УДК 004.056, 004.838.2
ВАК 05.13.00
РИНЦ 20.15.00

О.В. Лукинова

Компьютерное формирование целей и стратегий нарушителя безопасности информационной системы

Показана связь стратегических, тактических целей и стратегий, реализуемых нарушителем безопасности информационной системы. Разработаны семантические модели и алгоритмы для вывода целей и стратегий, описаны критерии и задачи для оценки параметров стратегий, основанные на представлении информационной системы в виде модели открытой среды.

***Ключевые слова**: цели нарушителя, стратегии нападения, потенциал нарушителя, оценочные критерии.*

COMPUTER TARGET AND STRATEGY FORMATION OF THE INFORMATION SYSTEM SAFETY VIOLATOR

The article deals with communication of strategic, tactical targets and the strategy realized by the violator of safety of information system. Semantic models and algorithms are developed for a conclusion of the purposes and strategies; criteria and tasks for strategies assessment, based on representation of information system in the form of the open environment model are described.

***Keywords:** purposes of the violator, attack strategy, potential of the violator, estimated criteria.*

Введение

Сложность и противоречивость решений, которые приходится принимать при управлении безопасностью информационных систем (ИС), требуют применения компьютерных систем, поддерживающих этот процесс. Такие системы, назовем их системами поддержки управления безопасностью (СПУБ), в своем составе должны иметь три необходимые компоненты: система поддержки принятия решений при планировании защиты ИС (СППР), система управления (СУ) безопасностью на этапе эксплуатации средств защиты (СЗ) и система мониторинга и анализа обстановки.

В рамках процесса управления, реализуемого СПУБ, возникает потребность прогнозирования списка целей нарушителя, перечня возможных стратегий их реализации (атак), а также моделирования развития атаки. Необходимость решения перечисленных задач вызвана двумя причинами: во-первых, возможностью проанализировать в рамках компьютерного сценария, насколько защитные средства будут противостоять возможным атакам; во-вторых, в случае наступления реальной атаки, знание о целях злоумышленника позволит выбрать более адекватные оперативные меры противодействия.

Рассматриваемая предметная область носит достаточно субъективный характер, поэтому для моделирования указанных понятий используются семантические формализмы и методы экспертных оценок. Кроме того, разработанные алгоритмы основываются на представлении объекта нападения, которым является ИС, в виде модели открытой среды [1].

1. Прогнозирование целей и стратегий нарушителя при проектировании системы защиты

Объектом негативных устремлений злоумышленника является информационная система. В [2] имеется подробное описание структурного представления ИС в виде модели открытой среды *POSIX OSE/RM* (Open System Environment/ Reference Model), включающего приложение, как средство реализации бизнес-процесса предприятия и платформу, обеспечивающую работу приложения своими услугами. Трехмерность модели позволяет структурировать не только функциональность самой ИС (плоскость <ИС>), но и систем администрирования и защиты (плоскости <А> и <З> соответственно).

При этом задача безопасного функционирования ИС ставится как задача обеспечения основных (хотя могут быть рассмотрены и другие) свойств: конфиденциальности (K), целостности (C), доступности (D) (критериев безопасности $KS^{цель}(K, C, D)$).

Цели, которые ставит перед собой нарушитель, планируя атаку на ИС, могут быть различны. Это может быть, например:

– обрушение какого-либо вида деятельности предприятия (например, электронного магазина); это значит, что бизнес-процессы (и со-

Ольга Васильевна Лукинова,
к.т.н., с.н.с.
Тел.: (495) 334-89-70
Эл. почта: lobars@mail.ru
Институт проблем управления им. В.А.Трапезникова РАН
www.ipu.ru

Olga V. Lukinova,
candidate of technical Sciences, senior research associate,
Tel.: (495) 334-89-70
E-mail: lobars@mail.ru
The Institute of Control Sciences of the Russian Academy of Sciences (ICS RAS).
www.ipu.ru

ответствующие приложения ИС), моделирующие эту деятельность, находятся в зоне риска;

– кража или модификация каких-либо данных;

– обрушение ОС или ее подсистем;

– взлом механизмов системы защиты;

– использование сервера для организации *DDOS*-атак («зомби», Smarf-усилитель);

– желание продемонстрировать свое умение, амбиции и т.п.

Обозначим G – множество целей нарушителя, при этом все множество целей G включает как стратегические цели StG_i, $i = 1,\ldots, I$, так и тактические TkG_s, $s = 1,\ldots, S$.

Цели StG_i всегда предполагают нанесение *прямого* вреда функционированию бизнес-процессов (приложения ИС), в то время как реализация тактических целей нарушает безопасность бизнес-процесса опосредованно. Так или иначе, все присущие злоумышленнику цели, направлены на нарушение критериев $\overline{KS^{цель}}(K, C, D)$ в «клетках» плоскостей <ИС>, <А>, <З> модели *OSE/RM* (рис. 1).

Чтобы нарушителю осуществить любую из целей StG_i, ему необходимо решить ряд задач, т.е. осуществить одну или несколько тактических целей TkG_s, $s = 1, .., n$. Например:

1. Получить возможность входа в локальный узел посредством: кражи пароля ОС (плоскость <З>), учетной записи пользователя (<А>), использования уязвимости программного обеспечения плоскостей <ИС>, <А>, <З> и т.п.

2. Получить возможность входа в сетевой узел посредством: кражи паролей ОС или сетевых, IP-адресов, использования незащищенных модемов или открытых портов и т.п.

3. Осуществить контроль над узлом, осуществляя модификации ОС или ядра, сокрытия файлов, процессов, сети, организации черных ходов и т.п.

4. Сокрытия следов присутствия в узле и т.п.

5. Реализовать деструктивные воздействия, нанеся вред напрямую бизнес-процессу (плоскость <ИС>), системе администрирования ОС (<А>), системе защиты (<З>).

Алгоритмы, моделирующие взаимосвязи стратегических и тактических целей, описаны в [3].

Описание алгоритма вывода списка стратегий нападения

Решая тактические задачи, нарушитель в действительности осуществляет последовательность действий, которые специалисты называют атакой. Мы такие действия будем квалифицировать как *стратегию нападения*. Поэтому СППР должна иметь в своем составе алгоритмы, которые позволят сформировать множество возможных атак в зависимости от ориентиров политики безопасности и предпочтений ЛПР.

На рис. 2 представлена концептуальная схема семантической модели для логического вывода множества стратегий нападения $\{Str^s\}$, соответствующих s-й тактической цели TkG_s^i и i-й стратегической StG_i. Список формируется при следующих входных данных: возможностей нарушителя *NP*, канал атаки КА, уязвимости клетки X^{KS}. Указанное множество формируется для каждой стратегической цели StG_i.

Собственно алгоритм формирования списка атак реализован в виде блока логического вывода продукционной системы, на множестве правил вида:

if (<*посылка*>) *then* <*действие*>.

Здесь <*посылка*> и <*действие*> строятся с использованием входных и выходных концептов, представленных на рис. 2. При этом утверждения в <посылке> и <действии> представляются в виде пар «атрибут – значение», если знания достоверные; если же в рассуждениях присутствуют неопределенность, неточность, нечеткость (в теории искусственного интеллекта их называют НЕ-факторами) – тройкой «атрибут – значение – коэффициенты НЕ-факторов» [4]. Примеры указанных продукций для достоверных утверждений:

if (NP = 'аутсайдер') then КА = 'визуальный'

if (КА = 'визуальный' & NP = 'аутсайдер')

then ОА = 'экранные формы'

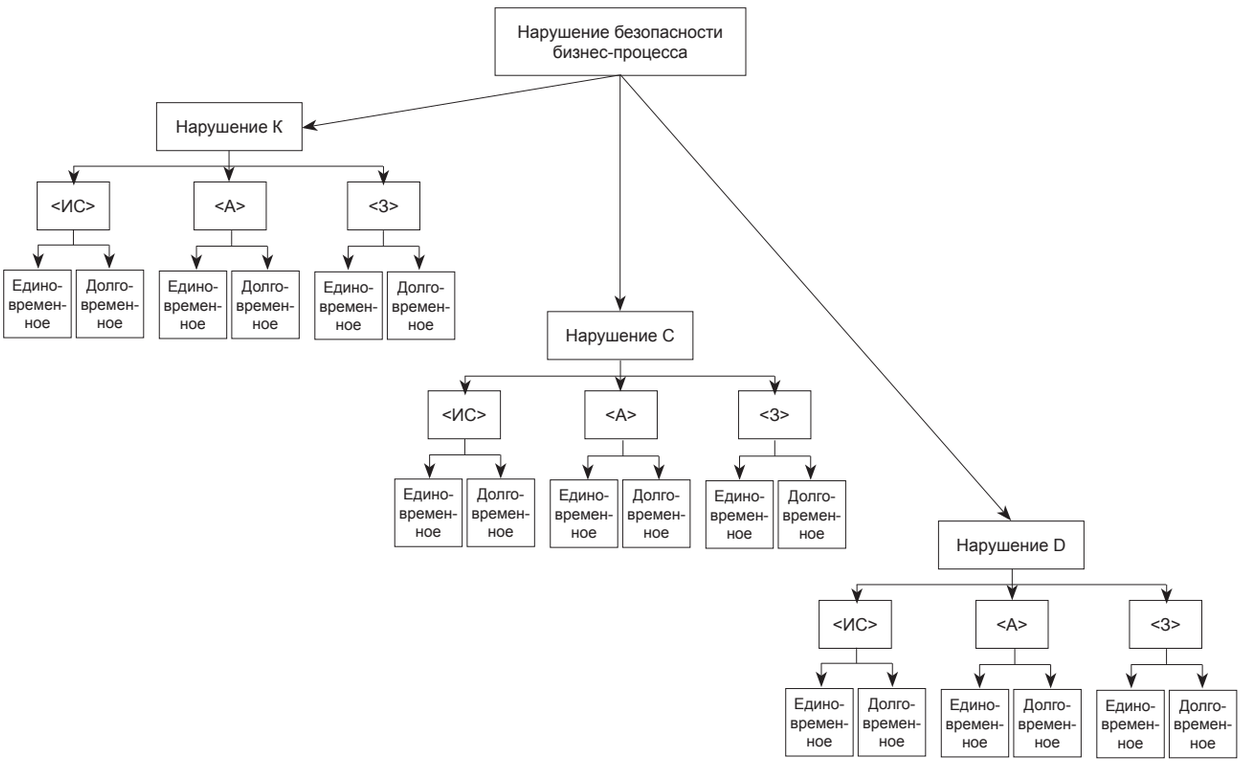


Рис. 1. Дерево стратегических целей нарушителя

if (StG = 'бизнес-процесс ' & OA = 'экранные формы') (*)

then TkG='кража данных '

if (TkG='кража данных ' & OA =' экранные формы')

then Str='просмотр экрана компьютера через окно помещения'

Таким образом, общий алгоритм формирования списка актуальных атак (стратегий) заключается в следующем:

1. СППР выводит на экран дерево стратегических целей, чтобы эксперты выбрали те цели, которые им кажутся актуальными, и проставили оценки степени актуальности по 4-балльной шкале: «неактуальна» – 1, «малоактуальна» – 2, «актуальна» – 3, «весьма актуальна» – 4.

2. СППР проводит процедуру согласования актуальных целей, выбранных экспертами и оценок актуальности по известным алгоритмам [5].

3. СППР выбирает из БД Модели угроз данные, требуемые для алгоритма вывода, или предлагает ввести недостающие:

– возможности нарушителя *NP = = (тип, используемые средства, время действия, характер знаний, место действия, степень информированности);*

– возможные каналы проникновения *KA = (носитель информации, физическая среда, канал связи)*;

– уязвимости $\tilde{X}^{KS}$ = *(стадии проектирования, стадии эксплуатации)* и их детализация.

4. Аналогично п.п. 1, 2 СППР проводит процедуры выбора и

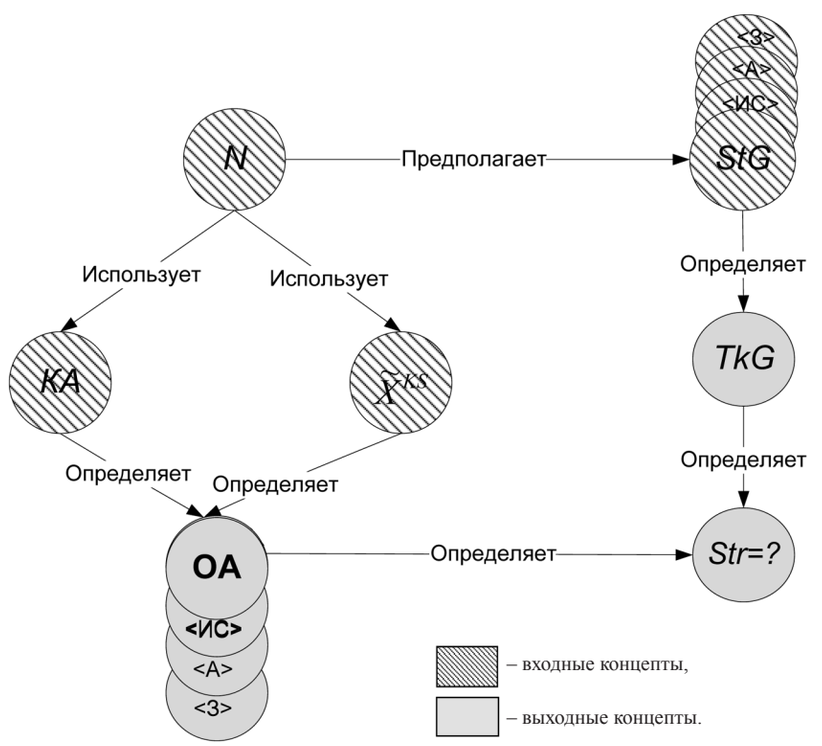
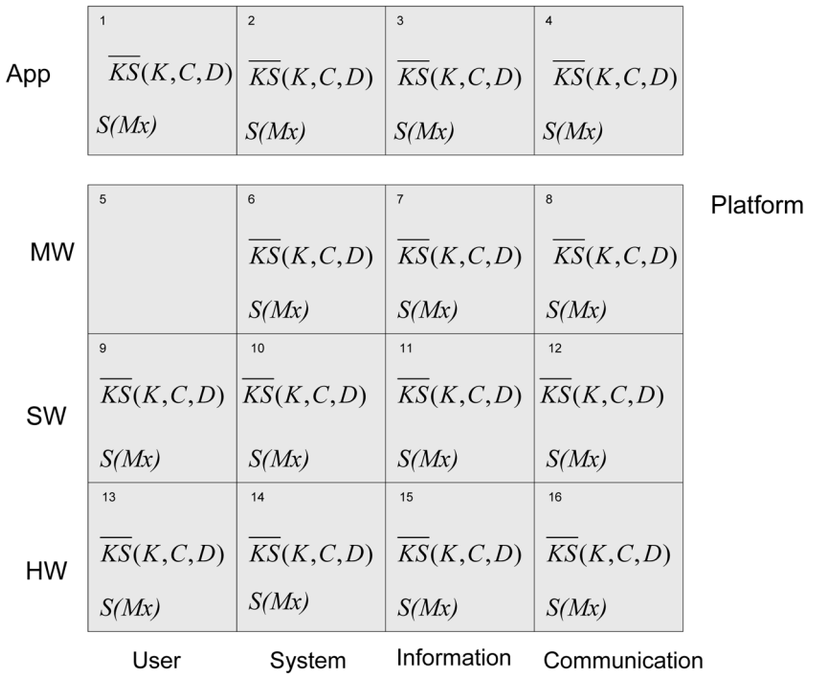


Рис. 2. Схема алгоритма прогнозирования потенциально опасных атак при проектировании



Здесь 1, 2, ..., 16 – номера «клеток».

Рис. 3. Распределение критериев безопасности и механизмов по «клеткам» OSE/RM

согласования тактических целей $\{TkG_s^i\}$, но с учетом их взаимосвязей со стратегическими $\{StG_i\}$.

5. СППР запускает блок логического вывода стратегий нападения (атак), реализующий правила типа (*), для каждой стратегической цели из списка, сформированного в п.п. 1–3.

6. СППР выводит полученный список атак на экран для просмотра экспертами. Если они согласны, то утверждают список. Если нет, система дает возможность поступить двумя способами:

– модифицировать список атак вручную и затем согласовать;

– изменить исходные данные и запустить блок логического вывода снова.

2. Моделирование развития атаки

Каждая атака может быть проведена несколькими способами, поэтому в СППР должны быть реализованы формализованные методы компьютерного моделирования различных способов осуществления стратегий $\{Str^s\}$ с учетом особенностей конкретной ИС. Для этого используется представление ИС в виде модели открытой среды *OSE\RM*. Имеется в виду, что стратегии нападения на информационную систему также можно смоделировать в виде последовательности (цепочки) «клеток» модели (разумеется, речь идет об атаках, производимых с помощью вычислительных программно-аппаратных средств и в среде данной информационной системы).

Пусть $\{Str^s\} = \{Str_1^s, Str_{2,...,}^s Str_L^s\}$ – множество возможных стратегий, соответствующих тактической цели TkG_s^i. Достичь цель TkG_s^i в рамках Str_1^s означает, что нарушитель должен осуществить некоторые действия в определенных «клетках», направленные на нарушение критериев безопасности $KS^{цель}(K,C,D)$, т.е. преодоление *Mx*, установленных в «клетке» (рис. 3). Назовем последовательность таких «клеток» цепочкой реализации A_k. Таким образом, каждая цепочка представима кортежем $A_k(a_1, a_2, \ldots, a_H)$, где a_h, $h = 1 \div H$ – номер «клетки», входящей в цепочку.

В свою очередь, каждую стратегию Str_l^s нарушитель может осуществить несколькими способами, т.е. $\forall Str_j$ можно поставить в соответствие одну или несколько цепочек A_k: $\forall Str_l^s \rightarrow (A_1, A_2,\ldots, A_K)$. Например, пусть стратегия Str_3^s может быть реализована цепочками A_1 или A_2, в составы которых входят «клетки» 16, 12, 8, 4, 3, 11,15 и 13, 10, 11, 15, т.е. $StG_3 = (A_1/A_2) = (16, 12, 8, 4, 3, 11, 15/13, 10, 11, 15)$. В [6] показана принципиальная возможность такого представления стратегии на модель *OSE/RM*.

Далее возникает задача автоматической генерации цепочек A_k, т.е. номеров включенных в цепочку «клеток». Для этого в СППР могут быть задействованы алгоритмы анализирующих грамматик $G = \{T, V, N, P, S, F\}$, где T – множество терминальных символов, т.е. номеров «клеток» 1÷16, V – множество переменных грамматик. Начальный символ N определяется номером «клетки», соответствующей каналу проникновения (*KA*) в систему. Это может быть визуальный – через экранные формы (номер1), физический – через устройства ввода/вывода (номер 13), для сетевых – номер «клетки» a_p, т.е. вывод цепочки может начаться с $N = (a_h = 1 / a_h = a_p / a_h = 13)$.

Далее на основании опроса экспертов СППР должна сформировать следующие матрицы:

– матрицы переходов $F = \|f_{ij}\|$, $i, j = 1 \div 16$, где

$f_{ij} = 0$, если переход из i-й «клетки» в j-ю невозможен,

$f_{ij} = 1$, если такой переход возможен;

– матрицу стратегий $S = \|s_{il}\|$, $i = 1 \div 16$, которая содержит экспертные оценки возможности использования i-й «клетки» при реализации l-й стратегии, где:

$s_{il} = 0$, если «клетка» в стратегии не может быть использована,

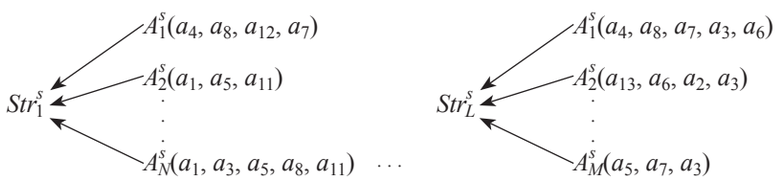


Рис. 4. Примеры соответствий цепочек A_k и стратегий Str_l^s

s_{il} равна некоторому числу в противном случае.

Правила вывода определяют формирование цепочек A_k, из «клеток» с $s_{il} \neq 0$ в соответствии с матрицами F.

В результате БД СППР пополнится соответствиями вида (рис. 4):

Совокупность $\{Str^s\} = \{Str^s_1, Str^s_2, ..., Str^s_L\}$, *в которой* $\forall Str^s_l \rightarrow (A_1, A_2, \ldots, A_K)$ *назовем моделью атак (МА), или моделью стратегий*.

Описанные алгоритмы генерации развития атак могут быть использованы и на этапе разработки системы защиты, и при ее эксплуатации. Правда, модели стратегий $\{Str^s\} = \{Str^s_1, Str^s_2, ..., Str^s_L\}$, сформированные в результате, будут несколько отличаться, так как при проектировании – это прогнозный вариант, а при эксплуатации – произошедший реально, характеристические параметры которого (например, номер «клетки») зафиксированы системой мониторинга.

Применение модели стратегий

Такая модель атак позволит решать задачи прогнозного и оперативного характера, задействованные в контуре управления СПУБ и описанные в этом разделе. Для решения этих задач возникает необходимость формирования оценок уязвимостей программно-аппаратного обеспечения по определенным критериям.

Можно выделить 3 класса критериев: к первому классу отнесем частоту использования нарушителем некоторой уязвимости $\tilde{x}_i$. Как правило, на предприятиях такая статистика ведется, обозначим такую оценку $R^N(\tilde{x}_i)$. Методика формирования такой оценки описана в [7].

Ко второму классу отнесем критерии, характеризующие влияние уязвимости $\tilde{x}_i$ на факт обрушения бизнес-процесса. Обозначим такую оценку $R^K(\tilde{x}_i)$. Для ее формирования СППР может предложить экспертам следующие критерии (разумеется, сам список также предварительно предлагается системой к согласованию или модификации):

1. Какова степень влияния уязвимости $\tilde{x}_i$ на функционирование реализаций «клеток» модели:
   a. плоскости <ИС>,
   b. плоскости <А>,
   c. плоскости <З>.
2. Каково значение ресурса «клетки» с уязвимостью $\tilde{x}_i$ для функционирования приложения (бизнес-процесса).
3. Каково значение «клетки» с уязвимостью $\tilde{x}_i$ для функционирования:
   a. «клеток» плоскости <ИС>,
   b. «клеток» плоскости <А>,
   c. «клеток» плоскости <З>.

Третий класс критериев будет оказывать влияние на оценку времени $R^T(\tilde{x}_i)$ до того момента, когда бизнес-процесс «рухнет»:

1. Сколько «клеток» модели связано с уязвимостью $\tilde{x}_i$.
2. Какова степень влияния уязвимости $\tilde{x}_i$ на функционирование «реализаций «клеток» модели.

На основании ответов СППР формирует оценки $R^N(\tilde{x}_i)$, $R^K(\tilde{x}_i)$, $R^T(\tilde{x}_i)$ как линейную или мультипликативную свертку, предварительно выявив у экспертов их мнения по поводу значимости критериев, т.е. весовые коэффициенты критериев.

Задачи прогнозирования при разработке

Задача 1. Оценка возможности осуществления стратегии нападения из списка возможных

Каждую стратегию (атаку) Str^s_l в практике ИБ принято оценивать с точки зрения ее осуществления исходя из возможностей нарушителя NP и уязвимостей «клетки» $\tilde{X}^{KS}$. Эти оценки могут и должны быть произведены как с учетом имеющихся объективных данных, так и используя субъективные представления руководителей и экспертов. Это означает, что СППР должна уметь оценивать возможность реализации цепочек A_k для стратегии Str^s_l.

В результате мониторинга имеем распределение параметров по референсной модели, представленное на рис. 5, где каждая «клетка» оценивается рейтингом уязвимостей $R^{клетки}$ и потенциалом нарушителя NP, который может воспользоваться уязвимостями $\tilde{X}^{KS}$ [8], т.е. $\forall a_h$ ставится в соответствие две оценки: рейтинг уязвимостей «клетки» и потенциал нарушителя: $\forall a_h \rightarrow (R^{клетки}, NP)$.

В [7, 8] было показано, что условие успешного нападения за-

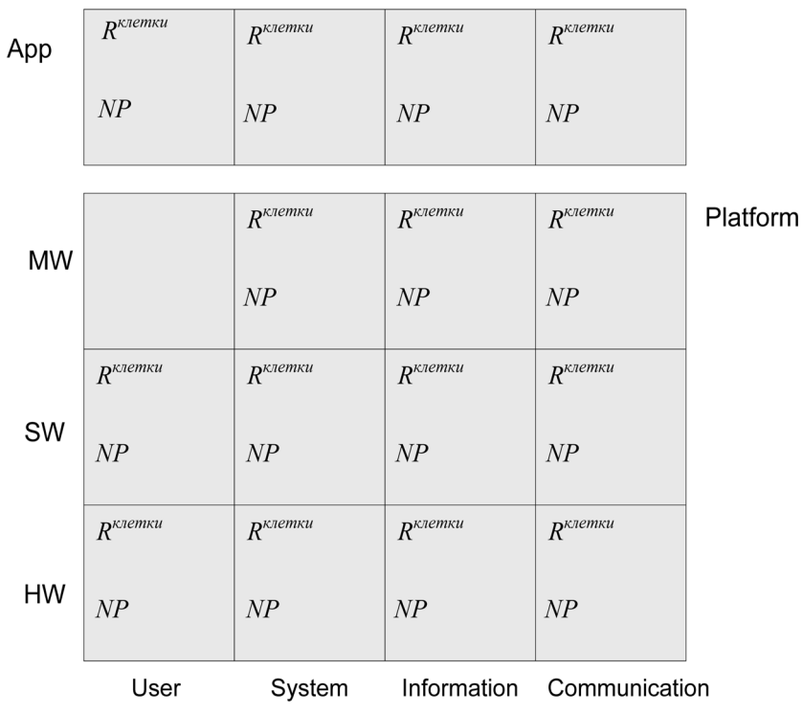


Рис. 5. Распределение потенциала *NP* и рейтинга уязвимостей $R^{клетки}$ по модели *OSE/RM*

ключается в том, что существует $NP = \max_j (NP^j)$ такое, что $R^{клетки} < NP$, где j – типы нарушителя, в нашей терминологии – это потенциал нарушителя, оценочная шкала которого определяется нормативными документами. В качестве рейтинга клетки будем использовать оценку частоты использования уязвимости нарушителем, т.е. $R^{клетки} = R^N(\tilde{x}_i)$.

Обозначим $P(a_h)$ – степень уверенности того, что нарушитель использует уязвимости «клетки» с номером a_h. Тогда степень уверенности в осуществлении нарушителем цепочки $A_k(a_1, a_2, \ldots, a_H)$ (возможность осуществления) $P(A_k)$ можно вычислить следующим образом:

1. Если для $\forall a_h\ R^N(\tilde{x}_i) > NP$, то $P(a_h) = 0$;

2. Если для $\forall a_h\ R^N(\tilde{x}_i) \le NP$, то $P(a_h) = (NP - R^N(\tilde{x}_i))$. Это означает, что если потенциал нарушителя NP больше рейтинга уязвимостей клетки $R^N(\tilde{x}_i)$, то нарушитель сможет воспользоваться уязвимостями со степенью уверенности $P(a_h)$. Нормируем $P(a_h)$ на отрезке [0, 1].

3. Оценку всей цепочки $A_k(a_1, a_2, \ldots, a_H)$ можно осуществлять разными способами, например:

a. $P(A_k) = \sum_{h=1}^{H} P(a_h)$, т.е. $P(A_к)$ – зависит от количество задействованных в цепочке клеток, у которых $NP > R^N(\tilde{x}_i)$;

b. $P(A_k) = \max_h(P(a_h))$, т.е. $P(A_к)$ определяется «клеткой», имеющей максимальное значение оценки.

Таким образом, каждой цепочке $A_k(a_1, a_2, \ldots, a_H)$ можно поставить в соответствие оценку $P(A_к)$, позволяющую судить об уверенности, с которой данная цепочка может быть реализована нарушителем.

Задача 2. Оценка возможности противодействия механизмов защиты цепочке A_k.

На рис. 3 показано, что каждой p-й «клетке» модели OSE/RM ставится в соответствие тот или иной защитный механизм Mx, адекватный требованиям целевых критериев $KS_p^{цель}(K(T^*), C(T^*), D(T^*))$, где T^* – заданное значение критериев. Mx, в свою очередь, характеризуются такой величиной, как

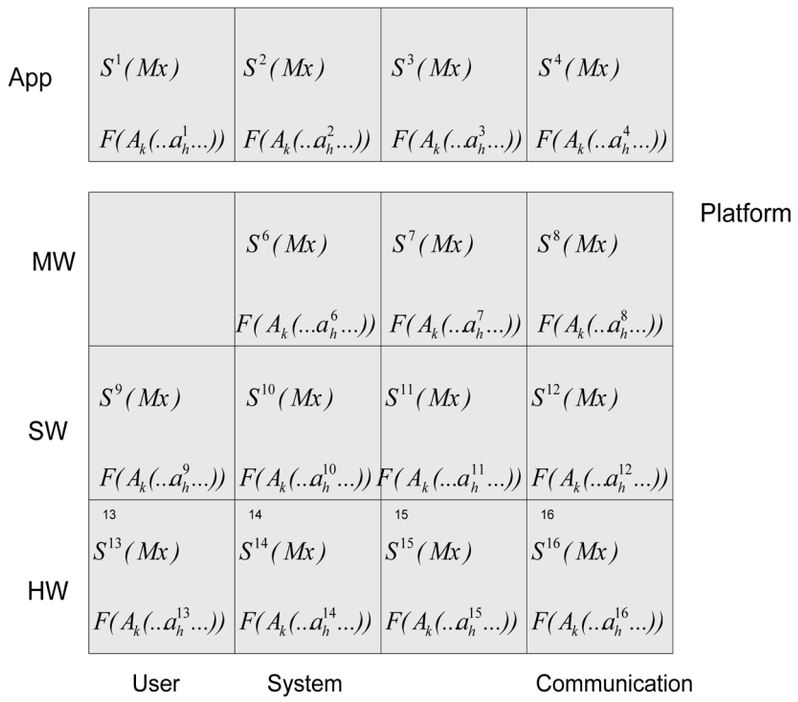


Рис. 6. Распределение параметров стойкости и силы атаки по «клеткам» модели OSE/RM

стойкость $S^p(Mx)$, которая характеризуется временем t, необходимым для взлома Mx, и быстродействием op, т.е. количеством операций, приводящих к взлому механизма.

Обозначим $F(Str_i^s(A_k))$ величину, определяющую силу атаки, т.е. цепочки A_k. Цепочка же представима последовательностью «клеток» *OSE/RM*. Тогда каждая p-я «клетка» характеризуется стойкостью Mx, обеспечивающим уровень критериев $KS_p^{цель}(K, C, D)$, а противостоит им сила атаки $F(Str_i^s(A_k(a_1, a_2, \ldots, a_h^p, \ldots, a_H)))$, рис. 6. Она зависит:

– от оценки возможностей нарушителя, его потенциала;

– мотивация нарушителя.

Чтобы оценить степень противодействия защитного механизма в p-й «клетке», надо оценить, насколько стойкость механизмов $S^p(Mx(t, op))$ выдержит силу атаки $F(Str_i^s(A_k(a_1, a_2, \ldots, a_h^p, \ldots, a_H)))$.

Задачи оперативного реагирования

Задача 3. Контроль «клеток», задействованных в цепочке A_k.

Контроль клеток A_k-й цепочки должен осуществляться в случае, если система мониторинга зафиксировала нарушение хотя бы в одной «клетке» данной цепочки. При этом СППР может поступить двумя способами:

1) выдать предупреждающее сообщение о том, что реализации «клеток», задействованных в данной цепочке, могут быть повреждены и специалисту следует уделить им особое внимание;

2) СППР может сама оценить степень защищенности (опасности, противодействия) «клеток» цепочки и выдать об этом сообщение. Для оценки степени защищенности необходимо использовать алгоритм задачи 2.

Задача 4. Оценка интервала времени до того момента, когда, вследствие реализации A_k-й стратегии нападения, бизнес-процесс «рухнет».

Для этого специальная программа-монитор, назначение которой заключается в контроле прикладного алгоритма, вычисляет момент времени, когда алгоритм приложения посредством системного API-вызова обратится к поврежденной «клетке» платформы. При этом следует обратить особое внимание на обращение к тем «клеткам», у которых значение оценки $R^K(\tilde{x}_i)$ велико, а оценки $R^T(\tilde{x}_i)$ мало.

3. Алгоритм вывода целей нарушителя при эксплуатации системы защиты

Понимание конечных целей нарушителя при нападении необходимо, чтобы грамотно выстроить стратегии защиты. На этапе эксплуатации ситуация отличается тем, что если система мониторинга зафиксировала нападение на ресурсы ИС, то это будет означать, что пострадал какой-нибудь конкретный объект атаки (ОА), ассоциированный с «клеткой» той или иной плоскости модели *OSE/RM*. При этом, если нарушитель воспользовался известной уязвимостью $\tilde{x}_i \in \tilde{X}^{KS}$, то, стало быть, ее оценка при выборе защитных механизмов *Mx* была занижена; если он обнаружил и воспользовался неизвестной уязвимостью $\tilde{z}_i \notin \tilde{X}^{KS}$, то ее надо внести во множество $\tilde{X}^{KS}$ и в дальнейшем перепланировать систему защиты. Аналогично пересматриваются оценки *KA* и возможностей нарушителя *NP*.

Таким образом, при эксплуатации в СППР задействованы 2 процедуры:

1. Первая процедура связана с модификацией моделей нарушителя, уязвимостей и каналов атак (двойные объекты на рис. 7). Покажем алгоритм на примере ситуации с уязвимостями (идеология модификации оценок каналов и возможностей нарушителя реализуются аналогично).

Пусть нарушитель воспользовался некоторыми уязвимостями и осуществил какое-либо из возможных действий $\hat{x}^* \subset \hat{X}$ (здесь и далее * означает совершенное действие, заданный параметр и т.п.). Это означает:

– либо нарушитель реализовал известные уязвимости $(\tilde{x}_1^*, \tilde{x}_2^*, ..., \tilde{x}_j^*) \subset \tilde{X}^{KS}$, $j \le 1, 2, \ldots, p + s + r$, т.е. $\forall \hat{x}_i^l, l = K/C/D, i = 1, 2, \ldots, n + m + k$ можно отобразить во множество уязвимостей $\hat{x}_i^l = f(\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_j)$. Тогда необходимо перейти к более высокой оценке степени опасности для этой уязвимости, например от «опасность средняя» к «опасно», и планировать защиту с тем же вектором уязвимостей $\tilde{X}^{KS}$.

– либо нарушитель обнаружил и воспользовался неизвестными до того уязвимостями $\tilde{z}_i (i = 1, 2, \ldots, n_1)$, $\hat{x}_i^l = f(\tilde{z}_1, \tilde{z}_2, ..., \tilde{z}_{n_1})$. Тогда множество уязвимостей можно дополнить новыми членами $\tilde{X}^{KS} = \tilde{X}^{KS} \cup (\tilde{z}_1, \tilde{z}_2, ..., \tilde{z}_{n_1})$ и осуществлять перепланирование защитных средств с учетом модифицированного вектора уязвимостей.

В результате СППР получает подправленные:

– возможности нарушителя *NP*;

– возможные каналы проникновения *KA*;

– уязвимости $\tilde{X}^{KS}$.

Вторая процедура, которую запускает СППР, касается вывода стратегий и целей нарушителя. В качестве входных данных она использует информацию подсистемы мониторинга, которая зафиксировав нарушение, определила ОА, т.е. «клетку», которая подверглась нападению. Тогда:

1. Если известна «клетка», т.е. ОА, то, воспользовавшись матрицами *F* и *S*, СППР определяет цепочки A^k, в которые входит пораженная «клетка». Пусть это будут цепочки $A_1(a_1, a_6, a_{10})$, $A_2(a_{13}, a_5, a_6, a_{12}, a_7)$, $A_3(a_5, a_{10}, a_6, a_7, a_3)$, а пораженная «клетка» – это «клетка» a_6 (она присутствует в каждой цепочке примера).

2. Тогда ясно, что «клетки», стоящие в цепочках до a_6, также поражены и СППР должна:

a. включить механизмы проверки состояния ресурсов данных «клеток»,

b. оценить степень и причины поражения,

c. начать принимать оперативные или иные меры по ликвидации вторжения,

d. либо выдать предупреждающее сообщение администратору о необходимости проведения проверки состояния ресурсов данных «клеток» и принятия соответствующих мер.

3. «Клетки», стоящие в цепочке после a_6, это путь дальнейшего развития атаки. При этом каждая «клетка» защищена тем или иным *Mx*. СППР запускает алгоритм задачи 2, описанной выше, чтобы оценить степень противодействия установленных *Mx* атаке. В результате каждая цепочка получает оценочное число r_i, характеризующее возможность дальнейшей осуществимости цепочки, т.е. $A_1(r_1)$, $A_2(r_2)$, $A_3(r_3)$.

4. Соответствия цепочек A_k и стратегий Str_l^s (см. рис. 4), хра-

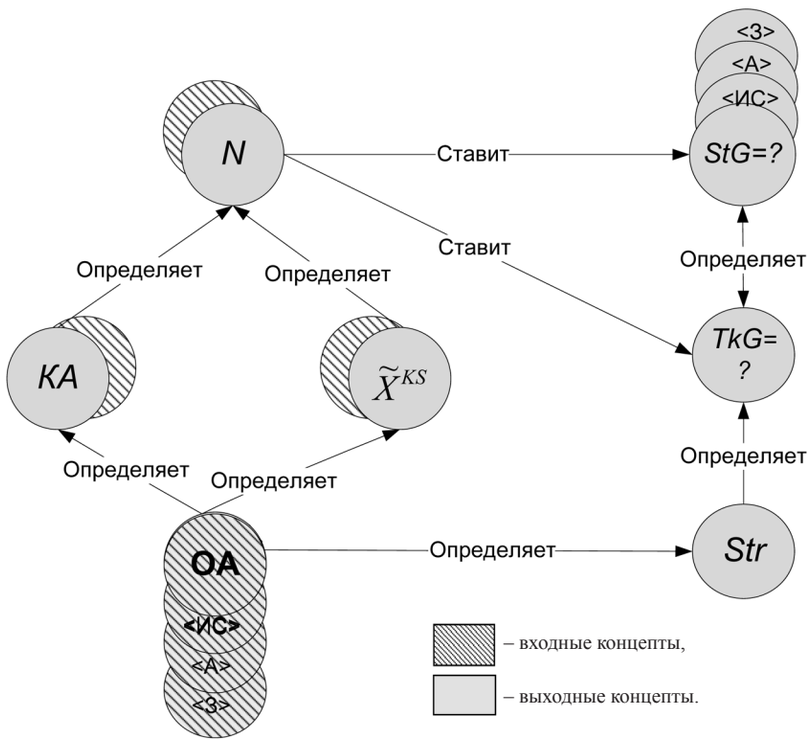


Рис. 7. Схема алгоритма прогнозирования потенциально опасных атак нарушителя при эксплуатации

нящихся в БД, позволяют СППР сделать идентификацию стратегий из списка возможных по цепочкам $A_1(r_1)$, $A_2(r_2)$, $A_3(r_3)$, т.е. номер цепочки получает верхний индекс, например $A_1(r_1) \rightarrow A_1^2(r_1) \rightarrow Str_l^2$, $A_2(r_2) \rightarrow A_2^1(r_2) \rightarrow Str_l^1$, $A_1(r_3) \rightarrow \rightarrow A_3^8(r_3) \rightarrow Str_l^8$.

5. Далее СППР определяет по спрогнозированным стратегиям тактические цели TkG_s^i, затем по тактическим – стратегические StG_i. При этом каждая цель получает оценочное число r_i.

6. СППР выводит на экран список стратегий, тактических и стратегический целей, ранжированных по оценочным числам r_i.

Заключение

В работе представлены алгоритмы, позволяющие осуществить логический вывод перечня атак, которые могут быть реализованы нарушителем, моделировать развитие атаки с привязкой к ИС как объекту нападения. Показаны задачи, которые необходимо решать при наступлении атаки и прогнозирования ее развития с целью оценки степени защищенности ИС при планировании защиты.

Литература

1. ISO/IEC TR 14252-1996 Guide to the POSIX Open Sys¬tem Environment.

2. *Лукинова О.В.* Методология проектирования систем защиты, построенных на основе референсной модели *POSIX OSE/RM* // Системы высокой доступности. – 2012. –№ 3. – С. 38–45.

3. *Лукинова О.В.* Компьютерный мониторинг состояния среды бизнес-процессов при эксплуатации системы защиты // Открытое образование. – 2012. – № 4. – С. 37–47.

4. Нариньяни А.С. НЕ-факторы и инженерия знаний: от наивной формализации к естественной прагматике / А.С. Нариньяни // КИИ-94. Сборник трудов Национальной конференции с международным участием по ИИ. «Искусственный интеллект - 94»: в 2-х т. – Т. 1. – Тверь: АИИ, 1994. – С. 9–18.

5. *Трахтенгерц Э.А., Степин Ю.П.* Методы компьютерной поддержки формирования целей и стратегий в нефтегазовой промышленности. – М.: Синтег, 2007. – 344 с.

6. *Кузнецов В.С., Лукинова О.В.* Представление информационных угроз на основе модели открытой среды // Научно-технический вестник информационных технологий, механики и оптик печати).

7. Common Methodology for Information Technology Security Evaluation. Part 2: Evaluation Methodology. Version 1.0 – CEM 99/045, August, 1999.

8. *Лукинова О.В.* Компьютерные методы мониторинга и анализа защищенности при функционировании автоматизированных бизнес-процессов компании // Открытое образование. – 2011. – № 4. – С. 37–47.