**Заключение.** В данной работе проведено экспериментальное исследование временных характеристик программной реализации операции умножения в трёхмодульной системе остаточных классов и её модифицированной совершенной форме. Показано, что использование последней позволяет существенно уменьшить время процесса вычислений за счёт исключения выполнения операции поиска обратного элемента по модулю и умножения на него при переводе в десятичную систему исчислений. Представлены графические зависимости временных характеристик, которые подчёркивают преимущества использования модифицированной совершенной формы системы остаточных классов.

**СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ**


1. Николайчук, Я.Н. Теория источников информации / Я.Н. Николайчук. – Тернополь : ТНЭУ, 2008. – 536 с.
2. Задирака, В.К. Компьютерная криптология / В.К. Задирака, А.С. Олексюк. – Тернополь : ТАНХ, 2002. – 504 с.
3. Yatskiv, V. The Use of Modified Correction Code Based on Residue Number System in WSN / V. Yatskiv, N. Yatskiv, Su Jun, A. Sachenko, Hu Zhengbing // Proceedings of the 7-th2013 IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS'2013). – Berlin, Germany. – 2013. – Volume 1. – P. 513–516.
4. Omondi, A. Residue number systems: theory and implementation / A. Omondi, B. Premkumar. – London: Imperial College Press, 2007. – 296 p.
5. Бухштаб, А.А. Теория чисел / А.А. Бухштаб. – Москва : Просвещение, 1966. – 384 с.
6. Вербицкий, О.В. Вступление в криптологию / О.В. Вербицкий. – Львов : Научно-техническая литература, 1998. – 248 с.
7. Kasianchuk, M. Algorithms of findings of perfect shape modules of remaining classes system / M.Kasianchuk, I.Yakymenko, I.Pazdriy, O.Zastavnyy // The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM-2015): proceedings of the XIII-th International Conference – Polyana-Svalyava (Zakarpattya), Ukraine. – 2015. – P. 168–171.
8. Nykolaychuk, Ya.M. Theoretical Foundations of the Modified Perfect Form of Residue Number System / Ya.M. Nykolaychuk, M.M. Kasianchuk, I.Z. Yakymenko // Cybernetics and Systems Analysis. – 2016. – Vol. 52, № 2. – P. 219–223.
9. Kasianchuk, M.N. Theory and Methods of Constructing of Modules System of the Perfect Modified Form of the System of Residual Classes / M.N. Kasianchuk, Ya.N. Nykolaychuk, I.Z. Yakymenko // Journal of Automation and Information Sciences. – 2016. – Vol. 48, № 8. – P. 56–63.


*Материал поступил в редакцию 14.01.2018*

***KASIANCHUK M.N. The experimental research of software implementation of multiplication operation in the three-module system of residual classes***

The present work is devoted to an experimental study of the time characteristics of the software implementation of the multiplication operation in the three-module system of residual classes and its modified perfect form. It has shown that the utilization of the modified perfect form allowed significantly reduce the time of the calculation process by excluding of the execution of the searching operation of the inverse element by the module and multiplying it by transition to the decimal system of calculation. Graphic dependencies of time characteristics were presented and they were emphasized the advantages of the modified perfect form of the residual class system.

УДК 004.056.53+004.492.3

***Комар М.П.***

# ПОВЫШЕНИЕ БЕЗОПАСНОСТИ СИСТЕМЫ ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ

**Введение**. На сегодняшний день для защиты компьютерных систем от вторжений, в основном, используется специализированное программное обеспечение (ПО) [1–9]. Однако такое ПО является уязвимым к самим вторжениям, из-за того, что могут перехватываться системные функции операционной системы, которая дает возможность активно противодействовать обнаружению и удалению их программными средствами. При этом вторжения способны блокировать запуск специализированного ПО, отслеживать его действия и возобновлять удаленные вредные процессы, изменять настройку в системном реестре и т. п.

В последнее время значительное развитие получили исследования в области коллективного (роевого) интеллекта (Swarm Intelligence) [10]. Данный научное направление возникло в рамках направления искусственного интеллекта и описывает коллективное поведение децентрализованной самоорганизующейся адаптивной системы. Системы коллективного интеллекта – это мультиагентные системы, состоящие из множества агентов, которые взаимодействуют между собой и с окружающей средой, образуя таким образом коллективный интеллект. В основу систем коллективного интеллекта положены биологические основы поведения животных и на основе наблюдений доказано, что групповой интеллект зачастую превосходит умственные способности одной особи.

В отличие от классического подхода искусственного интеллекта, по которому для определенной задачи создается одна интеллектуальная система, автором предложено использовать мультиагентный подход для обнаружения и классификации вторжений, где один агент имеет неполное представление о глобальной угрозе, поэтому создается множество агентов (детекторов атак) и обеспечивается эффективное взаимодействие между ними. Глобальное поведение всей системы рассматривается как результат взаимодействия множества простых агентов.

В данной работе в качестве агента (детектора атак) предложено использовать архитектуру нейронной сети [11, 12], которая разработана в [13–16]. Предложен подход интеграции нейросетевых агентов в искусственную иммунную систему. Реализация данного подхода осуществляется на основе базовых принципов и механизмов биологической иммунной системы: генерация и обучение иммунных агентов, отбор агентов, которые по определенным причинам генерируют ошибочные решения, функционирование агентов, активация агентов и формирования иммунной памяти; и на основе базовой схемы искусственной иммунной системы [17–19].

Итак, в интеллектуальной системе обнаружения вторжений применяется совокупность агентов [20–22], где каждый агент отвечает за обнаружение и классификацию вторжений определенного вида, а совокупность таких агентов выполняет защиту системы в целом. Однако программная реализация такого подхода имеет все недостатки программной защиты, которые указаны выше.

**Предлагаемый подход к повышению безопасности системы обнаружения вторжений.** Предлагается защищать компьютерные системы от вторжений аппаратными средствами. Аппаратное решение работает не в среде зараженной операционной системы, потому все действия вторжений будут безрезультатными, а вторжения будут

***Комар Мирослав Петрович,*** *к.т.н., доцент кафедры информационно-вычислительных систем и управления Тернопольского национального экономического университета, e-mail:* mko@tneu.edu.ua.
*Украина, ТНЭУ, 46000, г. Тернополь, ул. Львовская, 11.*

быстро обезврежены.

При этом к аппаратным средствам относятся следующие требования:

1. Высокая надежность системы защиты от вторжений в целом. Для этого необходимо выделить отдельный компьютер для постоянного анализа вторжений и формирования соответствующих средств защиты, например, нейросетевых агентов, которые образуют интеллектуальную мультиагентную систему;
2. Высокая стойкость к вторжениям интеллектуальной мультиагентной системы. Для этого анализ вторжений, обучения нейросетевых агентов и подготовка к модификации аппаратных средств текущего обнаружения и обезвреживания вторжений должна осуществляться на упомянутом выделенном компьютере, который не подключен к сети;
3. Высокая стойкость к вторжениям подсистемы текущего обнаружения и обезвреживания угрозы. Эта система должна быть полностью аппаратной;
4. Высокая гибкость подсистемы текущего обнаружения и обезвреживания угрозы. Для этого следует обеспечить возможность динамического периодического обновления нейросетевых агентов согласно результатам анализа вторжений. Для этого необходимо обеспечить запись новых нейросетевых агентов лишь упомянутым выделенным компьютером (а не компьютером, где функционирует аппаратное обеспечение подсистемы текущего обнаружения и обезвреживания угроз).

Для реализации поставленных требований целесообразно подсистему текущего обнаружения и обезвреживания угроз базировать на программируемых логических интегральных схемах (ПЛИС) [23]. При этом их перепрограммирование должен осуществлять выделенный компьютер анализа вторжений (он не подключен к сети, а потому не может быть объектом атаки). Подключенные к сети компьютеры не должны иметь ни средства реконфигурации ПЛИС, ни вообще доступа к выводам, которые руководят перепрограммированием ПЛИС. Из-за того, что ПЛИС является аппаратными узлами, возникает вопрос об оптимизации подсистемы текущего обнаружения и обезвреживания угроз, в частности, уменьшение количества нужных макроячеек. Для этого, по результатам анализа структуры системы защиты целесообразно специализировать функции каждой подсистемы. При этом необходимо выделить такие подсистемы:

- обнаружения и классификации вторжений;
- принятия решений о методах обеспечения противодействия угрозам;
- реализации принятых решений.

Целью данной работы является улучшение безопасности системы обнаружения вторжений за счет интеграции нейросетевых агентов в искусственную иммунную систему и аппаратного выполнения части узлов системы защиты.

**Обобщена структура интеллектуальной системы обнаружения вторжений.** Согласно сформированным выше требованиям к выполнению узлов системы защиты от вторжений разработана структурная схема (рисунок 1). Она состоит из двух частей, первая из которых реализована программно и содержит систему анализа вторжений (решает вопрос, являются ли подозрительные действия действительно неизвестными данной компьютерной системе вторжением), систему обучения нейронных сетей (если подозрительные действия действительно являются вторжением, то учится новый нейросетевой агент) и систему управления и планирования (планирует методы анализа вторжений и руководит записью и стиранием нейросетевых агентов в интеллектуальной системе защиты. Эта часть не должна обязательно работать в реальном времени.

Вторая часть содержит буфер памяти сообщений, которые поступают из внешней сети (по отношению к сети, которую мы защищаем), мультиагентную систему, которая содержит нейросетевые агенты, которые обнаруживают вторжения, систему принятия решений, которая, на основе исходных сигналов нейросетевых агентов, принимает решение о содержании буфера (нормальные действия или вторжения). Эта часть работает в реальном времени, потому, с учетом стойкости к вторжениям, ее целесообразно выполнить на программируемой логической интегральной схеме – ПЛИС.

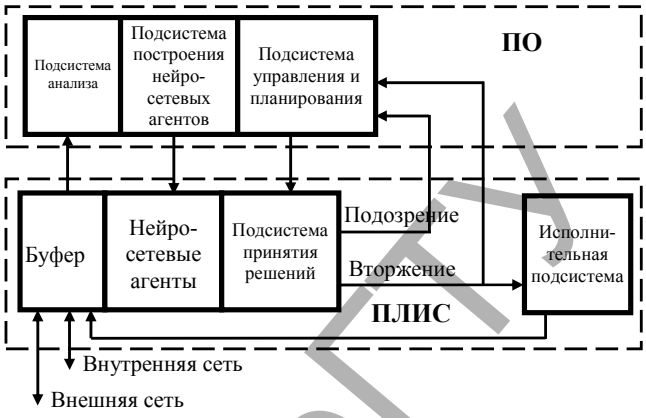


***Рисунок 1 –*** Обобщена структура системы обнаружения вторжений

Все сообщения из сети поступают в буфер, где задерживаются на время анализа угрозы. Анализ выполняет каскад нейросетевых агентов, при этом все агенты сканируют буфер постоянно и параллельно. При обнаружении угрозы каждый нейросетевой агент обращается к системе принятия решений. Эта система реализует принятие решений согласно правилам нечеткого вывода Мамдани [24, 25]. Такое техническое решение дает возможность гибко устанавливать и менять уровень «подозрения» к сообщениям, которые поступают из сети в соответствии с общим состоянием системы защиты в сети, уровня доверия к данному источнику сообщений и т. п. Система принятия решений имеет два выхода. Первый, «подозрение», дает команду системе анализа вторжений – появилась угроза. В таком случае система анализа должна провести распознавание и, если такие действия будут признаны вторжением, научить новый нейросетевой агент (или переучить старый). Тогда каскад нейросетевых агентов обновляется и повторяется анализ подозрительного кода. Если подозрительный код не будет классифицирован как вторжение, то проводится коррекция уровня срабатывания системы принятия нечетких решений [26].

Однако вероятен случай, когда подозрительный код будет распознан как вторжение каскадом нейросетевых агентов (его вывод будет однозначным). В таком случае система принятия решений дает команду «вторжение» непосредственно подсистеме выполнения решений на обезвреживание соответствующей угрозы. Последняя работает как жесткий цифровой автомат, который записывает логические нули в соответствующую область буфера. Для учета обнаруженных вторжений и их типов, а также обновления статистики угроз из разных источников сообщений, сигнал «вторжение» подается также на систему управления и планирования. Высокая стойкость системы защиты от вторжений обеспечена тем, что все ее постоянно работающие узлы выполнены на ПЛИС, то есть является абсолютно стойкими к вторжениям.

**Подсистема принятия решений.** Подсистема принятия решений работает согласно правилам нечеткого вывода Мамдани [24, 25], который предусматривает вычисление функции принадлежности как центра веса подобной трапеции сложной фигуры. Для вычисления координат центра веса допущено, что функции принадлежности являются плоской фигурой одинаковой толщины. Тогда радиус-вектор центра веса $r$ можно вычислить по формуле $r = \sum r_i m_i / \sum m_i$, где $r_i$ и $m_i$ – координата центра веса и масса *i*-го прямоугольника, из которых составлена фигура, центр веса ее следует найти.

Структурная схема подсистемы принятия решений разработана в [27–28].

**Подсистема построения нейросетевых агентов**. Для выбора базовой архитектуры нейронной сети проведены теоретические и экспериментальные исследования на базе трех нейросетей – многослойного персептрона (MLP), нейронной сети на основе радиально базисных функций (RBF) и векторного квантования (LVQ). Теоретические исследования показали, что для обучения сети MLP размер учебной выборки должен составлять 4420 образов ($L(O(W/\varepsilon)$, $W=m(n+k+1)+k$), где $L$ – размер учебной выборки, $W$ – общее количество параметров, которые настраиваются, $\varepsilon$ – допустимая ошибка классификации, $O()$ – порядок величины, $n$ – количество входных нейронов, $m$ – количество нейронов скрытого слоя, $k$ – количество нейронов исходного слоя. Для нейронной сети RBF – количество нейронов скрытого слоя растет пропорционально размеру учебной выборки ($L1/3=<m<=L$), а для LVQ сети с десятью нейронами Кохонена в скрытом слое необходимо иметь учебную выборку размерностью большую 20 образов ($L>= 2m$).

В результате проведения экспериментальных исследований [13, 14] наилучшие результаты показала нейронная сеть LVQ.

Вероятность обнаружения вторжения типа dos_back для нее составила 99% при уровне ошибок второго рода 0,2%. Эта же нейронная сеть показала результат обнаружения вторжений типа dos_neptune на 100% при уровне ошибок второго рода 0,1%. Поэтому за основу выбрана нейронная сеть векторного квантования (LVQ) с нейронами Кохонена в скрытом слое, которая характеризуется малым объемом учебной выборки и позволит учить нейросетевые агенты на вторжениях, которые характеризуются малым количеством записей.

Структура нейросетевого агента разработана в [13, 14]. Первый слой нейронных элементов является распределительным и предназначен для распределения входных сигналов на нейроны скрытого слоя. Входными сигналами являются параметры вторжения. Количество нейронных элементов распределительного слоя равняется количеству параметров вторжения. Второй слой нейронной сети состоит из нейронов Кохонена. Слой Кохонена осуществляет кластеризацию входного пространства образов, в результате чего образуются кластеры разных образов, каждому из которых отвечает свой нейронный элемент. Нейроны скрытого слоя функционируют по принципу «победитель берет все». Третий слой состоит из двух линейных нейронных элементов, которые используют линейную функцию активации и осуществляют отображение кластеров, сформированных слоем Кохонена, в два класса. Активность исходных нейронов характеризует вторжение или нормальные действия.

Отличной особенностью предложенного нейромережевого агента является то, что нейроны в скрытом слое разделены на две группы: первая группа – вторжение; вторая группа нейронов – нормальные (легитимные) действия, что позволяет отдельно осуществить кластеризацию вторжений и нормальных (легитимных) действий в скрытом слое.

Для сжатия входных данных предложено использовать метод главных компонент (Principal Component Analysis) [29, 30]. Для определения числа главные компонент предложено использовать критерий относительной информативности, чтобы определить количество информации в $i$-й компоненте. Проведены экспериментальные исследования [13, 14], которые показали, что одна главная компонента содержит 52,4% информации, две главных компоненты – 71,7% информации, три главных компоненты – 88,4% информации, а 12 первых главных компонент содержат 99,2% информации о сетевых соединениях.

Следовательно, экспериментально подтверждено, что для успешного обучения нейросетевых агентов достаточно использовать 12 главных компонент, а не 41. Это позволило уменьшить размерность анализируемой информации.

Дополнительные исследования показали, что подход описанный выше, не позволяет адаптироваться к изменению характера компьютерных атак. Поэтому предложено интегрировать нейросетевые агенты в искусственную иммунную систему [15, 16]. Данный подход можно представить в виде следующей совокупности шагов:

1. Генерация нейросетевых агентов. Случайным образом происходит генерация разнотипных нейросетевых агентов.
2. Обучение нейросетевых агентов. Учебная выборка формируется для каждого агента отдельно путем выбора случайным образом данных, которые относятся к определенному типу вторжения, и данных, которые относятся к нормальным (легитимным) действиям. Таким образом, формируется множественное число входных образов для оучения *i-го* нейросетевого агента. Обучение агента осуществляется на основе контролируемого конкурентного обучения в соответствии с правилом «победитель берет все». При таком обучении весовые коэффициенты нейрона - победителя модифицируются только тогда, когда происходит корректная классификация входного образа, то есть входной образ отвечает заданному множественному числу нейронов в слое Кохонена.
3. Отбор агентов. Каждый обученый агент проверяется на тестовой выборке с целью минимизации ошибок обнаружения и классификации вторжений. Если *i-й* агент корректно классифицирует тестовые данные, он «допускается» к анализу вторжений. Если в функционировании *i-го* агента оказываются ошибки, он уничтожается, а вместо него генерируется новый агент.
4. Функционирование агентов. Обученные и отобранные агенты внедряются в подсистему обнаружения и классификации вторжений. На вход агентов подаются параметры вторжений, а на выходе будем иметь их классификацию.
5. Обнаружение вторжений. При обнаружении вторжения происходит реагирование на угрозу (например, блокировка сетевого траффика и др.) и генерация сигнала пользователю.
6. Адаптация нейросетевых агентов. При обнаружении вторжения происходит его классификация. Если обнаруженное вторжение является новым, неизвестным раньше, то происходит выделение параметров такого вторжения (сигнатура) и обновление учебной выборки путем занесения сигнатуры новой угрозы в базу для обучения новых иммунных агентов.
7. Создание иммунной памяти. При обнаружении нового вторжения создается «специальный» агент, так называемый агент иммунной памяти. При обучении агента иммунной памяти используется выделенная сигнатура обнаруженной угрозы. В результате формируется агент, который реагирует на данную конкретную угрозу.
8. Уничтожение нейросетевых агентов по окончании времени жизни. Каждому наученному и отобранному агенту выделяется определенное время, которое называется временем «жизнь», в течение которого он функционирует в системе. По окончании времени жизни агент уничтожается, а на его место приходит новый. Такой подход обеспечивает постоянный прилив новых потенциально более сильных агентов в систему защиты.
9. Обмен информацией между агентами. Однотипные нейросетевые агенты, то есть агенты, обученные на определенный тип вторжения, должны обмениваться информацией.

Результаты проведенных экспериментов по исследованию обобщающего свойства нейросетевых иммунных агентов показали, что обученные детекторы обнаруживают и классифицируют не только атаки, на которых детекторы учились, но и атаки других типов. Достоверность обнаружения и классификации неизвестных атак, в отдельных случаях, может достигать 100% (таблица 1).

Проведены экспериментальные исследования адаптации нейросетевых иммунных агентов к неизвестным атакам в результате операции клонирования и мутации. Для этого в качестве родительского агента выбран агент 2, который учился на атаке типа DoS_land. Данный агент обнаружил неизвестные для него атаки – R2L_imap и Probe_portsweep. Допустим, что в базе данных такие атаки отсутствуют, сгенерируем два новых агента A1 и A2, добавим параметры обнаруженных атак в учебные выборки для этих агентов и научим соответствующие агенты-клоны (таблица 2).

***Таблица 1 –*** Результаты обобщающего свойства нейросетевых иммунных агентов

| Тип атаки | Агент 1 (обученный на *DoS_back*) % | Агент 2 (обученный на *Probe_portsweep*) % | Агент 3 (обученный на *R2L_ftpwrite*) % |
|---|---|---|---|
| *DoS-атаки* | | | |
| *Back* | 100,0 | 0,0 | 0,3 |
| *Land* | 0,0 | 100 | 23,8 |
| *Neptune* | 99,1 | 100 | 0,0 |
| *Probe-атаки* | | | |
| *Portsweep* | 2,1 | 100 | 0,1 |
| *Satan* | 13,3 | 92,2 | 2,1 |
| *R2L-атаки* | | | |
| *Spy* | 100,0 | 0,0 | 0,0 |
| *Ftp_write* | 0,0 | 0,0 | 100 |
| *U2R-атаки* | | | |
| *Loadmodule* | 88,9 | 0,0 | 0,0 |

***Таблица 2 –*** Адаптация нейросетевых иммунных агентов к неизвестным атакам

| Тип атаки | Агент 2 Sp(TNR)= 99,0% | Агент А1 Sp(TNR)= 99,1% | Агент А2 Sp(TNR)= 98,9% |
|---|---|---|---|
| | *Se (TPR),%* | *Se (TPR),%* | *Se (TPR),%* |
| *DoS-атаки* | | | |
| *Land* | 100,0 | 100,0 | 100,0 |
| *Pod* | 2,3 | 0,0 | 31,8 |
| *Probe-атаки* | | | |
| *Ipsweep* | 7,22 | 0,2 | 33,9 |
| *Portsweep* | 15,9 | 2,6 | 55,3 |
| *Satan* | 11,0 | 31,3 | 11,0 |
| *R2L-атаки* | | | |
| *Imap* | 83,3 | 91,7 | 83,3 |
| *Multihop* | 0,0 | 0,0 | 14,3 |
| *U2R-атаки* | | | |
| *Perl* | 0,0 | 66,7 | 0,0 |
| *Rootkit* | 0,0 | 20,0 | 0,0 |

Из таблицы 2 видно, что агент А1 начал лучше обнаруживать атаки отдельных классов, в частности Probe_satan – в 2,8 раза, а R2L_imap на 8,4%, а также он начал обнаруживать атаки U2R_perl и R2L_rootkit. В свою очередь агент А2 показал лучшие результаты на атаках Probe_ipsweep, DoS_pod, Probe_portsweep, R2L_multihop.

При проведении экспериментальных исследований размер учебной выборки составил 80 векторов (64 – атака одного из типов и 16 – нормальные соединения), тестовой выборки – множественное число записей базы KDD в разрезе типов, структура сети LVQ – 12-10-2.

Результаты оценки достоверности с использованием ROC-анализа рассмотрены на примере одного нейросетевого иммунного агента, обученного на атаке DoS_neptune (таблица 3).

ROC-кривые, которые отображают способность агента 1 обнаруживать и классифицировать сетевые атаки определенных типов, приведены на рисунке 2.

***Таблица 3 –*** Характеристики нейросетевого иммунного агента 1

| Агент 1 обучен на *DoS_neptune*. Sp (TNR) = 99,9%, FPR = 0,1% | | | |
|---|---|---|---|
| Тип атаки | *Se (TPR),%* | *FNR,%* | *Accu,%* |
| *DoS-атаки* | | | |
| *Land* | 100,0 | 0,0 | 100,0 |
| *Neptune* | 100,0 | 0,0 | 100,0 |
| *Teardrop* | 3,7 | 96,3 | 51,8 |
| *Probe-атаки* | | | |
| *Ipsweep* | 6,5 | 93,5 | 53,2 |
| *Portsweep* | 98,9 | 1,1 | 99,4 |
| *Satan* | 90,0 | 10,0 | 95,0 |

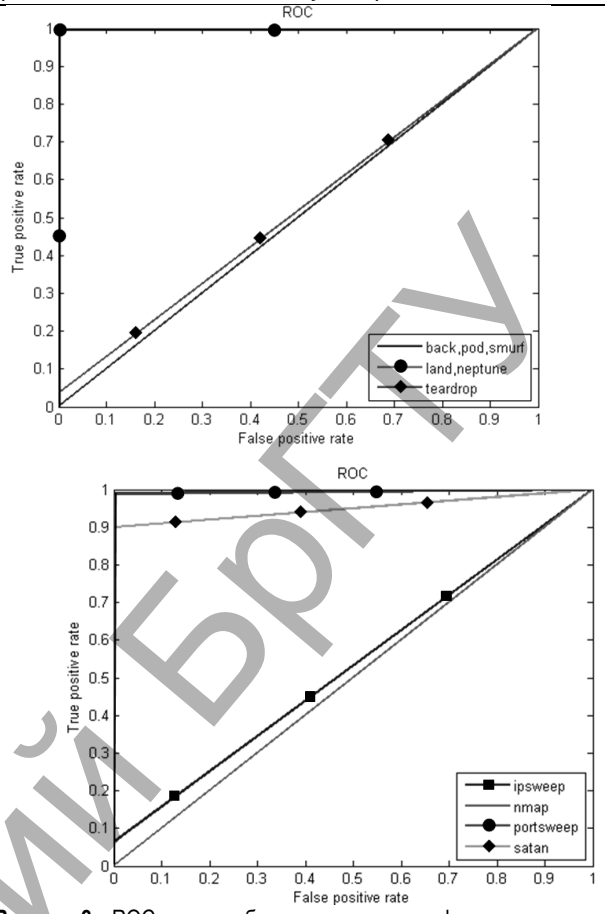


***Рисунок 2 –*** ROC-кривые обнаружения и классификации атак нейросетевым иммунным агентом 1

Как видно из таблицы 3 и рисунка 2, достоверность обнаружения и классификации атак типа DoS_neptune агентом 1 составила 100% при уровне ошибок второго рода (FPR) – 0,1%. Кроме этого, достоверность обнаружения и классификации атак DoS_land составила 100%, Probe_portsweep – 98,9%, Probe_satan – 90,0% и Probe_ipsweep – 6,5%.

Достоверность обнаружения и классификации сетевых атак совокупностью нейросетевых иммунных агентов представлена в таблице 4.

**Заключение.** Предложен подход к улучшению уровня безопасности системы обнаружения вторжений за счет реализации нейросетевых агентов на программируемых логических интегральных схемах и введения подсистемы принятия решений на основе правил нечеткого вывода Мамдани. Повышение стойкости системы обнаружения вторжений достигается:

1. Интеграцией нейросетевых агентов в искусственную иммунную систему, что позволило им адаптироваться к неизвестным вторжениям за счет осуществления операций клонирования и мутации, а также повысить достоверность обнаружения и классификации неизвестных угроз.
2. Обеспечением четкого обмена данными и четкой координации между подсистемой принятия решений, нейросетевыми агентами, подсистемой управления и планирования и исполнительной подсистемой.
3. Выполнением части узлов интеллектуальной системы защиты от вторжений на программируемых логических интегральных схемах, перепрограммирование которых осуществляет компьютер, который не контактирует с сетью, из которой возможно поступление угроз. Такое решение, в отличие от программной защиты, исключает влияние вторжений на систему защиты.
4. Компьютерное моделирование подтвердило, что разработанная интеллектуальная система обнаружения вторжений является более стойкой как к известным вторжениям, так и новым.

***Таблица 4 –*** Достоверность обнаружения и классификации сетевых атак

| d_back | d_land | d_neptune | d_pod | d_smurf | d_teardrop | p_ipsweep | p_nmap |
|---|---|---|---|---|---|---|---|
| 100 | 100 | 100 | 100 | 100 | 100 | 99,90 | 100 |
| p_prtsweep | p_satan | r_ftpwrite | r_gpasswd | r_imap | r_multihop | r_phf | r_spy |
| 99,99 | 99,99 | 100 | 99,85 | 99,89 | 99,30 | 100 | 100 |
| r_wclient | r_wmaster | u_overflw | u_ldmodul | u_perl | u_rootkit | | |
| 99,50 | 99,60 | 99,20 | 100 | 100 | 98,60 | | |

**СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ**


1. Bezobrazov, S. Artificial immune system approach for malware detection: neural networks applying for immune detectors construction / S. Bezobrazov, V. Golovko // Inernational journal of «Computing». – 2008. Vol. 7, №. 2. – P. 44–50.
2. Kachurka, P. Fusion Of Recirculation Neural Networks For Real-time Network Intrusion Detection And Recognition / P. Kachurka, V. Golovko // International Journal of Computing. – 2012. – Vol. 11, № 4. – P. 383–390.
3. Kotenko, I. Multi-agent Simulation of Attacks and Defense Mechanisms in Computer Networks // International Journal of «Computing». – 2008. – Vol. 7, № 2. – P. 35–43.
4. Computationally intelligent agents for distributed intrusion detection system and method of practicing same / A.H. Sung, S. Mukkamala, J.L. Lassez – US Patent 7,941,855, 2011.
5. Wee, Y.Y. Causal Discovery and Reasoning for Intrusion Detection using Bayesian Network / Y.Y. Wee, W.P. Cheah, S.C. Tan // International Journal of Machine Learning and Computing. – 2011. – Vol. 1, № 2. – P. 185–192.
6. Wang, G. A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering / G. Wang, J. Hao, J. Ma, L. Huang // Expert Systems with Applications. – 2010. – T. 37. – №. 9. – C. 6225–6232.
7. Banković, Z. Distributed intrusion detection system for wireless sensor networks based on a reputation system coupled with kernel self-organizing maps. Integrated Computer-Aided Engineering / Z. Banković, J.M. Moya, Á. Araujo [et. all.] – 2010, Volume 17, Issue 2. – P. 87–102.
8. Laheeb, M.I. Anomaly network intrusion detection system based on distributed time-delay neural network (DTDNN) // Journal of Engineering Science and Technology. – 2010. – № 5 (4). – P. 457–471.
9. Farid, D.M. Anomaly Network Intrusion Detection Based on Improved Self Adaptive Bayesian Algorithm / D.M. Farid, M.Z. Rahman // Journal of Computers. – 2010. – Vol. 5, № 1. – P. 23–31.
10. Bonabeau, Eric. Swarm Intelligence: From Natural to Artificial Systems / Eric Bonabeau, Marco Dorigo, Guy Therauaz. – NY: Oxford University Press Inc., 1999. – 306 p.
11. Haykin, S. Neural Networks and Learning Machines / S. Haykin. – Prentice Hall. – 2009. – 906 c.
12. Kohonen, T. The self organizing map / T. Kohonen // Proceedings oft he Institute of Electrical and E1ectronics Engineers. – 1990. – Vol. 78. – P. 1464–1480.
13. Komar, M. Intelligent system for detection of networking intrusion / M. Komar, V. Golovko, A. Sachenko, S. Bezobrazov // Intelligent Data Acquisition and Advanced Computing Systems : Technology and Applications : Proceedings of the 6th IEEE International Conference, Prague (Czech Republic), September 15-17, 2011. – V1. – P. 374–377.
14. Sachenko, A. Intrusion detection system based on neural networks / A. Sachenko, M. Komar / Scientific Papers of Silesian University of Technology. Organization and Management Series. – Gliwice (Poland), 2014. – Vol. 68. – P. 377–386.
15. Pat. Number 109640 Ukraine, IPC (2012) H04W 12/08, G06F 21/00, G06F 12/14. Method of detection of computer attacks by the neural network artificial immune system / M. Komar, A. Sachenko, V. Golovko, S. Bezobrazov // Patent holder Ternopil National Economic University. – № a201205350; appl. 28.04.12, publ. 25.09.15, № 18 (In Ukrainian).
16. Komar, M. Development of Neural Network Immune Detectors for Computer Attacks Recognition and Classification / M. Komar, V. Golovko, A. Sachenko, S. Bezobrazov // Intelligent Data Acquisition and Advanced Computing Systems : Technology and Applications : Proceedings of the 7th IEEE International Conference, Berlin (Germany), September 12–14, 2013. – V2. – P. 665–668.
17. Hofmeyr, S. Immunity by design: An artificial immune system / S. Hofmeyr, S. Forrest // Gecco. – 1999. – Vol. 2. – P. 1289–1296.
18. Dasgupta, D. Recent advances in artificial immune systems: models and applications / D. Dasgupta, S. Yu, F. Nino // Applied Soft Computing. – 2011. – T. 11. – №. 2. – P. 1574–1587.
19. Aickelin, U. Artificial immune systems – Intros / U. Aickelin, D. Dasgupta, Gu F. Search // Methodologies, Springer, 2014. – Chapter 7. – P. 187–211.
20. Katia, P. Sycara. Multiagent systems // AI Magazine. – 1998. – Vol. 10. – No. 2. – P. 79–93.
21. Wooldridge, M. Intelligent Agents: Theory and Practice / M. Wooldridge, N. Jennings // Knowledge Engineering Review. – June 1995. – Vol.10. – № 2. – P.115-152.
22. Savenko, O. Multi-agent based approach of botnet detection in computer systems / O. Savenko, S. Lysenko, A. Kryschuk // Computer Networks Communications in Computer and Information Science. – 2012. – Vol. 291. – P. 171–180.
23. Зотов, В. Проектирование цифровых устройств на основе ПЛИС фирмы Xilinx в САПР WebPack ISE. – Москва : Горячая линия – Телеком, 2003. – 624 с.
24. Штовба, С.Д. Обеспечение точности и прозрачности нечеткой модели Мамдани при обучении по экспериментальным данным / С.Д. Штовба // Проблемы управления и информатики. – 2007. – №4. – С. 102–114.
25. Shtovba, S.D. Ensuring Accuracy and Transparency of Mamdani Fuzzy Model in Learning by Experimental Data / S.D. Shtovba // Journal of Automation and Information Sciences. – 2007. – № 39 – P. 39–52.
26. Dubchak, L. Fuzzy Data Processing Method / L. Dubchak, N. Vasylkiv, V. Kochan, A. Lyapandra // Intelligent Data Acquisition and Advanced Computing Systems : Technology and Applications : Proceedings of the 7th IEEE International Conference, Berlin (Germany), September 12-14, 2013. – V 1. – P. 373–375.
27. Дубчак, Л.О. Средство ускоренной обработки нечетких данных на основе механизма Мамдани / Л.О. Дубчак, В.В. Кочан, Н.М. Василькив // Вестник Брестского государственного технического университета. Серия физика, математика, информатика. – 2016. – № 5. – С. 23–26.
28. Dubchak, L. Speedy procesing method of fuzzy data for intelligent systems of intrusion detection / L.Dubchak, M. Komar, A. Sachenko, V. Kochan // Projekt interdyscyplinarny projektem XXI wieku. – Bielsko-Biala, 2017. – Tom 2: Processing, transmission and security of information. – S. 65–74.
29. Jolliffe, I. Principal component analysis // Springer, 2010. – 516 p.
30. Shilpa, Lakhina. Feature Reduction using Principal Component Analysis for Effective Anomaly–Based Intrusion Detection on NSL-KDD // International Journal of Engineering Science and Technology. – 2010. – Vol. 2, № 6. – P. 1790–1799.


*Материал поступил в редакцию 08.01.2018*

***KOMAR M.P.*** **Improving of the security of intrusion detection systems**

Agent-oriented hardware and software are proposed to use in order to increase the security of intrusion detection system. A generalized structure of intrusion detection was developed. A neural network was used as its main agent. Agents as neuron immune detectors for detection and classification of intrusions were implemented on the programmable logic arrays. Mamdani fuzzy inference rules were offered to use in order to counteract intrusions. A structure of an appropriate subsystem of decision-making was developed.