А. В. Аграновский (д. т. н., профессор),
г. Ростов-на-Дону, ФГНУ НИИ «Спецвузавтоматика»
Н. Г. Милославская (к. т. н., доцент)
А. И. Толстой (к. т. н., доцент)
Р. Н. Селин
Московский инженерно-физический институт (государственный университет)

ВЫЯВЛЕНИЕ УГРОЗ БЕЗОПАСНОСТИ КАК СПОСОБ ПРЕДОТВРАЩЕНИЯ АТАК НА КОМПЬЮТЕРНЫЕ СЕТИ

1. Введение

При построении современной системы обнаружения вторжений (СОВ — в нашем исследовании мы будем пользоваться определениями «вторжения», «атаки» и пр. в том виде, как они даны в книге [1. С. 18—20]) необходимо прежде всего сформировать правильные взгляды на информационные процессы, проходящие не только в компьютерной сети, но и во всей информационной системе (ИС). СОВ по сути является специализированной системой обработки информации, предназначенной для чрезвычайно быстрого анализа огромного объема данных совершенно разного вида. Для того чтобы определить наиболее точно критерии эффективности такой системы и оценить параметры, которые наиболее сильно влияют на скорость и точность работы, необходимо проанализировать, какого рода данные будут обрабатываться в системе и каким образом это должно происходить.

При этом следует учитывать тот факт, что СОВ должна функционировать адекватно угрозам информационной безопасности (ИБ), характерным для рассматриваемых объектов ИС, поэтому исходной позицией является выявление перечня угроз ИБ, характерных для данной ИС.

К сожалению, практически все существующие СОВ лишены функциональности, позволяющей связывать риски и угрозы ИБ с происходящими в сетевой и локальной вычислительной среде событиями. Современные СОВ направлены самое большее на связывание уязвимостей аппаратно-программного обеспечения и систем защиты с сетевыми событиями. В результате такого, на наш взгляд, одностороннего анализа, когда в расчет принимаются только технические параметры сети и при этом их весьма ограниченное количество, страдает в первую очередь качество обнаружения вторжений. С другой стороны, предлагаемая связь рисков и угроз ИБ позволяет оценить возможные потери от реализации вторжения и сформировать наиболее экономически выгодные приоритеты по укреплению ИБ.

Более того, пользователь такой системы никогда не получит той информации, ради которой эти системы эксплуатируются, — информации о реализации угроз ИБ, которым подвержена защищаемая сетевая и локальная инфраструктура.

2. Обнаружение угроз безопасности

Для описания предлагаемого нового подхода введем понятия, которые будут применяться в дальнейшем. Под информационной системой в данной работе будет пониматься совокупность технических средств (компьютеров, коммуникационного оборудования, линий передачи данных), при помощи которых обеспечивается обработка информации в организации.

Под угрозой ИБ будем понимать потенциально возможное действие, предпринимаемое злоумышленником, а также все последующие события и процессы, которые это действие повлекло и которые могут привести к прямому или косвенному ущербу.

Атакой на ИС будем называть действие или некоторую последовательность действий, предпринимаемых злоумышленником для достижения результата в обход установленных политик безопасности (с учетом того, что в классическом понимании атака — это реализовавшаяся угроза). В нашем предложении рассматриваются действия, направленные на нарушение установленных владельцем правил функционирования системы, выполняемые при помощи различных средств вычислительной техники.

Целью предлагаемой концепции обнаружения угроз ИБ является определение новых требований и принципов конструирования СОВ, ориентированных на комплексную обработку информации о защищаемой инфраструктуре для своевременного выявления и предупреждения о возможности реализации угроз ИБ, присущих ИС.

На сегодняшний день пирамида информационной обработки данных в современной СОВ выглядит следующим образом (Рис. 1).

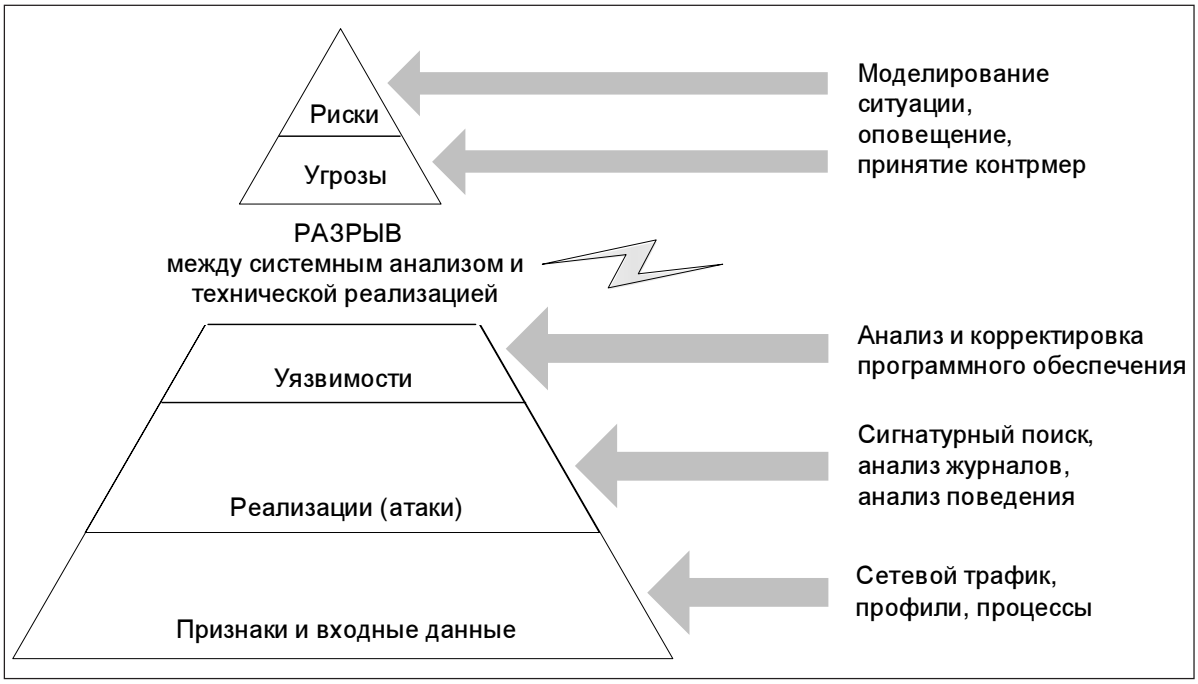

Рис. 1. Пирамида информационной обработки данных в современной СОВ.

Верхняя часть информационной пирамиды — это риски и угрозы ИБ, присущие рассматриваемой ИС. Ниже располагаются различные варианты реализаций угроз ИБ (атаки), и самый нижний уровень — это признаки атак. Конечный пользователь, равно как и СОВ, имеет возможность регистрировать только процесс развития конкретной атаки или свершившийся факт атаки по наблюдаемым характерным признакам. Признаки атаки — это то, что можно реально зафиксировать и обработать различными техническими средствами. Следовательно, необходимы средства фиксации признаков атак.

Если данный процесс рассматривать во времени, то можно говорить, что определенные последовательности наблюдаемых признаков порождают события безопасности. События безопасности

могут переводить защищаемые объекты ИС в небезопасное состояние. Следовательно, для СОВ необходим информационный срез достаточной полноты, содержащий все события безопасности, произошедшие в ИС за рассматриваемый период. Кроме того, поднимаясь вверх по пирамиде, для события безопасности можно указать, к реализации какого вида угроз ИБ оно может привести, для того чтобы в процессе развития атаки производить прогнозирование ее развития и принимать меры по противодействию угрозам ИБ, которые может вызывать данная атака.

Методология обработки данных в современных ИС подразумевает повсеместное использование многоуровневости. Для СОВ нового типа можно выделить следующие крупные уровни, на которых возможно осуществление доступа к обрабатываемой информации:

1. Уровень прикладного программного обеспечения (ПО), с которым работает конечный пользователь ИС. Прикладное ПО зачастую имеет уязвимости, которые могут использовать злоумышленники для доступа к данным, обрабатываемым этим ПО.

2. Уровень СУБД. Уровень СУБД является частным случаем средств прикладного уровня, но должен выделяться в отдельный класс в силу своей специфики. СУБД, как правило, имеет свою собственную систему политик безопасности и организации доступа пользователей, которую нельзя не учитывать при организации защиты.

3. Уровень ОС. ОС компьютеров защищаемой ИС является важным звеном защиты, поскольку любое прикладное ПО использует средства, предоставляемые именно ОС. Бесполезно совершенствовать качество и надежность прикладного ПО, если оно эксплуатируется на незащищенной ОС.

4. Уровень среды передачи. Современная ИС подразумевает использование различных сред передачи данных для взаимосвязи аппаратных компонентов, входящих в состав ИС. Среды передачи данных являются на сегодняшний день одними из самых незащищенных компонентов ИС. Контроль среды передачи и передаваемых данных — один из обязательных составляющих механизмов защиты данных.

Иллюстративно уровни обработки потоков данных в ИС изображены на рис. 2.

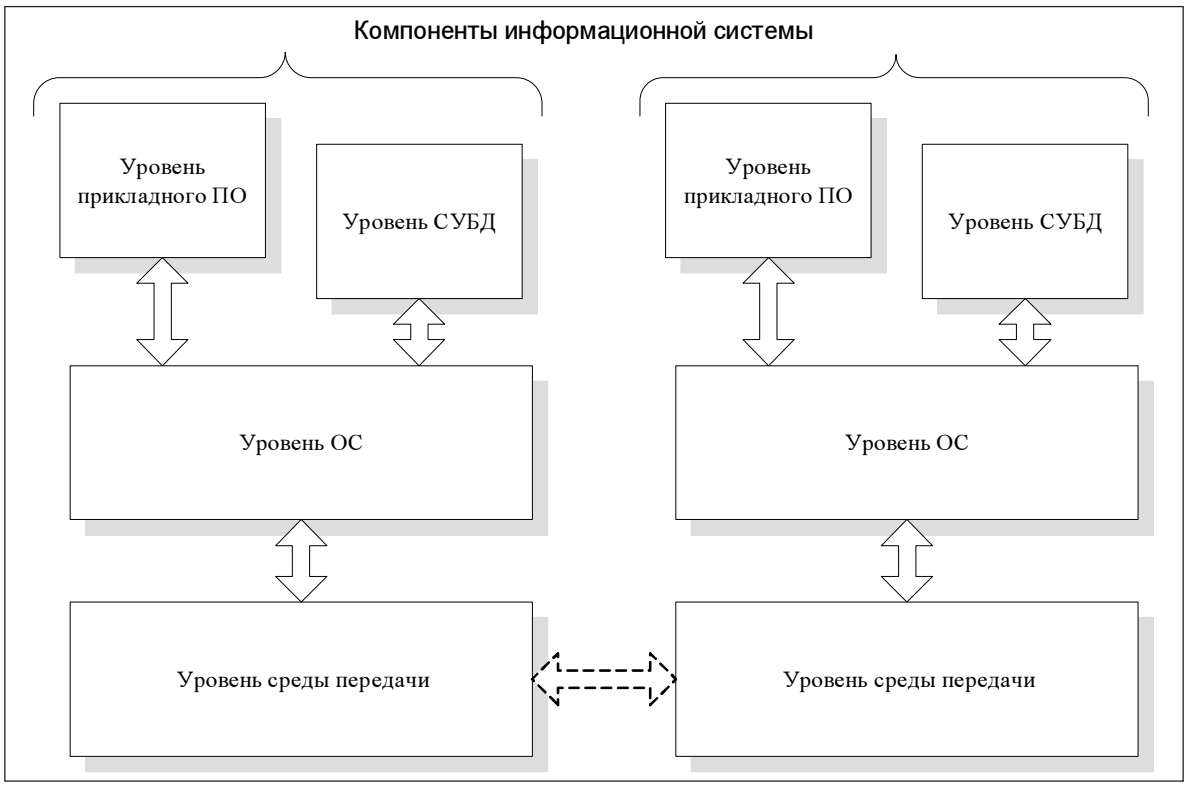

Рис. 2. Уровни обработки информации в ИС.

Исходя из вышесказанного, можно сделать вывод, что любые средства защиты информации, в том числе и системы обнаружения и предупреждения вторжений, обязаны иметь возможность анализировать обрабатываемые и передаваемые данные на каждом из выделенных уровней. Требование присутствия СОВ на каждом функциональном уровне ИС приводит к необходимости выделения подсистемы регистрации событий безопасности в отдельный комплекс информационных агентов СОВ, обеспечивающих сбор информации в рамках всей сети ИС. В то же время разнородность программно-аппаратных платформ и задач, решаемых различными объектами ИС, требует применения модульной архитектуры информационных зондов для максимальной адаптации к конкретным условиям применения.

3. Использование знаний об угрозах ИБ для обнаружения атак на ИС

Угрозы ИБ, как правило, каким-либо образом взаимосвязаны друг с другом. Например, угроза захвата уязвимого веб-сервера узла сети может привести в реализации угрозы полного захвата управления данным узлом. Поэтому в целях прогнозирования и оценки ситуации целесообразно учитывать вероятностную взаимосвязь угроз.

Если рассмотреть множество U — множество угроз ИБ рассматриваемой ИС, $u_i \in U$ — i-я угроза. В предположении, что множество угроз ИБ конечно, будем считать, что реализация i-й угрозы ИБ может с некоторой вероятностью приводить к возможности реализации других угроз ИБ. При этом возникает задача вычисления $P(u|u_{i_1}, u_{i_2},..., u_{i_k})$ — вероятности реализации угрозы ИБ u при условии реализации угроз ИБ $u_{i_1}, u_{i_2},..., u_{i_k}$ (Рис. 3).

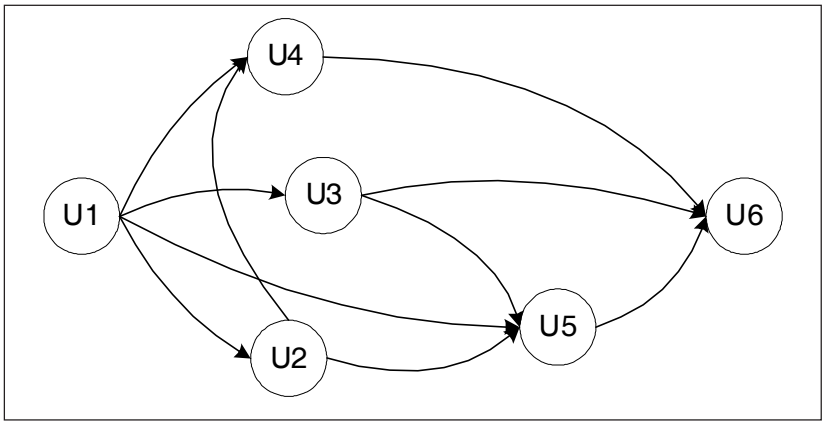

Рис. 3. Вид графа зависимости угроз ИБ.

Атаку можно обнаружить тем более полно, чем более полная имеется информация о произошедшем событии. Как видно из предыдущих разделов, современные СОВ чаще всего фиксируют атаки по наличию определенной, вполне конкретной сигнатуры.

Расширив этот подход, можно акцентировать внимание на процессе выделения в атаках различных этапов (фаз) их реализации [1]. Выделение фаз атак, особенно ранних, является важным процессом, который, в конечном счете, позволяет обнаружить атаку в процессе ее развития. Однако сделать это возможно, лишь определив соответствующим образом перечень угроз ИБ для ИС, которые могут реализовываться на каждой из фаз атаки, и соответствующим образом отразив данный факт в классификации.

В самом крупном приближении выделяются три основные фазы атаки: сетевая разведка; закрепление; сокрытие следов и реализация целей.

Анализ взаимосвязи угроз ИБ с фазами атаки и прогнозирование наиболее вероятных угроз ИБ, которые могут быть реализованы злоумышленником, являются важной задачей обеспечения ИБ и необходимы для своевременного принятия решений по блокировке злонамеренных воздействий. Такой анализ дает возможность не только предупредить наиболее опасные в смысле последствий атаки, но и снизить ошибки распознавания атак первого и второго рода.

Следующим элементом концепции обнаружения вторжений является классификация. Вопросы классификации атак до сих пор активно исследуются. Основная задача разработки классификации атак состоит в том, чтобы обеспечить удобство использования данной классификации на практике. Основными требованиями к классификации являются следующие: непересекающиеся классы, полнота, применимость, объективность, расширяемость, конечность. Интересные подходы к классификации сетевых атак предложены в [1, 3]. Классификация угроз ИБ должна учитывать структуру и фазы проведения атаки на ИС, определять такие атрибуты, как источники и цели атаки, их дополнительные характеристики, многоуровневую типизацию. Модель обнаружения вторжений должна строиться на базе разработанной классификации.

Таким образом, в перспективе необходимо решение следующих задач: определение наиболее вероятной реализации угрозы ИБ на текущий момент времени для того, чтобы иметь представление, какие последствия могут в ближайшее время ожидать ИС, а также составление прогноза развития ситуации с целью определения наиболее вероятной реализации угроз ИБ впоследствии.

4. Повышение эффективности систем обнаружения вторжений — интегральный подход

Вообще говоря, современные системы обнаружения атак еще далеки от эргономичных и эффективных с точки зрения безопасности решений. Повышение же эффективности следует вести не только в области обнаружения злонамеренных воздействий на защищаемые объекты информатизации, но и с точки зрения повседневной «боевой» эксплуатации данных средств, а также экономии вычислительных и информационных ресурсов владельца данной системы защиты.

Если же говорить непосредственно о модулях обработки данных, то, следуя логике предыдущего раздела, каждая сигнатура атаки в представленной схеме обработки информации об атаке является базовым элементом для распознавания более общих действий — распознавания фазы атаки (этапа ее реализации). Само понятие сигнатуры обобщается до некоторого правила (например, с помощью поиска аномалий в сетевом трафике или клавиатурном почерке пользователя). А каждая атака, наоборот, разбивается на набор этапов ее проведения — фаз атаки. Чем проще атака, тем проще ее обнаружить и тем больше возможностей появляется для ее анализа. Каждая сигнатура отображает определенное событие в вычислительной сетевой и локальной среде в фазовое пространство компьютерных атак. При определении фазы лучше сохранять достаточную степень детализации, чтобы иметь возможность описывать атаки с помощью подробных сценариев атак (списка фаз атак и переходов между ними).

Сценарий атаки в этом случае представляет собой граф переходов, аналогичный графу конечного детерминированного автомата. А фазы атак можно описать, например, следующим образом:

— перенос зоны (zone transfer) для домена;
— определение включенных в компьютерную сеть компьютеров;
— опробование портов;
— определение ОС;
— сбор баннеров;
— идентификация служб и приложений;
— идентификация программных и аппаратных средств;
— применение эксплоитов (реализаций уязвимостей программного обеспечения);
— дезорганизация функционала сети с помощью атак на отказ в обслуживании;
— управление через программные закладки;
— поиск установленных программных закладок и вирусов;
— поиск прокси-серверов;
— удаление следов присутствия и предпринятых действий;
— и т. д. (по необходимости с различной степенью детализации).

Преимущества такого подхода очевидны: в случае раздельной обработки различных этапов атаки появляется возможность распознавания атаки еще на этапе ее подготовки, а не на стадии ее реализации, как это происходит в существующих системах. При этом элементной базой для распознавания может являться как сигнатурный поиск, так и выявление аномалий, использование экспертных методов и систем, доверительных отношений и прочих информационных уже известных и реализованных сетевых и локальных методов оценки происходящего в вычислительной среде потока событий.

Обобщающий подход к анализу позволяет соответственно определять и комбинированные угрозы — как во временном, так и в логическом и физическом пространстве. Общая схема обработки поступающих событий также позволяет осуществлять поиск распределенных атак — путем последующей агрегации данных из различных источников и конструирования метаданных об известных инцидентах по защищаемому «периметру» (Рис. 4).

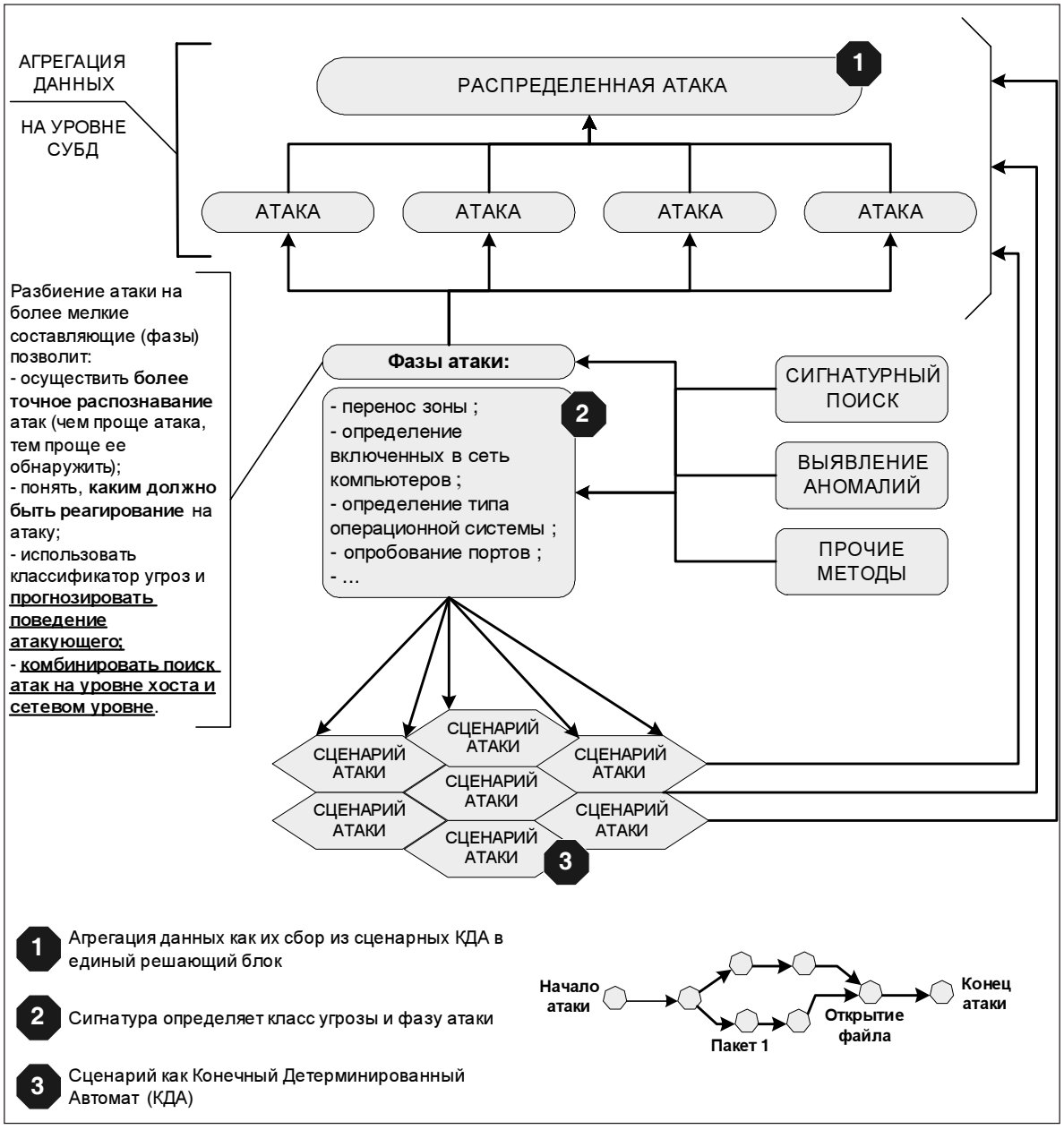

Рис. 4. Схема интегрального обнаружения вторжений.

Атаки выявляются путем агрегации данных о поступающих атаках и подозрительных действиях и сопоставления шаблонов и статистической фильтрации. Таким образом, оповещение о подозрительных действиях в компьютерных системах происходит на нескольких уровнях:

— нижний уровень сообщает о примитивных событиях (совпадении сигнатур, выявлении аномалий);

— средний уровень извлекает информацию из нижнего уровня и агрегирует ее с помощью конечных автоматов (сценариев атак), статистического анализа и механизмов пороговой фильтрации;

— высший уровень агрегирует информацию с двух предыдущих и позволяет выявлять обычные и распределенные атаки, их реальный источник и прогнозировать его дальнейшее поведение на основе применения методов анализа с элементами искусственного интеллекта.

Ядро системы обнаружения вторжений должно быть четко разделено с системой визуализации и сигнализации.

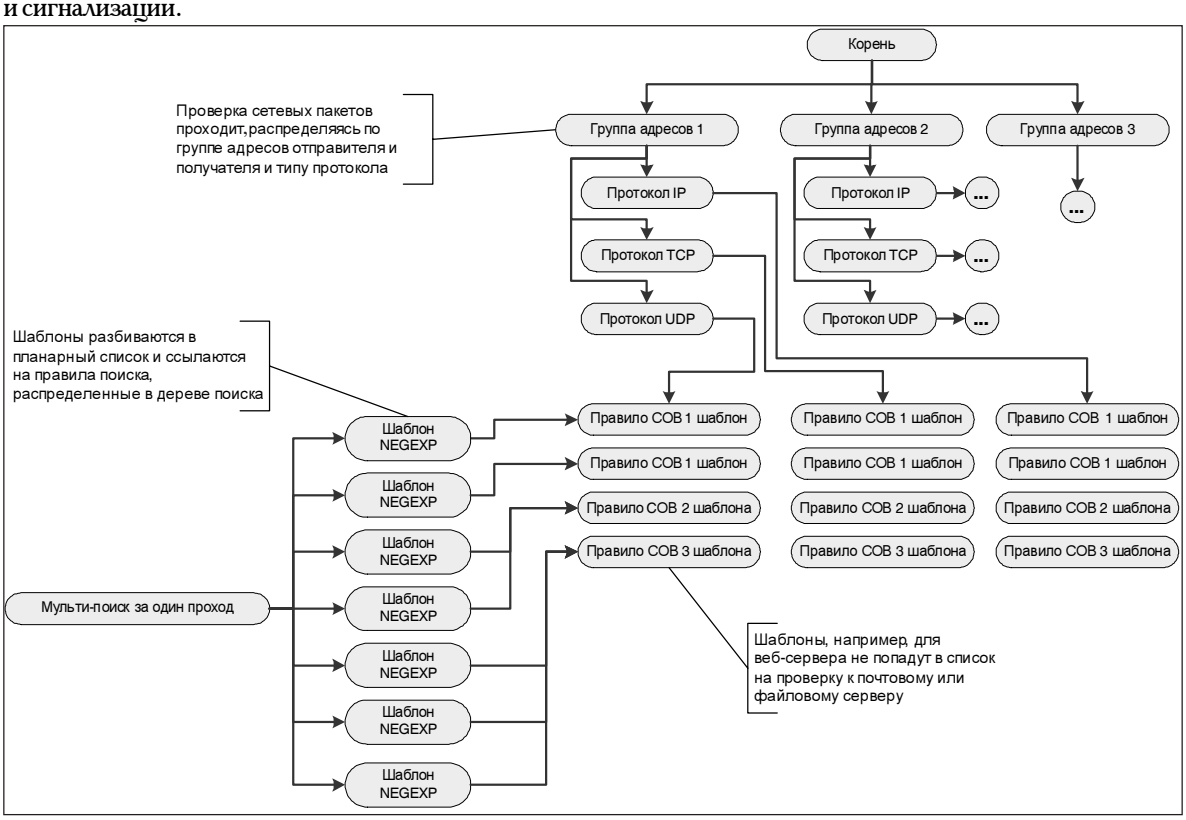

Рис. 5. Схема интегрального обнаружения компьютерных вторжений.

Для поиска сигнатур в сетевых пакетах используются правила, формирующие перечень опций (паспорт), по которым осуществляется проверка поступающих сетевых пакетов.

Существующие системы (как, например, Snort или PreludeIDS, которая использует правила Snort) применяют строчный вид описаний таких правил:

alert tcp $HOME_NET 1024:65535 -> $EXTERNAL_NET 1024:65535 (msg:"BLEEDING-EDGE TROJAN Trojan.Win32.Qhost C&C Traffic Outbound (case1)"; flow:established; dsize:>1000; content:"|00 00 00 28 0a 00 00 02 0f|Service Pack 1|00|"; classtype:trojan-activity; reference: url,/www.viruslist.com/en/viruses/ encyclopedia?virusid=142254; sid:2007578; rev:1;)

Такой вид более удобен для быстрой машинной обработки, но менее пригоден для человека. Кроме того, в нем отсутствуют возможности для расширения функциональности, которые заложены в XML-подобных реализациях сигнатурных баз. Например, простая «скобочная» (от англ. brace-like) конфигурация позволяет записать ряд управляющих переменных и описать правила в визуальной форме,

сохраняя возможность для легкого расширения функциональности. Так, определение фаз атак, защищаемых объектов и совершаемых в сети событий может выглядеть следующим образом:

```
            type_defs {
                  alert = 1;
                  warning = 2;
                  fail = 4;
            }
            srcdst_defs {
                  HOME_NET = 195.208.245.212
      localhost = 127.0.0.1
      }
      proto_defs {
                  tcp = 1;
                  udp = 2;
                  tcp-flow = 10;
      }
      phase_defs {
                  port_scanning = 1;
                  exploiting = 2;
                  icmp_sweeping = 3;
                  ftp_bouncing = 4;
                  shell_using = 5;
                  dir_listing = 6;
                  file_opening = 7;
      }
```

А конфигурационная секция определения угроз ИБ может иметь основные позиции, подобные следующей:

```
      treat_defs = {
                  treat {
      name = file-unauthorised-access;
      id = FUAC;
      msg = "message in english";
      }
      }
```

Кроме указанных в гибкой форме угроз, фаз атак и защищаемых объектов, интегральная обработка информации, связанная с выявлением угроз ИБ, позволяет ввести также сервис-ориентированный подход к обнаружению вторжений, формируя автоматическим или ручным способом описания сетевых и локальных служб, а также приоритезируя важность с точки зрения обеспечения должного уровня ИБ и жизнедеятельности информационной инфраструктуры сети.

```
      service_defs = {
                  service {
                        name = pop3;
      msg = "";
      rulesets = "backdoors, pop3scanners";
      security_tolerance = 3
      life_insurance = 5
      }
      }
```

Сами же правила выглядят, например, следующим образом:

```
      ruleset {
                  name = backdoors;
                  rule {
                        id = 0x1000;
                        type = alert;
```

```
proto = tcp;
src = localhost;
dst = 195.208.245.0/24:2000;
                    msg =  "service::what is bad in this alert";
                    options = AP,vice_versa;
                    contains = "|0a0a0d03|";
                    phase = exploiting;
                    treat = file-unauthorized-access;
revision = 1;
}
}
```

Здесь учитываются как классические признаки события (тип события, протокол обнаружения, источник и объект воздействия, краткое сообщение), так и добавочные — фаза атаки, тип угрозы, к возникновению которой относится данное событие. При этом сами правила могут быть сгруппированы в наборы, пригодные затем для связывания их с установленными в защищаемой системе сетевыми и локальными службами.

Если же вернуться к эффективности проверки правил в системах обнаружения вторжений, то следует отметить следующий факт. На текущий момент все правила в СОВ проверяются так, как показано на рис. 6. Проверка неоднородных по своей сущности правил происходит раздельно, правило за правилом, при этом однородные операции над пакетами выполняются все время порознь. Такой подход не позволяет эффективно распараллелить обработку сетевых пакетов, полностью использовать возможности нескольких конвейеров на современных процессорах, а также оптимизировать поиск частично похожих правил-сигнатур.

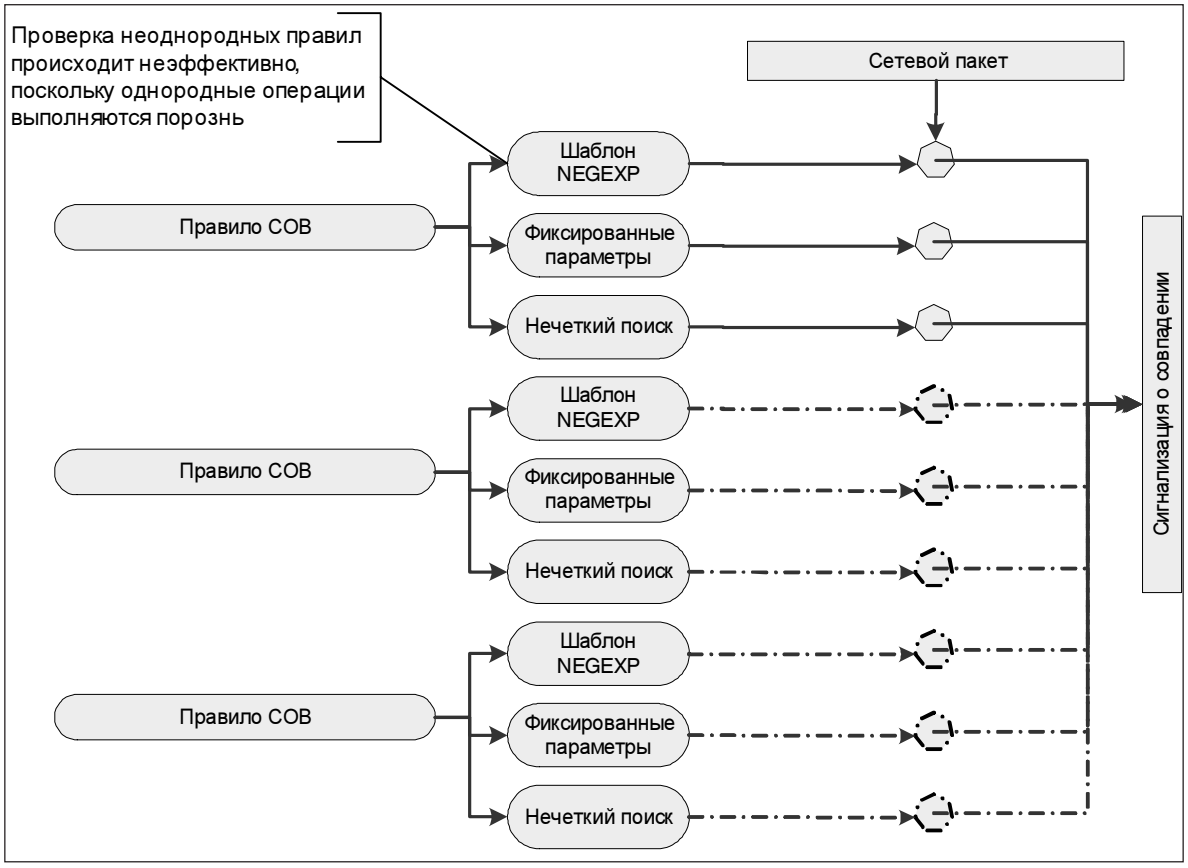

Рис. 6. Неэффективный способ проверки.

Для того чтобы ликвидировать эти недостатки, предлагается однородные (с точки зрения вычислительных действий) операции сводить в единые списки со ссылками на исходные правила (Рис. 7).

Параллельность обработки в этом случае достигается за счет использования «корзин» и «штрафных мячей» (каждое правило имеет пустую корзину, а за каждое совпадение шаблона или другой части правила в корзину добавляется один мяч; при достижении определенного количества мячей правило считается совпавшим). Единство списков позволяет убрать неэффективные, совпадающие в разных правилах, шаблоны.

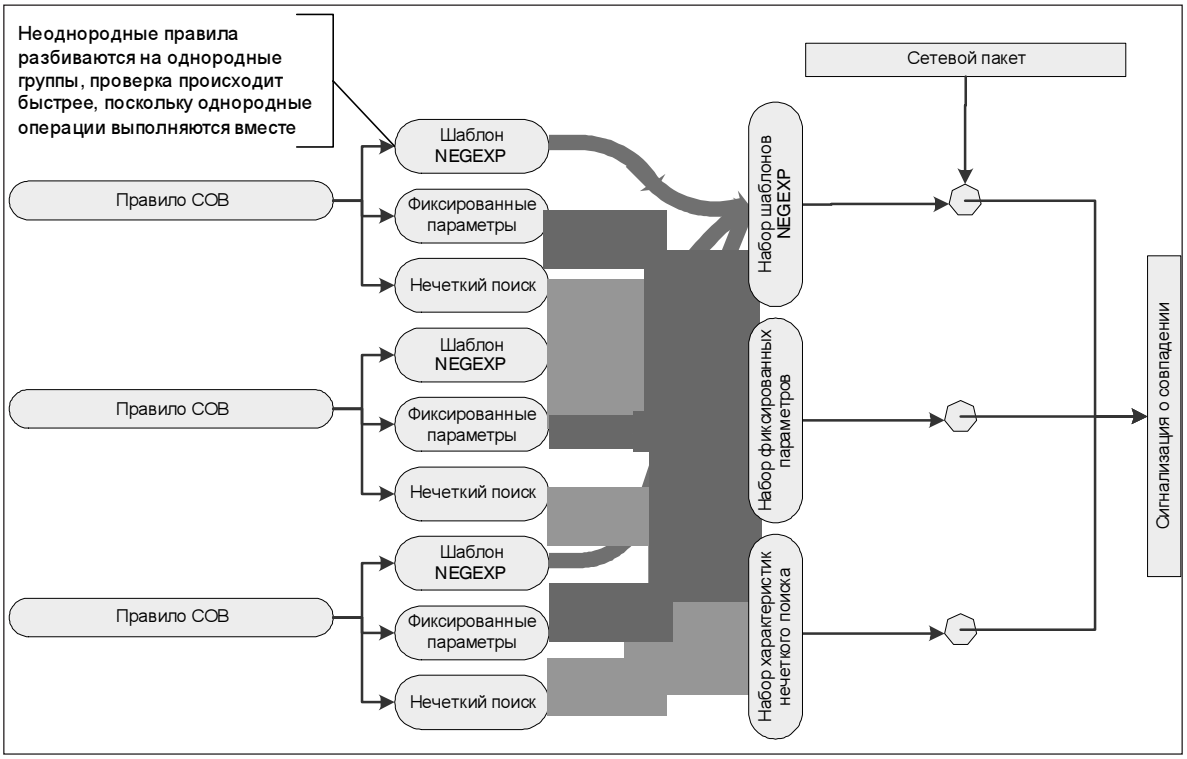

Рис. 7. Метод послойной проверки.

Однако есть минус и у такого подхода, когда, например, шаблоны связаны друг с другом (вот пример такого шаблона: найти первое вхождение, затем относительно него через несколько байт проверить наличие определенной бинарной последовательности). Правда, таких правил — меньшинство (даже если судить по стандартным правилам популярной СОВ Snort), что позволяет вынести их в отдельный класс распараллеливаемых методов и использовать в них любые простые методы последовательной проверки.

Помимо преимущества в распараллеливании процесса поиска сигнатур, становится возможным применение методов одновременного поиска многих сигнатур в сетевом потоке за один проход (можно, например, построить один большой конечный автомат для большинства шаблонов, участвующих в правилах, или использовать мультисигнатурную модернизацию алгоритма Бойера—Мура).

Экспериментальные проверки различных вариантов реализации методов одновременного поиска многих сигнатур показали, что наиболее быстрым оказывается реализация большого конечного автомата, модифицированного таким образом, чтобы он позволял «пропускать» однородные ошибки - пропуски и вставки произвольной длины, а также ошибки замены (в результате модификации сигнатуры, что является довольно частым явлением с целью ее сокрытия от СОВ).

Наиболее сложные в проверке правила (шаблоны) можно предварительно компилировать в бинарные подключаемые модули (как это сделано, например, в системе RealSecure IDS).

5. Заключение

Современный подход к построению СОВ и выявлению признаков компьютерных вторжений на ИС полон недостатков и уязвимостей, позволяющих, к сожалению, злонамеренным воздействиям

успешно преодолевать системы защиты информации. Переход от поиска сигнатур атак к выявлению предпосылок возникновения угроз ИБ должен в корне изменить данную ситуацию, сократив дистанцию отставания в развитии систем защиты от систем их преодоления.

Кроме того, такой переход должен способствовать повышению эффективности управления ИБ и, наконец, более конкретным примерам применения нормативных и руководящих документов, уже ставших стандартами.

СПИСОК ЛИТЕРАТУРЫ:

1. *Милославская Н. Г., Толстой А. И.* Интрасети: обнаружение вторжений. Учебное пособие для вузов. М., 2001. — 587 с.
2. *Лукацкий А. В.* Обнаружение атак. СПб., 2001. — 624 с. илл.
3. *Климовский А. А.* К анализу подходов классификации компьютерных атак // Материалы Международной научной конференции по проблемам безопасности и противодействия терроризму. М., 2006. — 480 с.
4. *Сердюк В. А.* Анализ современных тенденций построения моделей информационных атак // Информационные технологии. 2004. № 4.
5. *Новиков А. А., Устинов Г. Р.* Уязвимость и информационная безопасность телекоммуникационных технологий. М., 2003. — 296 с.
6. *Kotenko I., Man'kov E.* Agent-Based Modeling and Simulation of Computer Network Attacks // Proceedings of Fourth International Workshop Agent-Based Simulation 4 (ABS 4). Montpellier, France, 2003. P. 121–126.
7. *Kotenko I.* Teamwork of Hackers-Agents: Modeling and Simulation of Coordinated Distributed Attacks on Computer Networks // Proceedings of the 3rd International/Central and Eastern European Conference on Multi-Agent Systems (CEEMAS 2003). Lecture Notes in Artificial Intelligence, Springer-Verlag. Prague, Czech Republic, 2003. Vol. 2691. P. 464–474.