, алгебраические байесовские сети).

Определен состав типовых классов (классов агентов) многоагентной системы обучения обнаружению вторжений, а также их функциональности и роли в системе обучения. Множество классов агентов включает в себя следующие кассы: класс агентов управления данными обучения; класс агентов тестирования классификаторов; класс агентов формирования мета-данных; классы обучающих агентов.

8. Ход выполнения технических работ за рассматриваемый год

Ход выполнения работ за рассматриваемый год полностью соответствует плану работ как по содержанию, так и по срокам завершения предусмотренных этапов работ.

Основные достижения за рассматриваемый год

Основные достижения за рассматриваемый год связаны с решением запланированных задач. Это следующие задачи:

1. Разработка архитектуры многоагентной обучающей системы и математических методов, реализующих функциональности типовых обучающих агентов.
2. Разработка протокола взаимодействия интеллектуальных обучающих агентов (протокола переговоров) для обобщения решений отдельных агентов в соответствии с процедурой мета-классификации и разработка архитектуры многоагентной обучающей системы в целом.
3. Разработка объектно-ориентированного проекта многоагентной системы обучения обнаружению атак.

Основные результаты, полученные в рамках вышеназванных задач в течение первого года исследований, таковы.

1. Анализ и разработка формальных моделей и архитектур частных типовых агентов многоагентной обучающей системы.

Исследования по решению данной задачи проводились в двух направлениях.

Первое направление связано с разработкой *алгоритмического базиса* для решения задач обучения обнаружению вторжения. С этой целью проводилось изучение известных методов, опубликованных в последние годы, которые используются в настоящее время различными авторами в задачах обнаружения вторжений. Изучались также другие известные методы, которые, хотя и не используются пока в данной задаче, но которые потенциально могут быть использованы с этой целью. При этом основное внимание уделялось методам извлечения часто встречающихся паттернов (подпоследовательностей) из временных последовательностей событий. В частности, проведена алгоритмизация и разработано экспериментальное программное обеспечение для метода извлечения паттернов из последовательностей, известного под названием *FP-growth*, который является наиболее эффективным из известных методов поиска часто встречающихся эпизодов. Кроме того, проводилась работа по адаптации методов, которые были ранее разработаны исполнителями данного проекта для решения более традиционных задач обучения объединению данных, полученных из различных источников. В частности, разработано экспериментальное программное обеспечение для метода *GK2*, предложенного авторами. Этот метод совместно с методом визуального аналитического извлечения правил из данных был протестирован на примере задачи обучения обнаружению атак на компьютер на базе тестовых данных, которые использовались в соревновании программ обучения KDD Cup-1999. Велись исследования по использованию этих методов в задаче обучения мета-классификации, которая отвечает уровню объединения локальных решений системы обнаружения вторжений, полученных на базе частных источников информации.

Второе направление имеет целью разработку формальных моделей и архитектур частных классов агентов многоагентной системы обучения обнаружению вторжений. Эти модели и архитектуры разработаны для следующих классов агентов:

- класс агентов управления данными обучения,
- класс агентов тестирования классификаторов,
- класс агентов формирования мета-данных,
- классы обучающих агентов, а именно, (а) класс *агентов,* предназначенных для обучения классификаторов атак, входные данные которых представлены последовательностями событий, упорядоченных во времени и (б) класс *агентов,* предназначенных для обучения классификаторов, которые работают с описанием атак в форме вектора признаков.

Каждый из этих классов агентов имеет стандартные компоненты, которые отвечают за получение, синтаксическую обработку и отсылку сообщений, которыми обмениваются агенты, а также стандартные механизмы семантической обработки сообщений, которые реализуются с помощью абстрактных автоматов, называемых (в соответствии с терминологией, принятой в языке *UML*) "*машинами состояний*" ("*state machine*"). Индивидуальность в модели каждого класса агентов определяется конкретной структурой и конкретным содержанием компонент "машины состояний" (алфавитов состояний, переходов, функций смены состояний и сопутствующих действий, описываемых в терминах сценариев поведения, и др.) и содержимым баз данных. Разработаны модели всех стандартных компонент классов агентов, а также специализированные компоненты классов агентов, перечисленных выше. Эти модели описаны формально в терминах *USE CASE DIAGRAMs*, которые визуально представляют функциональное поведение компонент классов агентов, описанных формально в нотации языка *UML*.

2. Анализ задачи обучения обнаружению вторжений в компьютерную сеть и выявление специфических проблем, связанных с реализацией обучения.

Задача обучения обнаружению вторжений в компьютерную сеть во многом отличается от типичной задачи обнаружения знаний на основе накопленных данных. Основные специфические особенности этой задачи обусловлены спецификой данных, которые могут использоваться для обучения. Среди этих особенностей, прежде всего, следует выделить такое их свойство, как распределенность и гетерогенность, хотя, в не меньшей мере, эти особенности определяются также и структурами данных, доступных для использования в качестве обучающих данных.

Следы несакционированной деятельности пользователей (ошибочные команды и атаки на компьютерную сеть) проявляются в многочисленных распределенных источниках данных (в данных файлов *tcpdump*, вызовах операционной системы, данных аудита, логах приложений и т.д.). Эти данные могут быть представлены в различных структурах (последовательностях с временными метками, последовательностях без временных меток, реляционных,

транзакционных, и др.), они могут быть измерены в различных шкалах (булевых, категориальных, линейно упорядоченных, числовых), они могут быть различной точности и содержать неопределенности различного типа, быть неполными и содержать пропуски в данных. Эти особенности влекут специфические проблемы, которые нужно решать при построении систем обучения обнаружению вторжений.

Распределенность и *гетерогенность* данных создают ряд специфических проблем, на первый взгляд не связанных непосредственно с выбором алгоритмов обучения и классификации, однако на практике в значительной степени на них влияющих. Первая из них – это проблема обеспечения *глобальной однозначности семантики* терминов, используемых при спецификации данных локальных источников. Эта проблема возникает из-за того, что спецификация данных выполняется распределенными пользователями. Они могут использовать одинаковые термины в различном смысле и, наоборот, для одного и того же понятия могут использовать различные названия. В соответствии с современными взглядами, для решения проблемы однозначного понимания терминов необходимо использование высокоуровневой модели знаний, разделяемой всеми сущностями системы, т.е. согласованной и доступной каждой из них. Обычно эта база знаний строится в терминах проблемной онтологии, дополненной онтологией приложения и онтологией задач.

Поскольку в рассматриваемом классе задач данные распределены, то технология формирования непротиворечивой онтологии, согласованной на множестве источников данных, представляет собой специфическую задачу, которая во многих отношениях является новой. Такая технология разработана. Основная идея этой технологии состоит в том, что в онтологии приложения выделяется часть, которая согласована с проблемной онтологией и является общей для всех источников данных. Кроме того, для каждого источника данных строится компонента онтологии, которой "обладает" только этот источник, и которая недоступна для остальных компонент. Разработаны протоколы переговоров специализированных агентов, которые ответственны за формирование вышеназванных компонент онтологии приложения.

Вторая проблема известна как *проблема идентификации сущностей.* Она возникает в связи с тем, что информация об одной и той же сущности (ситуации, состоянии объекта и т.п.) представляется в распределенной форме, а потому необходимо иметь специальные механизмы для отождествления компонент данных и сущности, описание которой они представляют. Эта проблема должна решаться для того, чтобы можно было *выбирать* и *анализировать совместно* информацию об одной и той же сущности. Заметим, что при этом информация о некоторых сущностях в отдельных источниках может отсутствовать.

В данном проекте проблема идентификации сущностей решается следующим образом. В онтологии приложения для каждой сущности вводится свой идентификатор ("*ID entity"*). Этот идентификатор сущности рассматривается как ее первичный ключ (по аналогии с первичным ключом реляционной таблицы). Для каждого такого идентификатора в онтологии приложения определяется правило, каким образом может быть вычислено значение этого ключа. Например, в качестве такого правила может быть выбрана уникальная комбинация атрибутов этой сущности. Такое правило для каждой сущности задается применительно к каждому источнику данных, что позволяет связать идентификатор сущности, представленной в распределенном виде, с ее фрагментами в различных источниках. Такое правило описывает:

(1) как вывести значение первичного ключа сущности в локальном источнике на основании идентификатора сущности, используя значения атрибутов этой сущности в источнике, и

(2) как вывести значение идентификатора сущности по значению первичного ее ключа в локальном источнике.

Еще одна проблема возникает из-за того, что в каждом из источников информация об одной и той же *сущности может представляться в терминах атрибутов различной природы* (изображения, сигналы, экспертные данные и пр.). Эта проблема решается в проекте с помощью выбора подходящей схемы объединения решений локальных классификаторов. Возможные схемы объединения классификаторов построены.

Наконец, еще одна проблема возникает в связи с тем, что множества атрибутов различных источников могут пересекаться, и при этом *одинаковые* или "*сходные*" свойства в различных источниках могут быть *представлены в различных шкалах измерения* (номинальной, числовой и т.д.), с различной точностью и с другими отличиями. Эта проблема решается с помощью согласования шкал и единиц измерения на мета-уровне с последующим пересчетом различных представлений соответствующих атрибутов.

Возникают и другие специфические проблемы, обусловленные гетерогенностью и распределенностью данных.

В архитектуре системы обучения обнаружению вторжений используются специальные агенты, которые ответственны за решения всех вышеназванных задач. В частности, на мета-уровне за их решение отвечает агент *KDD master*, а на каждом локальном источнике данных в решении соответствующих подзадач принимают участие агенты *Data source managing.* Решение всех задач осуществляется на основе разработанных протоколов переговоров.

3. Анализ обучающих и тестовых данных для обучения обнаружению вторжений.

Проведенный анализ задачи обучения обнаружению вторжений обусловил выбор методов обнаружения знаний, в первую очередь, методов объединения решений на мета-уровне, а также на выбор архитектуры системы обучения. Однако, в большей мере, множество математических методов, которые покрывают потребности задач обучения обнаружению вторжений, определяется структурами данных, которые могут быть использованы для обучения. Проведен тщательный анализ структур данных, доступных для использования в процессе обучения. Результаты этого анализа кратко описываются далее.

Данные для обучения и тестирования систем обнаружения вторжений существуют в различных формах и могут быть получены из различных источников. Предложено использовать *три таксономии источников данных.* Эти таксономии базируются на использовании следующих признаков:

(1) местоположение источника данных или программы, генерирующей данные;

(2) уровень обработки (обобщения) данных;

(3) объект, информацию о котором несут данные.

Таксономия, в которой данные классифицируются в соответствии с местоположением источника данных или программы, генерирующей данные, включает в себя два основных типа данных: сетевые данные и данные, которые порождаются на конкретном хосте. Сетевые данные в свою очередь зависят от рассматриваемого уровня протоколов TCP/IP и типа используемого протокола. Данные, которые порождаются на хосте, включают в себя данные аудита операционной системы, системные логи и данные аудита конкретных приложений.

В таксономии, в которой данные классифицируются в соответствии с уровнем обобщения информации, выделяют три типа данных: первичные ("сырые") данные, предобработанные данные и обобщенные данные. К *первичным данным* относятся сетевой трафик, последовательности системных вызовов, а также другие данные подобного типа. К *предобработанным* данным относятся файлы *tcpdump* (для сетевых пакетов), предобработанные данные аудита операционной системы, системные логи, данные аудита различных приложений, запускаемых на хосте. К *обобщенным* данным относятся статистические данные, характеризующие различные источники.

В таксономии, в которой основой классификации являются объекты, к которым относятся данные, в одну группу объединяются сетевые данные (пакеты, соединения, сетевой трафик в целом), а в другую - все данные, порождаемые на хосте, в частности, трафик в пределах одного соединения, процессы, данные мониторинга работы отдельных пользователей, обращений к файлам, директориям, дискам, системному реестру и т.д.

Очевидно, что количество источников и объем данных в них, которые могут быть доступными для системы обнаружения вторжений, достаточно велики и все их вовлечь в процесс исследований невозможно.

В дальнейших исследованиях предполагается ограничиться следующими данными:

Данные сетевого уровня:

- предобработанные данные *tcpdump* для IP-, TCP-, UDP- и ICMP- пакетов и
- статистические данные, полученные обработкой файлов *tcpdump*.

Данные, которые порождаются на конкретном хосте:

- предобработанные данные аудита операционной системы и статистические данные, полученные его обработкой;
- системные логи (например, лог и статистические данные о командах пользователя и ресурсах, к которым он обращается, лог и статистические данные об ошибках входа в систему, логи и статистические данные о входах в систему и выходах, обо всех пользователях системы, запуски систем и выключения);

- данные аудита приложений (например, FTP-логи и FTP- статистические данные, TELNET-логи и TELNET статистические данные, Mail-логи и Mail-статистические данные, HTTP-логи и HTTP статистические данные, DNS-логи и DNS статистические данные).

Среди множества данных, упомянутых выше, встречаются четыре основных *типа структур данных*:

- данные типа временных последовательностей,
- последовательности (линейно упорядоченные события),
- реляционные данные и
- транзакционные данные.

Перечисленные выше структуры данных могут содержать компоненты, измеренные в различных *шкалах*, в частности:

- бинарные, или булевы,
- категориальные,
- линейно упорядоченные и
- вещественные.

4. Разработка математических методов, реализующих функциональности типовых агентов обучения.

Проведенный анализ особенностей работы с гетерогенными и распределенными данными, а также анализ источников данных с точки зрения особенностей представления данных в различных источниках, позволили обоснованно выбрать множество методов, которые позволяют решать задачи обнаружения знаний в таких данных.

Множество методов, покрывающих потребности задачи обучения обнаружению вторжений, включают в себя две группы методов:

(1) методы комбинирования решений, полученных на основании данных локальных источников;

(2) методы обучения классификаторов базового уровня.

Среди *методов комбинирования решений* были отобраны метод мета-классификации и метод, который предполагает анализ компетентности отдельных классификаторов базового уровня по отношению к каждому набору данных, используемому для принятия решения. Для обоих методов предложены модификации, которые учитывают особенности рассматриваемого приложения. Кроме того, для метода мета-классификации выполнена предварительная программная реализация и исследование метода на основе достаточно сложного набора данных KDDCup-99, использованного в 1999 году на соревнованиях программ извлечения знаний из данных.

Произведен также выбор *методов обучения классификаторов базового уровня*. Три из них отобраны для реализации.

1. Метод **FP-growth** (*Frequent pattern growth*), который ориентирован на извлечение часто встречающихся паттернов и ассоциативных правил из транзакционных баз данных. При дополнительной модификации (она разработана авторами этого метода) он может использоваться также для извлечения тех же знаний из последовательностей. Этот метод был предложен недавно, и по своим характеристикам он превосходит другие известные методы, в частности, методы, построенные на основе подхода, известного под названием **Apriori**. Такое заключение сделано как на основании теоретического анализа сложности, проведенного авторами метода, так и на основании экспериментов, проведенных авторами настоящей работы, в которых использовалась разработанная ими программная реализация. Этот метод включается в качестве компоненты разрабатываемой системы, которая называется *Server of learning methods*.
2. Метод **VAM** (*Visual Analytical Mining*), который эффективно работает с извлечением знаний из вещественных данных. Этот метод разработан авторами настоящего проекта и реализован программно. Свойства метода исследованы на нескольких приложениях, взятых из UCI репозитория. Этот метод также включен в качестве компоненты в программную компоненту разрабатываемой системы, названную *Server of learning methods*.
3. **GK2** алгоритм, предназначенный для извлечения правил из дискретных реляционных данных. Этот метод был разработан, программно реализован и исследован экспериментально авторами проекта. Метод теоретически обоснован и показал себя в

некоторых отношениях лучше известных методов аналогичного назначения. Преимущества данного метода по сравнению с известными методами в том, что он позволяет извлекать правила из данных, имеющих пропущенные значения без прогнозирования, как это обычно делается.

Для экспериментальной оценки двух последних методов, а также метода мета-классификации для объединения решений базовых классификаторов использован набор обучающих данных "KDDCup-99".

5. Разработка протокола взаимодействия агентов многоагентной обучающей системы.

Разработаны следующие типы *протоколов межуровнего взаимодействия (переговоров) интеллектуальных агентов*:

- протоколы для оперирования отдельными источниками данных;
- протоколы для управления созданием глобальной согласованной проблемной онтологии, разделяемых и частных компонентов онтологии приложения;
- протоколы для комбинирования решений основанных на источниках классификаторов.

Наиболее сложными протоколами являются протоколы для выполнения задачи создания глобальной согласованной проблемной онтологии, разделяемых и частных компонентов онтологии приложения. Эта задача заключается в создании и синхронизации основанных на источниках фрагментов онтологии приложения и ее синхронизации с онтологией проблемы слияния данных (Data Fusion - DF).

Эти протоколы служат для обеспечения взаимодействия элементов системы обучения обнаружению вторжений, размещенных на различных хостах, в процессе создания предварительной версии онтологии приложения и ее итеративной модификации при обеспечении согласованности. Были специфицированы протоколы для создания начальной (базовой) версии онтологии приложения (мы назвали их мета-протоколами) и протоколы для последующей синхронизации онтологии в процессе ее итеративной координации с локальными компонентами онтологии при их итеративной координации с локальными компонентами онтологии, а также при любой ее модификации.

Рассмотрены два мета-протокола: “сверху-вниз (восходящий)” и “снизу-вверх (нисходящий)”.

В первом случае эксперт мета-уровня, ответственный за формирование глобальной онтологии, создает ее базовый вариант, который включает список базовых сущностей приложения с минимально необходимым множеством атрибутов, и специфицирует идентификаторы сущностей. В случае использования многоагентной архитектуры системы IDLS, специальный агент ("KDD master"), управляемый экспертом мета-уровня, посылает базовый вариант локальных фрагментов онтологии приложения соответствующим агентам, расположенным в локальных источниках данных ("Data source managing agents" – DMAs), для анализа, коррекции, дальнейшего расширения и наполнения. Агенты DMA локальных источников, управляемые экспертами, выполняют модификацию и расширение полученной версии онтологии для обеспечения согласованности всей онтологии. Синхронизация изменений и расширений первой и последующих версий онтологии, сделанных агентами источников данных, выполняется мета-уровневым агентом шаг за шагом посредством обмена сообщениями с агентами источников данных. Содержание протокола синхронизации заключается в многофазовых переговорах, причем каждый агент источника реализует переговоры, основываясь только на разделяемой и своей собственной части онтологии приложения. Эти переговоры выполняются с использованием агента KDD master и приводят к разработке онтологии приложения, которая согласуется с проблемной онтологией и не имеет противоречий на уровне приложений. Все эти процедуры выполняются под наблюдением и при активном участии эксперта мета-уровня и экспертов локальных источников, взаимодействующих через своих агентов. После обработки указанной выше информации агент DMA локального источника подготавливает свои предложения относительно модификации и/или расширения локальных компонентов онтологии предметной области и посылает эти предложения агенту KDD master.

При использовании протокола “снизу-вверх” эксперты локального источника сначала формируют базовые варианты онтологии приложения в отношении своих разделяемых частей и собственных частей онтологии, а затем агент KDD master под наблюдением эксперта мета-уровня выполняет объединение, координацию и коррекцию полученных компонент онтологии приложения для подготовки ее следующего базового варианта. После этого соответствующие части этого варианта посылаются агентам DMA локальных источников для дальнейшей

коррекции в случае необходимости. Последующая работа выполняется аналогично описанным выше шагам протокола.

В обоих протоколах центральным компонентом является их часть, которая реализует синхронизацию компонент онтологии приложения, предложенных агентом KDD master и агентами DMA локальных источников онтологии приложения.

При рассмотрении взаимодействий между компонентами IDLS были учтены возможное пространственное распределение источников данных и наличие ненадежных каналов коммуникации между источниками данных и хост-сервером мета-уровня. Для реализации механизмов взаимодействия, функционирующих при таких условиях, были использованы протоколы, основанные на двухфазных “ленивых” (lazy) транзакциях. Эти протоколы в значительной степени схожи с протоколами синхронизации баз данных, за исключением того, что в используемых протоколах синхронизации невозможна верификация модификаций, выполняемых на сервере мета-уровня. Основное право по принятию решений относительно модификации онтологии лежит на эксперте приложения верхнего уровня, который несет основную ответственность за формирование и поддержку глобальной онтологии приложения. В его обязанности также входит периодический просмотр и верификация модификаций онтологии, предложенных экспертами приложения, работающими с локальными источниками.

6. Разработка процедур обобщения частных решений агентов в соответствии с подходом на основе мета-классификации.

Разработанные процедуры обобщения частных решений агентов системы обучения обнаружению вторжений базируются на специфицированной иерархии взаимодействия частных классификаторов в процессе осуществления глобального решения на базе иерархического комбинирования решений классификаторов нижнего уровня.

Для обобщения частных решений агентов было проанализировано несколько методов комбинирования решений классификаторов базового уровня решающих одну и ту же задачу. Эти методы можно условно разделить на четыре группы:

1. Методы, использующие в той или иной форме голосование.
2. Методы, основанные на использовании вероятностных или нечетких алгоритмов.
3. Методы мета-обучения (мета-классификации), основанные на использовании мета-данных. В зарубежной литературе такие методы объединяются термином "stacked generalization".
4. Методы, использующие оценку компетентности классификаторов.

Методы мета-классификации и методы, использующие оценку компетентности классификаторов, были выбраны и адаптированы для использования в системе обучения обнаружению вторжений. Необходимо отметить, что эти методы могут использоваться напрямую при обучении обнаружению вторжений из-за особенностей данных. Поэтому эти методы были приспособлены для реализации процедур распределенного обучения и принятия решения.

7. Разработка архитектуры многоагентной обучающей системы.

Разработанная архитектура многоагентной системы обучения обнаружению вторжений включает компоненты источников локальных данных и компоненты мета-уровня.

Основными компонентами архитектуры системы являются следующие агенты:

- *Агент-мастер обучения обнаружению вторжений*, реализующий функции поддержки разработки распределенной онтологии обучения обнаружению вторжений, поддержки разработки мета-модели принятия решений по обнаружению вторжений и управления распределенным обучением;
- *Агент мета-обучения обнаружению вторжений*, предназначенный для управления распределенным обучением, поддержки разработки мета-модели принятия решений по обнаружению вторжений и рассылки разработанных структур принятия решений агентам обучения обнаружению вторжений локального уровня;
- *Агент принятия решений (классификации) по обнаружению вторжений на мета-уровне*, выполняющий управление распределенным обучением и объединение решений классификаторов базового уровня (уровня источников данных);
- *Агент управления комбинированием данных,* разрабатывающий мета-модель принятия решений по обнаружению вторжений, управляющий распределенным обучением и объединением решений классификаторов уровня источников данных.

- *Агент обучения базовых классификаторов*, участвующий в разработке мета-модели принятия решений по обнаружению вторжений и управляющий обучением базовых классификаторов;
- *Агент принятия решений (классификации) по обнаружению вторжений локального источника данных*, управляющий принятием решений базовыми классификаторами, и реализующий функции принятия решений отдельными базовыми классификаторами.
- *Агент управления данными локального источника*, участвующий в процессе принятия решений по обнаружению вторжений, разработке распределенной онтологии и мета-модели принятия решений системы обучения обнаружению вторжений, процессе принятия решений, а также реализующий мониторинг источников данных с целью анализа наличия новых данных.

8. Разработка объектно-ориентированного проекта многоагентной системы обучения обнаружению атак.

Объектно-ориентированный проект системы обучения обнаружению вторжений задан в терминах *Uses cases*–диаграмм, *Collaboration*–диаграмм, *State-chart*–диаграмм и *Component*–диаграмм.

Объектно-ориентированный проект системы обучения обнаружению вторжений включает следующие спецификации:

- схема функционирования системы обучения обнаружению вторжений на верхнем уровне;
- схема принятия решений по обнаружению вторжений базовыми классификаторами;
- схема принятия решений по обнаружению вторжений мета-классификатором;
- схемы поведения агентов системы обучения обнаружению вторжений при подготовке данных, поиске информативных признаков, получении и обработке мета-характеристик источников данных, обучении классификаторов и мета-классификатора, трансформации шкал измерения признаков, извлечении правил, оценке качества работы базовых классификаторов, означивании истинностных значений правил баз знаний базовых классификаторов, разработке мета-модели принятия решений, формировании распределенной онтологии; и др

В целом они обеспечивают полную спецификацию компонентов системы, необходимую для написания программного кода.

9. Существующее положение дел с выполнением технических работ

Ход выполнения работ полностью соответствует предусмотренному плану и в коррекции не нуждается.

10. Сотрудничество с зарубежными партнерами

В соответствии с планом работ партнеру представлен один промежуточный отчет (1 июня 2002), в котором представлены соответствующие результаты исследований.

Исполнители проекта совместно с представителями партнера участвовали в семинаре по обсуждению результатов исследований за первый год в феврале 2002 г. в организации партнера в США.

11. Выявленные проблемы и предложения относительно их устранения

Нет

12. Перспективы дальнейшего развития разработанной технологии/научного исследования

Перспективы дальнейшего сотрудничества будут обсуждаться на встрече с представителями Партнера и Министерства обороны США ориентировочно в апреле 2003. Предложения по дальнейшему сотрудничеству были представлены Партнеру в сентябре 2002.

Приложение 1. Наглядные материалы, прилагаемые к основному тексту

Нет

Приложение 2. Другая дополнительная информация к основному тексту

Краткое содержание Промежуточных отчетов, представленных партнеру

Промежуточный отчет №2

Приложение 3. Резюме статей и докладов, опубликованных за рассматриваемый год

Список публикаций

1. Городецкий В.И., Карсаев О.В, Котенко И.В., Хабалов А.В. Программный инструментарий для разработки и реализации многоагентных систем. // Lecture Notes in Artificial Intelligence, Vol.2296, Springer Verlag, 2002. P.121-130.

Абстракт. В статье описывается разработанная технология и программное средство для проектирования и реализации основанных на знаниях многоагентных систем.

Программное средство включает две компоненты: “Типовой агент” и "Инструментарий разработки многоагентных систем - *Multi-agent System Development Kit*" (MAS DK). Первая компонента включает повторно используемые классы на Visual C++ и Java, а также типовые структуры баз данных и знаний. Вторая компонента содержит несколько ориентированных на пользователя редакторов, служащих для формальной спецификации прикладных многоагентных систем (МАС), находящихся на этапе проектирования и реализации в среде конкретной компьютерной сети. Разработанная технология и MAS DK были использованы при проектировании и реализации прототипов МАС для защиты компьютерных сетей, обнаружения вторжений и моделирования распределенных атак.

2. Городецкий В.И., Котенко И.В. Многоагентные системы для обеспечения безопасности компьютерных сетей // IEEE ICAIS-02. IEEE International Conference “Artificial Intelligence Systems”. Proceedings. IEEE Computer Society. 2002. P.297-302.

Абстракт. В статье рассмотрены приложения многоагентной технологии для проектирования и реализации многоагентных систем (МАС), предназначенные для кооперативного решения критически важных задач в области обеспечения безопасности компьютерных сетей. Эти МАС представлены базирующейся на агентах системой моделирования атак на компьютерные сети, многоагентной системой обнаружения вторжений и многоагентной системой обучения обнаружению вторжений. Каждая из этих МАС базируется на строгом формальном подходе, предложенном авторами, и спроектирована и реализована в виде программного прототипа на основе общей технологии и программного средства “Инструментарий разработки многоагентных систем - Multi-agent System Development Kit”, разработанного авторами работы. В статье проводится обзор указанных МАС, и анализируются преимущества использования многоагентной архитектуры для обеспечения безопасности компьютерных сетей.

3. V.Gorodetski. Multi-agent Data Fusion: Design and Implementation Issues. 5th International Conference on Information Fusion (Fusion-2002), CD Proceedings of the section "*AFOSR Information Fusion Initiative*". Annapolis, MD, USA, July 3-6, 2002. Abstract

Абстракт. Нижеследующие аспекты технологии разработки и программной реализации многоагентных систем слияния данных составляют содержание работы:
(1) Каким образом может быть разрешена проблема слияния решений, полученных на основе распределенных источников, содержащих гетерогенные данные? В частности, каким образом может быть обеспечено однозначное понимание сообщений, которыми обмениваются агенты многоагентной системы слияния данных? Как может быть решена проблема идентификации данных об одном и том же объекте, представленном фрагментами в различных базах данных? Каким образом можно справиться с разнообразием шкал измерения и структур представления данных в различных источниках? В рамках основного вопроса возникает также ряд других частных задач.
(2) Каким образом решается задача распределенного обучения, в частности, каким образом организуется слияние данных, полученных на основе локальных источников данных, использующих различные данные и методы классификации?
(3) Какова архитектура системы слияния данных и каким образом целесообразно распределить различные задачи между ее компонентами?
В работе также рассматривается предложенная технология поддержки разработки многоагентных приложений, которая, в частности, используется для разработки многоагентных систем слияния данных. Даются сведения о приложениях, использованных для отработки предложенных подходов и технологии в целом.

4. V.Gorodetski, O.Karsayev and V.Samoilov. Multi-agent Data Fusion Systems: Design and Implementation Issues. Proceedings of the 10th International Conference on Telecommunication Systems - Modeling and Analysis, Monterey, CA, October 3-6, vol.2, pp.762-774, 2002.

Абстракт. Рассматривается задача принятие решений на основе совместной обработки данных, полученных из распределенных источников об одном и том же объекте. Основное внимание уделяется технологическим аспектам разработки и программной реализации систем решающих задачи названного класса. В частности, в работе представлен анализ специфических проблем, возникающих при разработке таких систем, методы их решения, а также многоагентная архитектура системы распределенного обучения, имеющего целью построение систем объединению данных. Отдельно проанализированы особенности проблемы распределенного обучения, а также известные подходы к решению ее центральной задачи – обучение объединению решений классификаторов, принимающих решения на основе данных отдельных источников.

5. Котенко И.В. Многоагентные технологии для обеспечения обнаружения вторжений в компьютерные сети // X Российская научно-техническая конференция (по Северо-западному региону) “Методы и технические средства обеспечения безопасности информации”. Тезисы докладов. Санкт-Петербург. Издательство СПбГПУ. 2002. С.44-45.

   Абстракт. В работе представлены основные приложения многоагентных систем в области обнаружения вторжений в компьютерные сети, разработанные в Лаборатории интеллектуальных систем СПИИРАН: (1) агентно-ориентированная система моделирования атак на компьютерные сети; (2) многоагентная система обнаружения вторжений (атак); (3) многоагентная система обучения обнаружению вторжений в компьютерные сети. Каждое из разработанных приложений базируется на предложенных учеными лаборатории формальных моделях и архитектурах и реализовано с использованием собственного инструментария разработки многоагентных систем MAS DK (“Multi-Agent System Development Kit”).

6. Котенко И.В. Восстановление формальных грамматик, задающих сценарии компьютерных атак, по прецедентам // Искусственный интеллект-2002. Материалы научно-технической конференции. Том.1. Таганрог: Изд-во ТРТУ, 2002.

   Абстракт. В работе анализируются подходы к синтезу формальных грамматик, задающих сценарии атак на компьютерные сети, в том числе: (1) индуктивное восстановление по множеству прецедентов формальными методами; (2) задание экспертом на основе знаний о намерениях злоумышленника и возможных способах реализации этих намерений; (3) комбинирование первого и второго способа. Представлены примеры использования алгоритмов восстановления грамматик, специфицирующих атаки.

7. Котенко И.В. Восстановление формальных грамматик, задающих сценарии компьютерных атак, по прецедентам // Международный научно-теоретический журнал “Искусственный интеллект”. № 3, 2002.

   Абстракт. В работе предлагается подход к синтезу (генерации) формальных грамматик, определяющих модели атак на компьютерные сети. Выделено две группы алгоритмов восстановления грамматик: алгоритмы восстановления грамматик перечислением; алгоритмы восстановления грамматик индукцией. В качестве наиболее адекватного выбран метод восстановления регулярной грамматики индукцией по положительному образцу (метод Фельдмана). Дано описание его алгоритмической реализации. Для демонстрации возможностей использования алгоритмов восстановления грамматик, служащих для описания атак на компьютерные сети, рассмотрено несколько прецедентов атак. Представлен пример использования алгоритма восстановления грамматик для спецификации атак на компьютерные сети. Данный подход предлагается использовать при построении многоагентной системы моделирования атак на компьютерные сети.

8. Котенко И.В., Карсаев О.В., Самойлов В.В. Онтология предметной области обучения обнаружению вторжений в компьютерные сети // Международная конференция по мягким вычислениям и измерениям. SMC’2001. Сборник докладов. Том 1. СПб: СПбГЭТУ, 2002. С.255-258.

   Абстракт. В соответствии с современными взглядами на информационные технологии онтология является одной из наиболее важных компонентов каждой информационной системы, в первую очередь, когда данная система относится к классу распределенных, имеет большой размер и основывается на знаниях. В работе рассматривается разработанная архитектура многоагентной системы обучения обнаружению вторжений и онтология предметной области обучения обнаружению вторжений в компьютерные сети, предназначенная для построения этой системы. Онтология формирует высокоуровневую концептуальную модель распределенных знаний, представленных в виде структурированного множества базовых понятий с ясной и четко определенной семантикой, независимой от реализационных аспектов конкретного приложения. Специфика рассматриваемой предметной области отражается в комбинировании знаний и, следовательно, онтологий, представляющих три области, а именно, “Слияние (объединение) данных и обучение слиянию (объединению) данных”, “Обнаружение вторжений” и “Обучение обнаружению вторжений”. Указанные онтологии представлены в работе.

Приложение 4. Информация о патентах и правах на собственность.

Нет

III. Приложения

1.1. Сводка по участию персонала

Участник	Кол-во дней
1-ая категория	
Городецкий Владимир Иванович	30
Карсаев Олег Владиславович	50
Котенко Игорь Витальевич	35
Маньков Евгений Викторович	50
Конюший Виктор Григорьевич	50
Ткач Анатолий Федорович	???
Дружинин Аркадий Васильевич	20
Бакурадзе Дмитрий Викторович	???
Солодухин Анатолий Николаевич	???
Самойлов Владимир Владимирович	27
2-ая категория	
Хабалов Алексей Владимирович	72

1.2. Приобретенное основное оборудование

Нет

IV. ПОДПИСИ

Руководитель исследований по задаче 2

доктор технических наук профессор
И.В. Котенко