2.3. Полиморфизм программных злоупотреблений.

Описание тактических и стратегических программных злоупотреблений с точки зрения достижения конечных целей, логики работы основных механизмов, предпринято с целью обоснования возможности их интеграции, по аналогии с компьютерными вирусами.

Объекты живой и неживой природы обладают свойствами полиморфизма [129, с.1028]. Для неживой природы полиморфизм - свойство некоторых веществ существовать в нескольких кристаллических состояниях (модификациях) с разной структурой. Примером подобного полиморфизма является алмаз и графит. Полиморфизм объектов живой природы заключается в наличии в пределах одного вида резко отличающихся особей. Примером может служить пчелинная семья и наличие в ней матки, трутней и рабочих пчел.

По аналогии представляется возможным дать определение полиморфизма программных злоупотреблений - набор свойств, присущих отдельным специально разработанным программным средствам, позволяющий осуществлять внедрение, распространение, контроль и выполнение специфических функций в АИС в обход средств СИБ. Дополнительными свойством можно считать возможность перерастания простых (тактических) злоупотреблений в интегрированные (стратегические). В разделе 2.2 данной работы были описаны только общие формы реализации программных злоупотреблений.

Несмотря на данный общий подход, на сегодняшний день можно говорить о так называемом *полиморфизме программных злоупотреблений*. При этом имеется в виду, что сложные программные злоупотребления создаются за счет использования одного или нескольких простых программных злоупотреблений. Сложные злоупотребления могут обладать элементами экспертных систем и в

зависимости от обстоятельств могут “вести себя" в соответствии с конфигурацией, состоянием системы, могут менять поведение в зависимости от выполнения определенных внутренних или внешних условий. В качестве примера рассмотрим схему реализации полиморфного программного злоупотребления, представленного на рис. 13. В качестве полиморфного рассматривается программное злоупотребление, которое реализует тактический механизм для достижения тактической цели и после этого реализует стратегический механизм для достижения стратегической цели.

После запуска полиморфное программное злоупотребление должно проводить анализ характеристик среды на предмет возможности реализации своих механизмов. Анализ производится по двум направлениям. Первое направление - это анализ аппаратной части для определения платформы (персональные ЭВМ - могут быть Intel, Macin-tosh; рабочие станции - Sun, DEC и др.). Второе направление ориентировано на определение операционной системы и рабочей среды, используемой на каждом конкретном компьютере, информация о которой будет использоваться для определения возможного типа атаки.

После анализа среды, на основе полученной информации выбирается тактический механизм, используемый для получения доступа и реализуется попытка получения несанкционированного доступа к информационной системе. Следует отметить, что данный процесс является многоитерационным и может повторятся до тех пор, пока не будет получен доступ к информационной системе.

После получения доступа к системе реализуется дополнительный тактический механизм, если таковой предусмотрен. Дополнительный тактический механизм реализует основные функции по внедрению и запуску стратегическиого механизма, а именно: дополнительно анализируется среда и посылается злоумышленнику сообщение о возможности внедрения стратегического механизма.

Рис. 13 Блок-схема реализации полиморфных программных злоупотреблений

После анализа среды, в зависимости от целевой функции и конкретной аппаратной и программной среды производится выбор стратегического механизма. При этом могут быть выбраны пассивные и/или активные механизмы.

Следущим этапом является внедрение стратегического механизма, целью которого являестя реализация основной функции программного злоупотребления. В случае, если внедрение стратегического механизма произошло со сбоями, реализуется возврат на один шаг назад и производится попытка выбора другого стратегического механизма.

В случае, когда внедрение стратегического механизма призошло нормально, тактический механизм реализует подготовку среды действия стратегического механизма, после чего уничтожается все следы действия тактического механизма во избежание раскрытия программного злоупотребления стандартными методами.

Все рассмотренные ранее этапы реализации полиморфного программного злоупотребления - вспомогательные. Основным этапом в реализации полиморфного программного злоупотребления является реализация стратегического механизма, призванного достичь основную цель программного злоупотребления. Если реализация этого механизма не проходит нормально, то происходит возврат на этапе выбора другого стратегического механизма или на этапе выбора и реализации тактического механизма (для случая, когда надо получить доступ к другому компьютеру, сети, системе). При успешной реализации рассматриваемого механизма может выполняться выдача нарушителю информации о проделанных операциях и достигнутых результатах. В качестве путей передачи информации нарушителю могут использоваться “скрытые каналы”.

Последним этапом реализации полиморфного программного злоупотребления является уничтожение следов действия стратегического механизма в информационной системе.

Анализируя опубликованные в литературе сведения о случаях реализации программных злоупотреблений, следует отметить, что первым программным злоупотреблением с полиморфными свойствами явился компьютерный “червь Морриса” [104]. Знаменитый компьютерный червь состоял из двух частей - блок внедрения (тактический механизм) и блок размножения (стратегический механизм).

Таким образом, можно сделать вывод о том, что в арсенале злоумышленников могут быть, кроме обычных, всем известных программных злоупотреблений еще и множество дополнительных, которые будут использовать элементы экспертных систем и самообучения для проникновения и нарушения безопасности АИС.

В связи с этим, при разработке системы безопасности информации в АИС следует иметь в виду наличие у возможных нарушителей полиморфных программных злоупотреблений, что значительно затрудняет противостояние.

Опираясь на исследования, связанные с процессами становления кибернетики [155,21,24 и др.], следует выделить логико-математические понятия, выражающие одинаковость (изоморфизм) и уподобление (гомоморфизм) программных злоупотреблений.

Программное злоупотребление A называеся изоморфным по отношению к злоупотреблению A^1, если между их элементами установлено взаимно-однозначное соответствие. При этом выполняются следующие условия:

- каждому элементу α, принадлежащему программному злоупотреблению A ($\alpha \hat{I} A$), соответствует элемент $\alpha^1 \hat{I} A^1$ и наоборот;

- каждой функции j, определенной в программном злоупотреблении A, для A^1 соответствует единственная функция j^1, и наоборот, функции j^1 в A^1 соответствует единственная функция j в A;

- для каждого свойства P, которым обладают элементы A, и каждого отношения R, в котором находятся наборы элементов из A, для

образцов этих элементов в A^1 существуют взаимно-однозначно соответствующие им свойства P^1 и R^1.

Замена первого условия более слабым требованием приводит к более общему (и более слабому) отношению гомоморфизма. Гомоморфный образ упрощает структуру пробраза программного злоупотребления, так как допускает множество “склеенных” элементов, соответствующих элементу $\alpha^1 \hat{I}A^1$.

Аналогично ослабление второго и третьего условий ведет к понятиям, выражающим упрощение уподобления программного злоупотребления A^1 системе A.