УДК 004
ББК 32.81

ПРИМЕНЕНИЕ МНОГОАГЕНТНОГО ПОДХОДА ДЛЯ ПОЛУНАТУРНОГО МОДЕЛИРОВАНИЯ ЗЛОУМЫШЛЕННЫХ ВОЗДЕЙСТВИЙ

М.Ю. Умницын

Предлагается подход к анализу защищенности информационных систем от злоумышленных воздействий, основанный на применении многоагентных систем с предварительным моделированием сценариев поведения злоумышленника посредством формальной верификации моделей.

Ключевые слова: *информационная система, имитационное моделирование, многоагентная система, JASON, верификация моделей, сценарии злоумышленника.*

Современные информационные системы обладают высокой структурной сложностью. Наличие уязвимостей в этих системах, а также вариативность и сложность злоумышленных воздействий вкупе с серьезностью последствий их успешной реализации обусловливают важность разработки методов и средств, позволяющих оценивать защищенность от таких воздействий.

Основная часть проблем, возникающих при проведении оценок, вызвана недостаточной проработанностью подходов к построению устойчивых моделей интегрированных адаптивных систем защиты, функционирующих в распределенных средах.

Предлагается подход к анализу защищенности информационных систем от злоумышленных воздействий, основанный на применении многоагентных систем с предварительным моделированием сценариев поведения злоумышленника посредством формальной верификации моделей.

Для построения сценариев проводится предварительное моделирование поведения информационной системы. Модель ИС представляется тройкой объектов:

$$M_{\text{ИС}} = \langle Obj, Vul, Res \rangle, \qquad (1)$$

где $Obj = \{obj_i\}$ – множество объектов информационной системы;
$Vul = \{vul_i\}$ – множество уязвимостей;
$Res = \{res_i\}$ – множество ресурсов ИС, найденных при исследовании информационной системы.

Множество объектов ИС представляется следующим образом:

$$Obj = \langle Host, Component \rangle, \qquad (2)$$

где $Host = \{host_i\}$ – множество хостов ИС;
$Component = \{component_j\}$ – множество компонентов ИС, функционирующих в ней.

Каждый компонент в терминах модели Крипке [1] представляется как

$$component_j = (S, S_0, R, L). \qquad (3)$$

Хост определяется как объединение компонент с учетом их взаимосвязей.

Уязвимости представляются как:

$$vul_i = \langle id, desc, action, SF, PF, V_{exploit} \rangle \qquad (4)$$

где id – идентификатор уязвимости;

$desc$ – описание уязвимости;
$action$ – правила эксплуатации уязвимости;
SF – множество предусловий;
PF – множество постусловий;
$V_{exploit}$ – сложность эксплуатации уязвимости.

Процесс эксплуатации уязвимости выражается как перевод компонентов информационной системы из одного состояния в другое. Множество уязвимостей формирует базу данных уязвимостей.

Под ресурсами *Res* понимается циркулирующая в ИС информация. Ресурсы непосредственно не моделируются – они представляются как множество состояний, в которых производится несанкционированное воздействие по отношению к информации, циркулирующей в информационной системе: базам данных, документам, правам доступа, настройкам служб.

Ресурс определяется как:

$$res_i = \langle id, S', C, R_Д \rangle, \quad (5)$$

где id – идентификатор (название) ресурса;
S' – подмножество состояний, в которых нарушитель осуществляет доступ к ресурсу;
C – стоимость ресурса (количественная оценка ущерба), в случае несанкционированного доступа к ресурсу;
$R_Д$ – допустимый риск.

Сценарий злоумышленника представляется в виде четверки:

$$tr_i = \langle id, S, R, P_{у.сц} \rangle, \quad (6)$$

где id – идентификатор сценария;
S – множество состояний информационной системы в сценарии;
R – множество дуг между состояниями;
$P_{у.сц}$ – вероятность выбора данного сценария.

Множество сценариев $Tr = \{tr_i\}$ – результат верификации модели информационной системы $M_{ИС}$ при заданных целях злоумышленника. Цели злоумышленника определяются в терминах темпоральных логик CTL или LTL [1].

Множество сценариев именуется библиотекой сценариев и хранится в виде XML-файла. Библиотека сценариев легко интерпретируется в виде графа (см. рис. 1).

Для формализации модели ИС и целей злоумышленников, а также последующей ее верификации и построения сценариев используется средство верификации моделей NuSMV. Считается, что все сценарии независимые.

В качестве многоагентной среды для имитационного моделирования выбрана среда Jason, реализующего модель Belief-Desire-Intention (BDI). Основным языком реализации агентов является AgentSpeak, наследующий и развивающий программный каркас Procedural Reasoning System (PRS) [2; 3].

Достоинством данной среды является то, что агенты могут функционировать в любом окружении (Environment). Данное свойство реализуется одноименным суперклассом. Оно позволяет агенту взаимодействовать как с виртуальным окружением (функционирующим в рамках среды Jason), так и с реальным компонентом (посредством Java API).

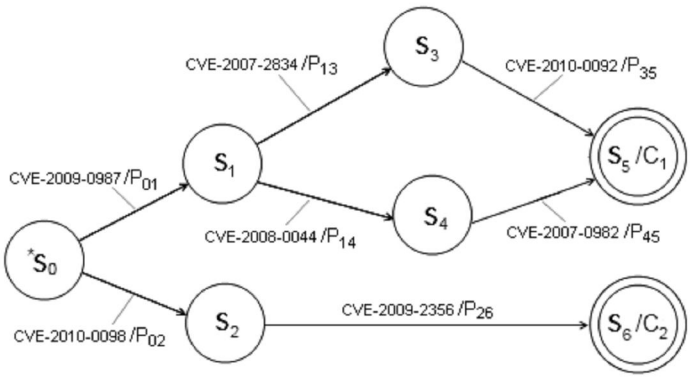

Рис. 1. Граф библиотеки сценариев агентов-злоумышленников

Для имитации злоумышленных воздействий, соответствующих заданной библиотеке сценариев, предлагается использовать два типа агентов:

- $\{ag^{ps_{strategy}}\}$ – множество агентов, которым известны сценарии воздействий на информационную систему (то есть последовательность посещаемых вершин графа, стратегия); они выполняют мониторинг состояния информационной системы, определяют, какой вершине соответствует текущее состояние и отдают приказы агентам второго множества на эксплуатацию уязвимостей для перевода системы в состояние, соответствующее одной из следующих вершин графа;
- $\{ag^{ps_{tactics}}\}$ – множество агентов, которым известны тактики эксплуатации уязвимостей (то есть последовательность действий, приводящих к переходу в следующую вершину графа), но представления о стратегии злоумышленного воздействия на информационную систему они не имеют.

Для управления ходом имитации, а также для оценки результатов злоумышленных воздействий в многоагентную систему введен специальный мастер-агент, обладающий сведениями о рисках, связанных с каждым из сценариев.

Архитектура многоагентного комплекса представлена на рисунке 2.

В составе многоагентной системы выделяются следующие компоненты:

- Коммутатор, к которому подключаются агенты-злоумышленники, входящие в состав модуля имитации злоумышленных воздействий, а также модуль виртуальных компонентов и шлюз связи с информационной системой. Блок коммутации, используя информацию, расположенную в таблице коммутации, выполняет пересылку сообщений между портами коммутатора.
- Модуль виртуальных компонентов информационной системы – является основой для обеспечения полунатурного подхода при имитации злоумышленных воздействий. Он содержит модели тех элементов информационной системы, на которых в процессе имитации могут выполняться деструктивные, разрушающие воздействия.

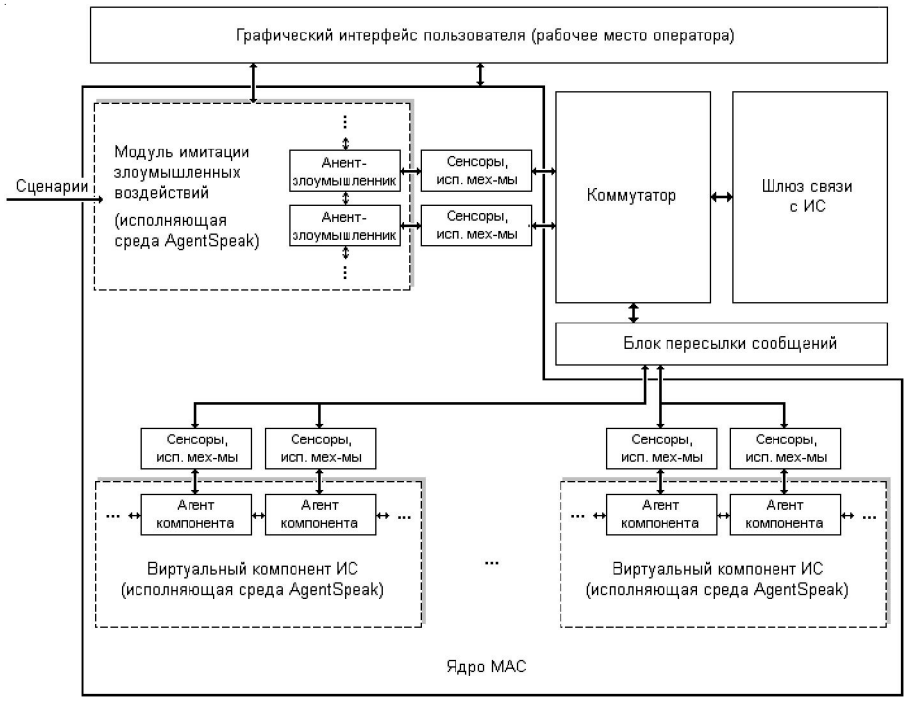

Рис. 2. Архитектура многоагентной системы

- Шлюз связи. Он предназначен для пересылки сообщений, идущих от агентов-злоумышленников к информационной системе и обратно: модуль играет роль моста, связующего многоагентную систему с информационной системой.

Одно из достоинств выбранного многоагентного средства состоит в том, что агенты могут работать в двух режимах: в режиме имитации взаимодействия с ИС, в режиме непосредственного взаимодействия с ИС. В рамках проверки модели для полунатурного режима были построены автоматы, описывающие процессы функционирования системы обнаружения вторжений. При наличии в информационной системе виртуального средства защиты агентам-злоумышленникам не удалось получить доступа к требуемым ресурсам.

Однако использование данного подхода, при всех его достоинствах, сопряжено с рядом сложностей:

1. Для получения более качественных сценариев необходимо использовать более детальные модели компонентов ИС, что повышает сложность исследования ИС. Тем не менее для построения сценариев можно попробовать использовать другие подходы [4].

2. Качество сценариев напрямую зависит от качества разработанных моделей компонентов ИС.

3. Необходимость серьезной модернизации многоагентного комплекса под каждую ИС, что связано с большим числом комбинаций существующих уязвимостей. Единственный выход – сформировать базу данных типовых сценариев.

СПИСОК ЛИТЕРАТУРЫ

1. Кларк, Э. М. Верификация моделей программ. Model Checking / Э. М. Кларк, О. Грамберг, Д. Пелед. – М. : Изд. дом «МЦНМО», 2002. – 416 с.

2. Bordini, R. H. Programming multi-agent systems in AgentSpeak using Jason / R. H. Bordini, F. J. Hubner, M. Wooldridge. – Wiley, 2007. – 294 p.

3. Georgeff, M. The Belief-Desire-Intention model of agency / M. Georgeff, B. Pell, M. Pollack [et al.] // Materials of Cognitive Modeling and Multi-Agent Interactions Conference. – CRC Press, 2005.

4. Heberlein, T. Attack Graphs. Identifying Critical Vulnerabilities Within An Organization. 2004 / T. Heberlein, M. Danforth, T. Stallard. – Mode of access: http://seclab.cs.ucdavis.edu/seminars/AttackGraphs.pdf (date of access: 10.06.2010).

APPLICATION OF MULTIAGENT APPROACH FOR SCALED-DOWN SIMULATION OF MALICIOUS IMPACT

M.Yu. Umnitsyn

Describes the approach to the analysis of security of information systems from the malicious impact, based on application multiagent systems with preliminary modeling of scenarios of behavior of the intruder with model checking approaches.

Key words: *information systems, simulation, multiagent systems, JASON, model checking, malicious scenarios.*