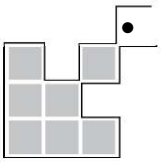

СИСТЕМЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

УДК 355.01: 004.056

С.В. Жуков, В.А. Маклаев, П.И. Соснин

МЕТОЧНАЯ зАщИТА ПРОЕКТНЫХ зАДАЧ В РАЗРАБОТКЕ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Жуков Станислав Владиславович, *окончил факультет экономической кибернетики Московского экономико-статистического института (МЭСИ). Соискатель Ульяновского государственного университета. [e-mail: texthapb@mail.ru].*

Маклаев Владимир Анатольевич, *кандидат технических наук, окончил радиотехнический факультет Ульяновского политехнического института. Генеральный директор ФНПЦ ОАО «НПО «Марс». Имеет статьи в области САПР. [e-mail: mars@mv.ru].*

Соснин Петр Иванович, *заслуженный работник высшей школы РФ, доктор технических наук, профессор, окончил радиотехнический факультет Ульяновского политехнического института. Заведующий кафедрой «Вычислительная техника» УлГТУ. Имеет многочисленные труды в области концептуального проектирования автоматизированных систем. [e-mail:sosnin@ulstu.ru].*

Аннотация

В статье представлен комплекс средств меточной защиты проектных задач в разработке автоматизированных систем. Специфику предлагаемых средств определяет единообразное иерархическое представление задач и оргструктуры коллектива разработчиков, оперативное взаимодействие между которыми обеспечивается программными агентами. Комплекс средств защиты реализован как расширение инструментальной моделирующей среды WIQA.Net, обслуживающей концептуальное проектирование автоматизированных систем.

Ключевые слова: автоматизированное проектирование, информационная безопасность, меточная защита, программные агенты, проектные задачи.

Stanislav Vladislavovich Zhukov, *graduated from the Faculty of Economical Cybernetics of Moscow Institute of Economics and Statistics; doctoral candidate at Ulyanovsk State University. e-mail: texthapb@mail.ru.*

Vladimir Anatolyevich Maklaev, *Candidate of Engineering, graduated from the Faculty of Radioengineering of Ulyanovsk Polytechnic Institute; Director General of Federal Research-and-Production Center Open Joint-Stock Company 'Research-and-Production Association 'Mars'; author of articles in the field of CAD. e-mail: mars@mv.ru.*

Petr Ivanovich Sosnin, *honored worker of the Higher School of the Russian Federation, Doctor of Engineering, Professor; graduated from the Faculty of Radioengineering of Ulyanovsk Polytechnic Institute; head of the Chair 'Computers' at Ulyanovsk State Technical University; author of numerous papers in the field of conceptual design of computer-aided systems. e-mail: sosnin@ulstu.ru.*

Abstract

The article presents a system of means for label security of design tasks when developing com-puter-aided systems. The peculiarity of the suggested means determines homogenous hierar-chical presentation of the tasks and organizational

structure of project teams. The operational interaction among developers is provided by software agents. The system of security means is implemented as an extension of the tool modeling environment wIQA.Net servicing conceptual design of computer-aided systems.

Key words: computer-aided design, information security, label security, software agents, design tasks.

ВВЕДЕНИЕ

Инструментально-технологические комплексы (Итк), обслуживающие разработку автоматизированных систем (АС), относятся к классу автоматизированных систем типа collaborative Development Environments (cDE, среды коллективной разработки), специфика которого детально раскрыта в публикации [1]. В то же время, учитывая специфику, в разработках Итк следует также использовать опыт разработок АС, нашедший свое нормативное представление в богатейшей коллекции стандартов.

Наиболее последовательно и полно опыт лучших практик разработок АС встроен в мастер-методологию Rational unified process (Rup), созданную корпорацией IBM [2]. Более 10 лет этот продукт доминирует на рынке Итк и используется как образец для построения новых инструментальных систем такого типа, в которые вкладываются все новые и новые средства, обеспечивающие повышение степени успешности разработок АС.

к числу таких средств относится и постоянно развивающаяся система стандартов информационной безопасности, находящих свое материальное воплощение в Итк, причем как в создании Итк, так и в их использовании [4], [5] и [7].

Анализ отмеченных информационных источников позволяет выделить следующие особенности, которые должны приниматься в расчет при создании Итк и их использовании:

1. Жизненный цикл любой разрабатываемой АС должен быть представлен системой проектных задач $S(\{Z_i\})$, оперативное решение которых обеспечивается коллективом проектировщиков $T(\{D_j\})$.

2. Средства управления жизненным циклом должны включать динамическую систему назначений ^({^Д), связывающую каждого проектировщика D_j с определенным подмножеством решаемых задач $S^j(\{Z^j\})$ в любой текущий момент времени.

3. Любая задача Z_i жизненного цикла АС решается членом D_j коллектива проектировщиков $T(\{D_j\})$, ответственным за ее решение, который может обращаться за помощью к другим членам коллектива или к другим лицам (stakeholders, стейкхолдерам), заинтересованным в разработке АС.

4. Система назначений $S(\{4\})$ должна поддерживать работы в рамках совокупности ролей, исполняемых проектировщиками и стейкхолдерами в их отношениях с проектными задачами $\{Z_i\}$.

Выделенные особенности были использованы авторами в создании системы средств меточной защиты, обеспечивающей информационную безопасность в оперативной работе проектировщиков с проектными задачами. Специфику средств защиты определяет единообразное иерархическое представление систем $S(\{Z_i\})$ и $T(\{D_j\})$ в их оперативных отношениях, которые обеспечиваются программными агентами, описанными ниже.

МОДЕЛИРОВАНИЕ ПРОЕКТНЫХ ЗАДАЧ и КОЛЛЕКТИВА ПРОЕКТИРОВЩИКОВ

Разработанные средства меточной защиты включены в состав инструментальной среды WIQA [6], обслуживающей концептуальное проектирование АС. В статье эти средства описаны схематично с позиций защиты проектных задач от несанкционированных действий.

В среде WIQA жизненный цикл проектируемой АС представляется деревом задач, каждая из задач которого существует в форме ее вопросно-ответной модели (вопросно-ответного протокола рассуждений проектировщика, использовавшихся им в процессе решения задачи). текущее состояние дерева задач и всех вопросно-ответных моделей (QA-моделей) регистрируется в вопросно-ответной базе (QA-базе) данных на сервере, а их составляющие доступны проектировщикам на каждом клиентском месте в форме, представленной на рисунке 1.

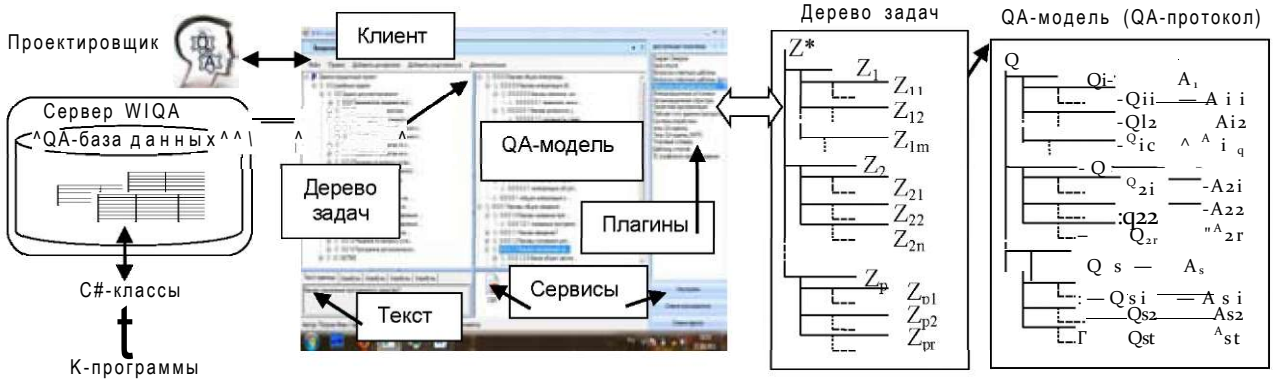

Рис. 1. Представление дерева проектных задач и их QA-моделей

Центральное место в представлении занимают задачный и логико-лингвистический виды, моделирующие систему постановок задач и вопросно-ответную структуризацию их решений с поясняющими диаграммными схемами и иллюстрациями других типов (если это необходимо или полезно). Проектировщику на его рабочем месте доступны интерактивные объекты трех типов - задачи Z_n, вопросы Q_m и соответствующие ответы A^m. Каждый из объектов имеет уникальное системное имя, которое создается автоматически при его порождении и включает символ типа ("Z", "Q" или "A"), дополненный индексами, регистрирующими место в иерархии. Текущее состояние дерева задач и QA-моделей является динамическим образованием, которое порождается проектировщиками в процессах пошаговой детализации проектного задания. Заметим, что доступ к QA-базе данных открыт как для проектировщиков, так и для компьютерных программ (K-программ), например, программных агентов, использующих либо библиотеку функций доступа, либо C#-классы, порождаемые с помощью объектно-реляционных преобразований [6].

В любой момент времени существующие отношения между проектировщиками и проектными задачами обеспечиваются с помощью плагина «Оргструктура». Этот плагин позволяет назначить и зарегистрировать ответственность в коллективе проектировщиков за проектные задачи. Более того, он позволяет не только распределить задачи между членами коллектива, но и указать роль для любого проектировщика, который привлекается к решению определенной задачи. Другими словами, для каждой задачи Z_i этот плагин помогает назначить проектировщика D_j, ответственного за ее решение, а также помощников $\{D'^j_k\}$, которые способны внести вклад в решение задачи, если в этом появится необходимость. В этом случае помощники будут выполнять назначенную им роль.

Любая назначенная роль определяет не только соответствующую компетенцию, но и ответственность с точки зрения доступа к объектам Z-, Q- и A-типов ("свободный доступ", "только читать доступные объекты", "возможность создать новый объект" и "возможность модифицировать доступный объект").

Плагин «Оргструктура» обеспечивает реализацию возложенных на него функций с использованием части базы данных, представленной выше. Эта часть включает 24 отношения, определенных на базе 155 атрибутов. Для решения задачи назначений используется подмножество этих отношений и атрибутов. Кроме задачи назначений плагин «Оргструктура» поддерживает решение задач информационной безопасности, мониторинга процесса проектирования и коммуникативного взаимодействия.

ПРЕДЛАГАЕМЫЙ ПОДХОД к МЕТОЧНОЙ ЗАЩИТЕ ПРОЕКТНЫХ ЗАДАЧ

Описанная выше материализация проектных задач и назначений не затрагивает вопросов, связанных с проверкой прав доступа в оперативном взаимодействии проектировщиков с интерактивными Z-, Q- и A-объектами. Такая проверка относится к классу задач информационной безопасности, для решения которых следует использовать опыт, аккумулированный в современных международных стандартах безопасности, прежде всего в стандарте ISO/IEC 15408 (2005) [3].

Представляемая в статье задача проверки была сформулирована в постановке, которая согласована с этим стандартом в части его рекомендаций для профиля меточной защиты [8], а решение задачи, в виде комплекса средств, встроено в инструментальную среду WIQA. Специфику разработанной подсистемы меточной защиты определяют следующие особенности:

1. В решении задачи защиты используются дерево задач $S(\{Z_i\})$ и копия той части модели $T(\{D_j\})$ организационной структуры, в которой оперативно регистрируются назначения $S(\{A_k\})$ с учетом ролей. Администратор, отвечающий за решение задачи защиты на рабочем месте, включенном в корпоративную сеть среды WIQA, приписывает каждой проектной задаче метки и осуществляет мониторинг событий, связанных с нарушениями прав доступа.

2. Копия модели $T(\{D_j\})$ формируется в базе данных в форме (рис. 1), которая используется для дерева задач и их QA-моделей.

3. Оперативное информационное состояние назначений $S(\{A_k\})$ создается и поддерживается специализированным серверным программным агентом, названным «агентом назначений» (A-агентом).

4. Любая попытка доступа любого проектировщика к любой задаче дерева задач обнаруживается и регистрируется специализированным программным агентом, действующим на соответствующем клиентском рабочем месте. Такой агент, названный «агентом событий» (E-агент), формирует поток событий, в каждом из которых регистрируется имя проектировщика и имя задачи, к которой осуществляется попытка доступа. Такие события ниже будут обозначаться как T-события.

5. Программный агент, названный «агентом мониторинга» (M-агент), объединяет потоки событий, регистрируя их на «доске объявлений» сервера, с которой будут считываться события для проверок.

6. Программный агент, названный «агентом проверки» (C-агент), действующий на рабочем месте администратора защиты, проверяет на правомерность попытки доступа, зарегистрированные на доске объявлений. В соответствии с результатами проверок C-агент либо открывает доступ для проверенной попытки, либо блокирует его, регистрируя очередную запись в протоколе нарушений.

Описанные особенности и действия в обобщенной форме представлены на рисунке 2, на котором также демонстрируются используемые интерфейсы.

На схеме защиты различаются четыре версии участия проектировщиков в процессе защиты. Руководитель группы проектировщиков имеет право выбрать и назначить (1) ответственного за решение новой проектной задачи или включить в процесс решения помощника. Каждое новое назначение обнаруживается ^-агентом, который вносит соответствующее изменение в копию модели коллектива (в копию части оргструктуры), используемую в решении задачи меточной защиты.

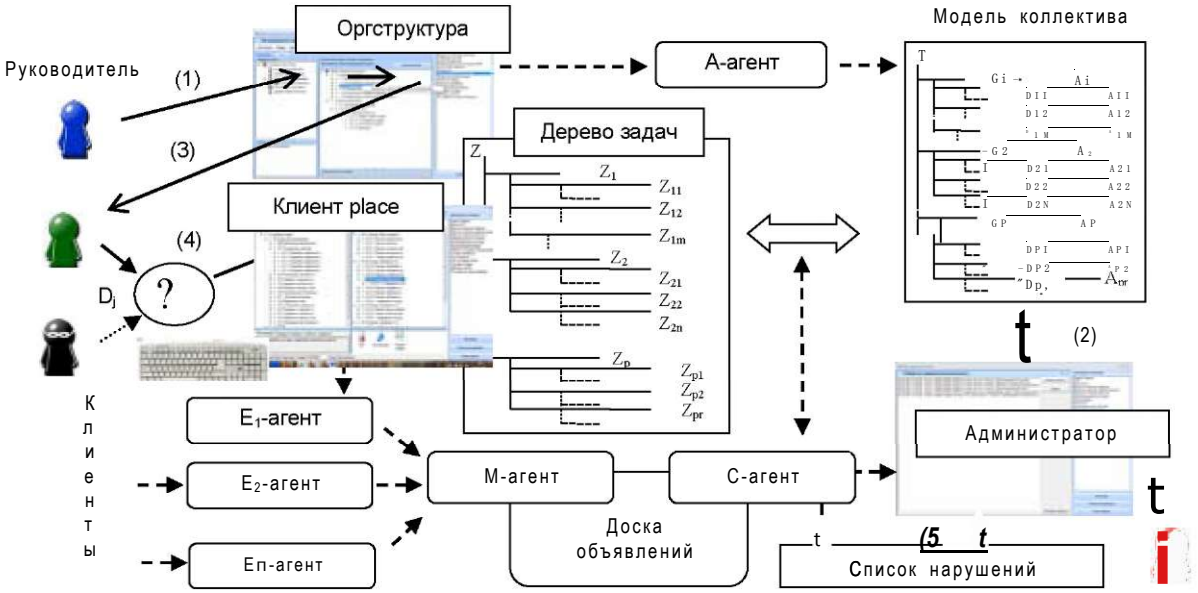

Рис. 2. Обобщенная схема меточной защиты

Каждое новое назначение обрабатывается (2) администратором, ответственным за выделение очередной метки и использование ее для шифрования связи между соответствующей задачей (Z_i) и выбранным членом группы проектировщиков.

После проведения отмеченных назначений (1) и (2) проектировщик D_j может осуществить доступ (4) к задаче Z_i, но это может быть либо в рамках выделенных ему прав, либо с их нарушениями.

Отметим, что, в общем случае, доступ к задаче Z_i может быть активирован случайно или намеренно проектировщиком, у которого нет права доступа к этой задаче. Поэтому любая попытка (4) доступа к любой задаче, а значит и к задаче Z_i должна быть обнаружена. Для такого обнаружения клавишные действия проектировщика (клавиатура, мышь) обрабатываются соответствующим агентом £-типа, который отправляет потенциально опасные события M-агенту, регистрирующему их на доске объявлений сервера. Как уже отмечалось, события, зарегистрированные на доске объявлений, обрабатываются C-агентом, а результаты обработки используются для управления меточной защитой проектных задач.

АГЕНТ НАЗНАЧЕНИЙ

Решение о применении модели оргструктуры было обусловлено, прежде всего, необходимостью разделить информацию о назначениях, которой оперерируют проектировщики, от информации о назначениях, используемой администратором информационной безопасности. Такое разделение обеспечивается A-агентом, который извлекает очередные изменения в базе данных оргструктуры, связанные с назначениями, и загружает их в структуры данных модели коллектива, которые имеют тот же тип (рис. 3), что и представление дерева задач с QA-моделями в QA-базе данных.

В процессе дублирования A-агент обеспечивает в текущий момент времени адекватность описаний как для коллектива, так и для назначений базе данных оргструктуры и QA-базе данных. Дополнительная модель коллектива наследует все возможности, которые предоставлены в среде WIQA для работы с деревом задач и QA-моделями, но этими возможностями пользуется только администратор информационной безопасности в его взаимодействиях с моделью коллектива.

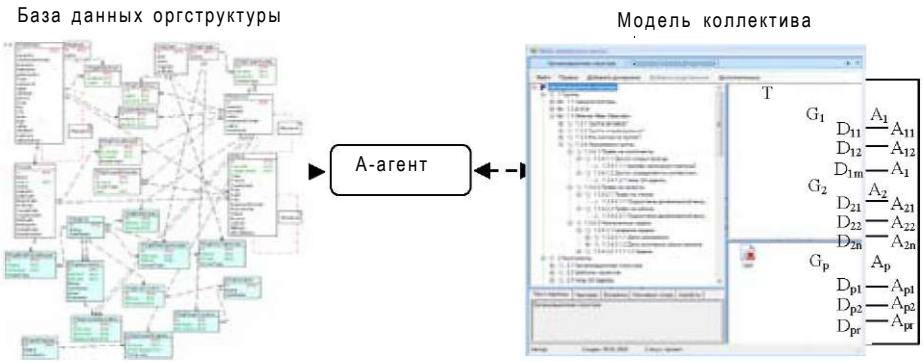

Рис. 3. Дублирование информации о коллективе проектировщиков и назначениях задач

КРИПТОГРАФИЯ МЕТОЧНОЙ ЗАЩИТЫ

Включение меток в отношения защиты между членами группы проектировщиков и проектными задачами является основной функцией администратора безопасности, работа которого заключается в приписывании меток защиты и их шифровании для новых назначений. Содержание этих функций раскрывает схема, представленная на рисунке 4.

Рис. 4. Место меток в защите задач

Предположим, что задача Z_{2_j} назначена с учетом прав и ролей A_{1m} проектировщику D_{1m} и это назначение $A_{1m}(D_{1m}, Z_{2j})$ зарегистрировано ^-агентом в модели коллектива проектировщиков. Сам факт такой загрузки является событием, которое требует от администратора безопасности связать меткой назначение $A_{1m}(D_{1m}, Z_{2j})$ с задачей Z_{2t} в дереве задач. Для этого администратор генерирует новую метку и добавляет ее имя в список меток, каждая из которых связывает проектировщика D_{1m} с другими задачами, назначенными ему ранее. Этот список $R(\{L'_{mn}\})$ включается в соответствующую позицию модели коллектива. Каждая метка в списке $R(\{L'_{mn}\})$ шифруется и включается в список кодов меток, каждый из которых связывает задачу Z_2. с соответствующими назначениями для этой задачи. Список кодов меток $R(\{C(L^2\ ^)\})$ прикрепляется к задаче как дополнительный атрибут (с помощью механизмов дополнительной атрибутики инструментария WIQA). В любой момент администратор может заменить использовавшуюся процедуру шифрования на другую процедуру (планово или если в этом появилась необходимость).

АГЕНТЫ СОБЫТИЙ И МОНИТОРИНГА

Вторым типом оперативных информационных единиц, используемых в меточной защите, являются Т-события, регистрирующие попытки доступа к проектным задачам. В представляемом подходе такие события обнаруживаются в протоколах клавишных операций проектировщиков на каждом рабочем месте. На рисунке 5, демонстрирующем формирование Т-событий, как источник клавишных операций указана клавиатура.

Все визуальные компоненты (дерево задач, QA-модели, модель коллектива) основного интерфейса комплекса WIQA являются наследниками класса Windows. Forms. Control. Этот факт позволяет использовать интерфейсы Control для ссылок на любой визуализируемый объект, используемый в задаче меточной защиты, а значит открывает возможность обнаружения попыток доступа проектировщиков к выбранным проектным задачам.

Каждая клавишная операция, осуществляемая проектировщиком D_j на его рабочем месте, включается в специальный протокол, анализируя который Е-агент обнаруживает очередную попытку доступа к проектным задачам. Имя выбранной задачи (пусть T_k) сравнивается со списком имен задач в профиле доступа, в котором для каждого сеанса работы (от включения до выключения компьютера рабочего места) регистрируются задачи, правомерность доступа к которым в сеансе уже проверялась. Проверка правомерности доступа наследует результат из профиля.

Если имя задачи T_k в профиле отсутствует, то E_j-агент создает и включает выбираемую задачу T_{jk} в поток Т-событий

$$\{.(t^{*D} \quad t^{Z} \ \$_r, K\ )\ \}$$

где t_n - время возникновения Т-события;

D_j - проектировщик, являющийся инициатором события;

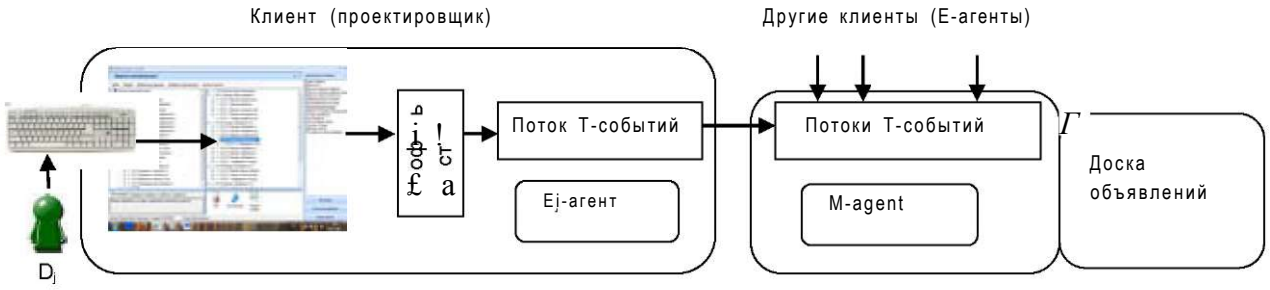

Рис. 5. формирование потоков Т-событий

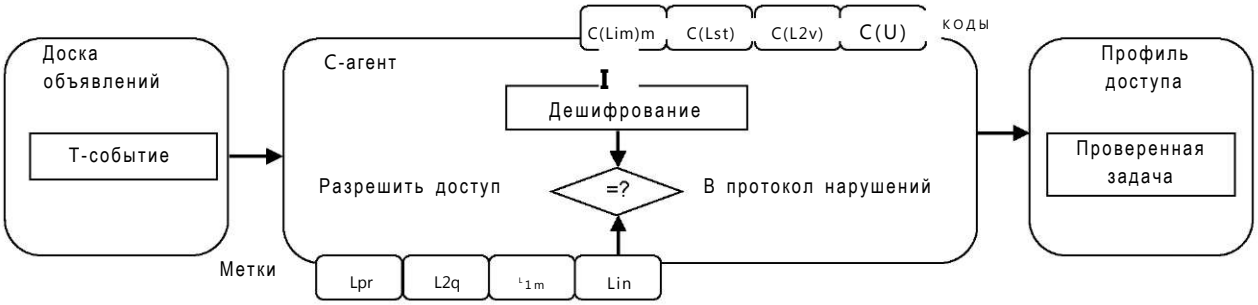

Рис. 6. Обработка Т-события

Z_i - имя выбранной им задачи;

S_r - совокупность подзадач задачи Z_i;

K_q - тип предполагаемых действий с задачей Z_i.

М-агент объединяет сообщения о Т-событиях, поступающих от ^-агентов, в их поток, регистрируемый на доске объявлений сервера WIQA.

АГЕНТ ПРОВЕРКИ ПРАВ ДОСТУПА

Очередное Т-событие с доски объявлений сервера обрабатывается С-агентом, ответственным за своевременную проверку прав для соответствующей попытки доступа. Схема проверки обобщенно представлена на рисунке 6.

Имя задачи Z_j из описания $Л_{jк}$ проверяемой попытки доступа используется С-агентом для извлечения списка кодов меток $R(\{C(I^2_{pl})\})$ для этой задачи, зарегистрированной в дереве задач. Элементы этого списка дешифруются и записываются в список $R(\{L^3_{pq}\})$. Подобным образом, но по идентификатору проектировщика D, этот агент из модели коллектива извлекает список $R(\{L^l$ }). В результате сравнения списков $R(\{L^1_{mn}\})$ и $R(\{I?_{pq}\})$ формируется ответ о правомерности попытки доступа, информация о которой используется оперативно, а также регистрируется в профиле доступа соответствующего рабочего места.

ЗАКЛЮЧЕНИЕ

Представленные средства меточной защиты прошли экспериментальную проверку на системе задач проектного документирования. Описанная в статье схема защиты действует надежно. Опасность в схеме представляет использование проектировщиками и администратором информационной безопасности общей QA-базы данных. По этой причине принято решение реализовать функционал меточной защиты как независимое приложение, также созданное на базе инструментальной среды WIQA. Для осуществления такого решения достаточно перепрограммировать коды А-агента. Намечены два направления развития комплекса средств меточной защиты, одним из которых будет расширение защиты на инструментальные средства, предоставляемые проектировщикам. Другое направление связано с меточной защитой активов, модели которых размещены в репозитории базы опыта проектной организации.

СПИСОК ЛИТЕРАТУРЫ

1. Booch, G.and Brown, A.W. "Collaborative Development Environments," http: // www. booch. com/architecture/ blog/artifacts/CDE.pdf. - 2004.

2. Kroll, P. and Ph. Kruchten, Ph. "The Rational Unified Process Made Easy: A Practitioners Guide to the RUP," Addison-Wesley. - 2003.

3. ISO/IEC 15408. "Common Criteria for Information Technology Security Evaluation Part 1~3 Version 3.0". - 2005.

4. Owens, B. D., Herring, M. S., Dulac, N., Leveson, N. G., Ingham, M. D. and Weiss, K. A. "Application of a safety-driven design methodology to an outer planet exploration mission". In Proc. IEEE Aerospace Conference. - 2008.

5. Ramirez Caceres, G. H. and Teshigawara Y. "Study on a Threat-Countermeasure Model Based on International Standard Information". The 12th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI 2008), Orlando, Florida, USA. - 2008. - pp. 227-232.

6. Sosnin P. "Means of question-answer interaction for collaborative development activity" // Hindawi Publishing Corporation,Advances in Human-Computer Interaction, Volume 2009, Article ID 619405. - 2009. -18 pages.

7. Stringfellow, M.V., Leveson, N.G. and Owens. B.D. "Safety-Driven Design for Software-Intensive Aerospace and Automotive Systems". Proceedings of the IEEE vol. 98, Issue 4. - 2010. -pp. 515-525.

8. Zheng, L. and Myers A. C. "Dynamic Security Labels and Static Information Flow Control". International Journal of Information Security Volume 6, Issue 2. - 2007.

9. Zhukov S.V., Sosnin P.I. Approach to Materializing the Metrics of Information Technology Security// The 8th International conference on Interactive Systems and Technologies: the Problems of Human Computer Interaction.- Collection of scientific papers. - Ulyanovsk: ULSTU. - 2009. -pp. 111-116.