ОШ МАМЛЕКЕТТИК УНИВЕРСИТЕТИНИН ЖАРЧЫСЫ

ВЕСТНИК ОШСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

BULLETIN OF OSH STATE UNIVERSITY

ISSN 1694-7452 e-ISSN: 1694-8610

№2/2026, 448-460

ТЕХНИКА

УДК: 004.056:004.8

DOI: 10.52754/16948610_2026_2_33

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ ОСНОВЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ МОДЕЛИРОВАНИЯ И ПРОГНОЗИРОВАНИЯ УГРОЗ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ НА ОСНОВЕ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

ЖАСАЛМА ИНТЕЛЛЕКТТИН ЫКМАЛАРЫНА НЕГИЗДЕЛГЕН МААЛЫМАТТЫК КООПСУЗДУК КОРКУНУЧТАРЫН МОДЕЛДӨӨ ЖАНА БОЛЖОЛДОО ҮЧҮН ПРОГРАММАЛЫК КАМСЫЗДООНУН МАТЕМАТИКАЛЫК НЕГИЗДЕРИН ИШТЕП ЧЫГУУ

DEVELOPMENT OF A MATHEMATICAL FRAMEWORK FOR SOFTWARE SUPPORTING MODELING AND PREDICTION OF INFORMATION SECURITY THREATS BASED ON ARTIFICIAL INTELLIGENCE METHODS

Асилбеков Тынчтыкбек Майрамбекович
Асилбеков Тынчтыкбек Майрамбекович
Asilbekov Tynchtykbek Mairambekovich

преподаватель, Ошский государственный университет
окутуучу, Ош мамлекеттик университети
teacher, Osh State University
mir.titan.90@gmail.com
ORCID: 0009-0002-4292-1580

Орозов Максатбек Омурбекович
Орозов Максатбек Омурбекович
Orozov Maksatbek Omurbekovich

к.ф.-м.н., доцент, Ошский государственный университет
ф.-м.и.к.,, доцент, Ош мамлекеттик университети
Candidate of Physical and Mathematical Sciences, Associate Professor, Osh State University
orozov@oshsu.kg

Асанов Азизбек Кыпчакович
Асанов Азизбек Кыпчакович
Asanov Azizbek Kypchakovich

преподаватель, Ошский государственный университет
окутуучу, Ош мамлекеттик университети
lecturer, Osh State University
aasanov@oshsu.kg
ORCID: 0009-0008-8314-3475

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ ОСНОВЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ МОДЕЛИРОВАНИЯ И ПРОГНОЗИРОВАНИЯ УГРОЗ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ НА ОСНОВЕ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Аннотация

Актуальность. В статье рассматривается научно-методологическая основа для будущей разработки программного обеспечения, предназначенного для моделирования, раннего выявления и прогнозирования угроз информационной безопасности на основе методов машинного обучения, глубокого обучения, графового анализа и объяснимого искусственного интеллекта. Актуальность исследования обусловлена ростом сложности кибератак, распространением многоэтапных APT-кампаний, увеличением числа атак на облачные, корпоративные и IoT-инфраструктуры, а также ограниченностью классических сигнатурных систем обнаружения вторжений при работе с новыми, модифицированными и заранее неизвестными угрозами. В отличие от подходов, ориентированных только на фиксацию уже произошедшего инцидента, в работе обосновывается необходимость перехода к прогнозной модели информационной безопасности, способной учитывать временную последовательность событий, связи между узлами сети, вероятностный характер развития атаки, изменение сетевой топологии и необходимость объяснения решений искусственного интеллекта для аналитика SOC. Предлагается математическая постановка задачи, включающая описание сетевого события как вектора признаков, представление инфраструктуры в виде динамического графа, формирование функции риска, построение вероятностного прогноза атаки на заданном временном горизонте и определение критериев качества будущей программной реализации. Практическая значимость исследования заключается в том, что сформированная теоретическая база может быть использована как фундамент для последующей разработки интеллектуального программного комплекса, интегрируемого с SIEM/SOC-инфраструктурой и предназначенного для анализа событий информационной безопасности, оценки риска, прогнозирования угроз и объяснения причин тревожного вывода.

Ключевые слова: информационная безопасность; искусственный интеллект; машинное обучение; прогнозирование угроз; математическое моделирование

Жасалма интеллекттин ыкмаларына негизделген маалыматтык коопсуздук коркунучтарын моделдөө жана болжолдоо үчүн программалык камсыздоонун математикалык негиздерин иштеп чыгуу

Аннотация

Макалада машиналык окутуу, терең окутуу, графтык талдоо жана түшүндүрмөлүү жасалма интеллект ыкмаларына негизделген маалыматтык коопсуздук коркунучтарын моделдөө, эрте аныктоо жана болжолдоо үчүн иштелип чыгуучу программалык камсыздоонун илимий-методологиялык негизи каралат. Изилдөөнүн актуалдуулугу киберчабуулдардын татаалдашуусу, көп баскычтуу APT-чабуулдардын кеңири жайылышы, булуттук, корпоративдик жана IoT-инфраструктураларга багытталган чабуулдардын көбөйүшү, ошондой эле жаңы, өзгөртүлгөн жана мурда белгисиз болгон коркунучтарды аныктоодо салттуу сигнатуралык кийлигишүүнү аныктоо системаларынын мүмкүнчүлүктөрүнүн чектелгендиги менен шартталган. Макалада буга чейин болуп өткөн инциденттерди гана каттоого багытталган ыкмалардан айырмаланып, окуялардын убакыттык ырааттуулугун, тармак түйүндөрүнүн өз ара

Development of a mathematical framework for software supporting modeling and prediction of information security threats based on artificial intelligence methods

Abstract

This paper presents the scientific and methodological foundation for the future development of software designed to model, detect at an early stage, and predict information security threats using machine learning, deep learning, graph analysis, and explainable artificial intelligence techniques. The relevance of the study is driven by the increasing complexity of cyberattacks, the growing prevalence of multi-stage Advanced Persistent Threat (APT) campaigns, the rising number of attacks targeting cloud, corporate, and IoT infrastructures, as well as the limited effectiveness of traditional signature-based intrusion detection systems against new, modified, and previously unknown threats. Unlike conventional approaches focused solely on detecting incidents after they occur, the proposed framework justifies the transition toward a predictive information security model capable of considering the temporal sequence of events, relationships among network nodes, the probabilistic evolution of attacks, changes in network topology, and the need to

байланышын, чабуулдардын өнүгүүсүнүн ыктымалдык мүнөзүн, тармактык топологиянын өзгөрүшүн жана SOC аналитиги үчүн жасалма интеллекттин чечимдерин түшүндүрүүнүн зарылдыгын эске алган маалыматтык коопсуздуктун болжолдоочу моделине өтүүнүн негиздемеси сунушталат. Маселенин математикалык коюлушу сунушталып, анда тармактык окуя белгилер вектору түрүндө сүрөттөлөт, инфраструктура динамикалык граф катары берилет, тобокелдик функциясы түзүлөт, белгилүү убакыт аралыгына чабуулдун ыктымалдык божомолу иштелип чыгат жана келечектеги программалык ишке ашыруунун сапат критерийлери аныкталат. Изилдөөнүн практикалык мааниси түзүлгөн теориялык негиз келечекте SIEM/SOC-инфраструктурасы менен интеграцияланууучу, маалыматтык коопсуздук окуяларын талдоого, тобокелдиктерди баалоого, коркунучтарды болжолдоого жана кооптуу эскертүүлөрдүн себептерин түшүндүрүүгө арналган интеллектуалдык программалык комплексти иштеп чыгуу үчүн негиз боло ала тургандыгында.

Ачкыч сөздөр: маалыматтык коопсуздук; жасалма интеллект; машиналык окутуу; коркунучтарды болжолдоо; математикалык моделдөө

provide interpretable AI decisions for Security Operations Center (SOC) analysts. The paper formulates the mathematical basis of the problem, including the representation of network events as feature vectors, modeling the infrastructure as a dynamic graph, defining a risk function, constructing probabilistic attack forecasts over a specified time horizon, and establishing quality criteria for the future software implementation. The practical significance of the study lies in the fact that the proposed theoretical framework can serve as a foundation for developing an intelligent software system integrated with SIEM/SOC infrastructures for information security event analysis, risk assessment, threat prediction, and explanation of security alerts.

Keywords: information security; artificial intelligence; machine learning; threat prediction; mathematical modeling

Введение

Развитие цифровых технологий, облачных сервисов, корпоративных сетей, IoT-инфраструктуры и государственных информационных систем приводит к постоянному увеличению количества цифровых активов, сетевых соединений, пользователей, сервисов и событий информационной безопасности. Вместе с ростом цифровизации возрастает и сложность киберугроз. Современный злоумышленник всё чаще действует не как одиночный источник вредоносного трафика, а как субъект, выполняющий последовательность скрытых действий: разведку, получение первичного доступа, закрепление в системе, повышение привилегий, перемещение внутри сети и последующую эксфильтрацию данных. Такой характер атак особенно выражен в APT-кампаниях, где отдельные действия могут не выглядеть критичными, но их совокупность во времени указывает на развитие инцидента (ENISA, 2025; Verizon, 2026; IBM Security, 2025; Anderson J.P., 1980).

Актуальные отчёты по киберугрозам подтверждают, что проблема раннего выявления и прогнозирования атак сохраняет высокую значимость. ENISA Threat Landscape 2025 анализирует 4875 инцидентов за период с 1 июля 2024 года по 30 июня 2025 года и подчёркивает сложность современной киберэкосистемы, в которой угрозы быстро адаптируются к изменениям цифровой среды (ENISA, 2025). Отчёт Verizon DBIR 2026 показывает, что 31% нарушений безопасности начинается с эксплуатации программных уязвимостей, что указывает на необходимость более раннего обнаружения цепочек атак, а не только фиксации конечного инцидента (Verizon, 2026). IBM Cost of a Data Breach Report 2025 фиксирует среднюю мировую стоимость утечки данных на уровне 4,44 млн долларов США, что дополнительно подчёркивает экономическую значимость задач обнаружения, прогнозирования и реагирования на инциденты (IBM Security, 2025). Следовательно, разработка интеллектуальных средств анализа угроз является не только технической, но и организационно-экономической задачей.

Классические системы защиты информации, такие как IDS, IPS, антивирусные средства и SIEM-системы, традиционно используют сигнатуры, корреляционные правила и пороговые значения. Такие подходы обладают высокой скоростью работы и хорошо подходят для обнаружения известных угроз. Однако они имеют принципиальное ограничение: если атака является новой, модифицированной, распределённой или растянутой во времени, заранее заданное правило может не сработать. Данная проблема была отмечена ещё в ранних работах по обнаружению вторжений, начиная с моделей мониторинга безопасности Anderson (Anderson J.P., 1980) и Denning (Denning D.E., 1987), а позднее получила развитие в исследованиях по машинному обучению для IDS (Tavallaee M., 1987; Buczak A.L.,2016; Guven E. A, 2016; Sharafaldin I., 2018).

Переход к машинному обучению позволил анализировать не только известные сигнатуры, но и статистические закономерности поведения. В работах Tavallaee et al., Buczak и Guven , Sharafaldin et al. , Lashkari et al. (Lashkari A.H.,2017), Ferrag et al. (Ferrag M.A.,2022) показано, что методы анализа данных могут эффективно применяться к задачам обнаружения сетевых атак. При этом используются признаки сетевых потоков: длительность соединения, число пакетов, объём переданных байтов, порты, протоколы, TCP-флаги, частота обращений, статистика временных окон и другие параметры. Однако классические методы машинного обучения, такие как Random Forest, SVM, Logistic Regression, XGBoost и CatBoost, чаще всего рассматривают событие как отдельный вектор признаков и не всегда учитывают долгосрочные

временные зависимости и топологию сети (Buczak A.L.,2016; Ferrag M.A., 2022; Breiman L., 2001; Chen T., 2016).

Для описания сетевого события введём вектор признаков:

Формула 1:

$x(t) = [x1(t), x2(t), ..., xd(t)]$

где x(t) - событие или сетевой поток в момент времени t;
d - количество признаков;
x1(t), x2(t), ..., xd(t) - числовые характеристики события.

Например, в такой вектор могут входить: длительность соединения, количество пакетов, число байтов, номер порта, протокол, количество ошибок авторизации, частота запросов, среднее время между пакетами, число SYN-флагов, энтропия полезной нагрузки и другие параметры. Если рассматривать только один такой вектор, можно выполнить классификацию события. Но для прогнозирования угрозы этого недостаточно, потому что атака часто проявляется не в одном событии, а в последовательности.

Поэтому поток событий информационной безопасности целесообразно представить, как временную последовательность:

Формула 2:

$X = \{x(1), x(2), ..., x(T)\}$

где X - последовательность событий;
T - длина наблюдаемого временного окна;
x(t) - событие в момент времени t.

В этом случае задача системы заключается не только в ответе на вопрос «является ли текущее событие атакой?», но и в ответе на вопрос «какова вероятность того, что в ближайшем будущем текущая последовательность событий приведёт к атаке?». Такая постановка переводит задачу из области простого обнаружения в область прогнозирования.

Вероятностную задачу прогнозирования можно записать следующим образом:

Формула 3:

$P(y(t + h) = attack \mid x(1), x(2), ..., x(t))$

где P - вероятность;
y(t + h) - состояние системы через горизонт прогнозирования h;
h - временной горизонт прогноза;
attack - класс атаки;
x(1), ..., x(t) - уже наблюдаемая последовательность событий.

Смысл этой формулы простой: имея историю событий до текущего момента, модель должна оценить вероятность того, что через определённое время возникнет атака. Например, система может прогнозировать риск атаки через 1 минуту, 5 минут, 30 минут или другой заданный интервал. Именно такая постановка является более ценной для SOC, поскольку она позволяет не только реагировать на уже произошедший инцидент, но и заранее предупреждать аналитика.

В научной литературе для анализа последовательностей применяются рекуррентные нейронные сети, LSTM, GRU, временные сверточные сети и Transformer-архитектуры (Hochreiter S.,1997; Kim J.,2016; Staudemeyer R.C., 2015; Vaswani A. 2017; Yu W., 2021; Brown T.B., 2020). LSTM хорошо подходит для обработки последовательностей, однако при больших потоках сетевых событий может иметь высокую вычислительную стоимость. Transformer-модели эффективно выявляют зависимости между элементами последовательности, но

механизм self-attention имеет квадратичную сложность относительно длины входа, что ограничивает их применение в режиме реального времени при больших потоках данных. Поэтому для будущей программной реализации целесообразно рассмотреть Temporal Convolutional Network, поскольку TCN позволяет обрабатывать временные последовательности параллельно и учитывать историю событий через расширенные причинные свёртки (Bai S.,2018).

Обобщённо преобразование временной последовательности в TCN можно представить так:

Формула 4:

$z(t) = TCN(x(t), x(t-1), ..., x(t-k))$

где z(t) - скрытое представление текущего состояния;
TCN - временная сверточная модель;
k - глубина исторического окна;
x(t-k), ..., x(t) - события, учитываемые моделью.

Смысл данной формулы заключается в том, что модель получает не одно событие, а фрагмент истории. Например, если в течение некоторого времени наблюдаются сканирование портов, неудачные попытки входа, обращение к административному интерфейсу и рост исходящего трафика, TCN может выделить этот фрагмент как подозрительную временную закономерность.

Однако временного анализа недостаточно. Современная информационная система имеет сетевую структуру. Узлы, пользователи, сервисы, IP-адреса, приложения и устройства взаимодействуют друг с другом. Поэтому инфраструктуру целесообразно представить в виде графа:

Формула 5:

$G(t) = (V(t), E(t), A(t))$

где G(t) - граф сети в момент времени t;
V(t) - множество узлов сети;
E(t) - множество связей между узлами;
A(t) - матрица смежности или матрица связей.

Узлами графа могут быть серверы, рабочие станции, пользователи, IP-адреса, контейнеры, сервисы или IoT-устройства. Рёбрами могут быть сетевые соединения, обращения к сервисам, попытки авторизации, передача файлов, DNS-запросы или другие взаимодействия. Такой подход соответствует современной логике кибератак, поскольку злоумышленник редко действует изолированно: он перемещается между узлами, использует связи, меняет маршруты и постепенно расширяет область воздействия.

Графовые нейронные сети позволяют учитывать не только признаки отдельного узла, но и его окружение. Основы графовых сверточных сетей были заложены в работе Kipf и Welling (Kipf T.N.,2017), а механизм графового внимания получил развитие в GAT-модели Veličković et al. (Veličković P., 2018). В контексте сетевого обнаружения вторжений значимыми являются работы E-GraphSAGE и Anomal-E, где сетевые потоки рассматриваются как графовые структуры (Lo W.W.,2022; Caville E.,2022). E-GraphSAGE показывает, что flow-based NIDS естественно представимы в графовой форме, поскольку сетевые потоки отражают взаимодействия между узлами. Это делает GNN-подход перспективным для будущей разработки интеллектуальной системы прогнозирования угроз.

Обновление признака узла в графовой модели можно представить так:

Формула 6:

$$h'_v = \sigma\left(W \cdot h_v + \sum_{u \in N(v)} \alpha_{vu} \cdot W \cdot h_u \right)$$

v	—	Индекс целевого (обновляемого) узла графа
u	—	Индекс соседнего узла, принадлежащего множеству N(v)
N(v)	—	Множество всех соседей узла v (в некоторых реализациях включает сам узел v для учета собственных признаков)
$\mathbf{h}_v$	R^d	Вектор признаков (эмбеддинг) узла v до обновления; d — размерность входного пространства
$\mathbf{h}_u$	R^d	Вектор признаков соседнего узла u до обновления
h'_v	$R^{d'}$	Вектор признаков узла v после обновления; d' — размерность выходного пространства
$\mathbf{W}$	$R^{d'} \times d$	Обучаемая матрица весов, выполняющая линейное преобразование признаков всех узлов
$\mathbf{W} \cdot \mathbf{h}_v$	$R^{d'}$	Линейно преобразованные признаки самого узла v
$\mathbf{W} \cdot \mathbf{h}_u$	$R^{d'}$	Линейно преобразованные признаки соседнего узла u
α_{vu}	R	Скалярный коэффициент внимания, определяющий степень важности соседа u для узла v
$\sigma(\cdot)$	$R^{d'} \to R^{d'}$	Нелинейная функция активации (ReLU, ELU, Sigmoid, Tanh и др.), применяемая поэлементно к вектору

Смысл формулы заключается в следующем: состояние узла оценивается не только по его собственным признакам, но и по поведению связанных с ним узлов. Например, если один внутренний сервер начинает обмениваться данными с подозрительным внешним адресом, а затем несколько рабочих станций начинают обращаться к этому серверу с нетипичной частотой, графовая модель может выявить изменение риска на уровне всей подсети.

Коэффициент внимания можно записать так:

Формула 7:

$$\alpha_{vu} = \mathrm{softmax}\big(\mathrm{score}(\mathbf{h}_v, \mathbf{h}_u)\big)$$

где α_{vu} показывает, насколько сильно узел u влияет на оценку узла v. Если связь между двумя узлами важна для выявления атаки, коэффициент внимания будет выше.

Для построения будущей модели прогнозирования целесообразно объединить временную и графовую компоненты:

Формула 8:

$$r(t+h) = F\big(\mathrm{TCN}(\mathbf{X}_t), \mathrm{GNN}(\mathbf{G}_t)\big)$$

где r(t + h) - прогнозируемый риск атаки через горизонт h;
TCN(Xt) - временное представление последовательности событий;
GNN(Gt) - графовое представление сетевой инфраструктуры;
F - объединяющая функция или нейронный слой принятия решения.

Эта формула отражает основную идею будущего программного обеспечения: риск атаки должен рассчитываться не только по одному событию, а на основе истории событий и структуры сети. Именно такая комбинация может стать фундаментом интеллектуального модуля для SIEM/SOC-системы.

Для обучения такой модели необходимо определить функцию потерь. Если задача является многоклассовой классификацией атак, можно использовать кросс-энтропию:

Формула 9:

$$L_{class} = -\sum_{k} y_k \cdot \log(p_k)$$

где y_k - истинная метка класса;

p_k - предсказанная моделью вероятность класса k.

Если же задача рассматривается как прогноз риска, можно использовать ошибку между фактическим и прогнозируемым риском:

Формула 10:

$$L_{risk} = (r_{real} - r_{pred})^2$$

где r_{real}- фактический уровень риска;

r_{pred} - прогнозируемый уровень риска.

Для учёта дисбаланса классов можно использовать взвешенную функцию потерь:

Формула 11:

$$L_{weighted} = -\sum_{k} w_k \cdot y_k \cdot \log(p_k)$$

где w_k - вес класса k.

Чем реже встречается класс атаки, тем выше может быть его вес.

Это особенно важно для информационной безопасности, поскольку нормальных событий обычно намного больше, чем атакующих. Если не учитывать дисбаланс, модель может научиться хорошо распознавать нормальный трафик, но плохо обнаруживать редкие атаки.

Дополнительно необходимо учитывать устойчивость к состязательным воздействиям. В задачах компьютерного зрения широко известны FGSM, PGD и другие методы состязательных атак (Koroniotis N.,2019; Moustafa N., 2020). Однако в сетевой безопасности такие методы нельзя применять напрямую, поскольку изменение признаков должно сохранять физический и логический смысл сетевого потока. Например, нельзя произвольно изменить количество пакетов, флаги протокола и длительность соединения так, чтобы получился невозможный с точки зрения сети объект. Поэтому в будущей реализации необходимо использовать ограниченные состязательные возмущения:

Формула 12:

$$\mathbf{x}_{adv} = \mathbf{x} + \boldsymbol{\delta} \text{ при условиях: } \| \boldsymbol{\delta} \|_p \leq \varepsilon$$

$$\mathbf{x}_{adv} \in D_{valid}$$

где $\mathbf{x}_{adv}$ - изменённый злоумышленником пример;

δ - малое возмущение признаков;

ε - максимальный размер возмущения;

D_{valid} - область допустимых сетевых признаков.

Смысл данной формулы заключается в том, что атакующий может попытаться изменить признаки трафика, но эти изменения должны оставаться реалистичными. Поэтому будущая модель должна проверяться не только на обычной тестовой выборке, но и на устойчивость к допустимым состязательным изменениям.

Важной частью будущей системы является объяснимость. Методы объяснимого искусственного интеллекта, такие как SHAP, LIME, DeepLIFT и LRP, позволяют показать, какие признаки сильнее всего повлияли на решение модели (Goodfellow I.J.,2015; Madry A.,2018; Warnecke A.,2020; Lundberg S.M.,2017). Для аналитика SOC это имеет практическое значение. Если система сообщает только «атака вероятна», этого недостаточно. Необходимо объяснить, почему: например, из-за роста числа SYN-запросов, нетипичного порта, резкого увеличения исходящего трафика, необычной связи между узлами или последовательности неудачных авторизаций.

В общем виде вклад признака можно представить так:

Формула 13:

$f(x) = \varphi 0 + \varphi 1 + \varphi 2 + ... + \varphi d$

где $f(x)$ - итоговый прогноз модели;

$\varphi 0$ - базовое значение;

$\varphi 1, \varphi 2, ..., \varphi d$ - вклад каждого признака в прогноз.

Эта идея лежит в основе SHAP: итоговое решение модели раскладывается на вклады отдельных признаков. Для будущей программной системы это важно, потому что объяснение повышает доверие к модели и помогает аналитику быстрее принять решение.

Таким образом, анализ литературы показывает, что современная задача прогнозирования угроз информационной безопасности требует комплексного подхода. Статистические методы дают основу для обнаружения отклонений (Denning D.E., 1987). Классические методы машинного обучения позволяют строить классификаторы на признаках сетевого трафика (Buczak A.L.,2016; Ferrag M.A., 2022; Breiman L., 2001; Chen T., 2016). Рекуррентные, сверточные и Transformer-модели позволяют учитывать временные зависимости (Hochreiter S.,1997; Kim J.,2016; Staudemeyer R.C., 2015; Vaswani A. 2017; Yu W., 2021). Графовые нейронные сети позволяют учитывать структуру взаимодействий между узлами (Kipf T.N.,2017; Lo W.W.,2022; Caville E.,2022, Hamilton W.L., 2020; Scarselli F., 2009; Hamilton W.L., 2017). Методы XAI позволяют объяснять решения модели (Goodfellow I.J.,2015; Madry A.,2018; Warnecke A.,2020; Lundberg S.M.,2017). Стандарты и фреймворки, такие как MITRE ATT&CK и NIST Cybersecurity Framework 2.0, позволяют связать математическую модель с практикой управления киберрисками (Murphy K.P., 2012; MITRE).

Особое значение имеют современные датасеты. CICIDS2017, CSE-CIC-IDS2018, CICIoT2023, DAPT 2020, Bot-IoT, ToN-IoT, UNSW-NB15 и Edge-IIoTset используются для проверки методов обнаружения и прогнозирования атак (Sharafaldin I.,2018; Lashkari A.H.,2017; Neto E.C.P., 2023; Myneni S., 2020; Moustafa N., 2015; Koroniotis N.,2019; Moustafa N., 2020; Ferrag M.A., 2022). При этом CICIoT2023 содержит 33 типа атак в IoT-топологии из 105 устройств, что делает его полезным источником для будущей проверки моделей на IoT-сценариях. DAPT 2020 ориентирован на моделирование APT-угроз и может использоваться для проверки способности модели выявлять многоэтапные атаки. Однако важно учитывать, что результаты на датасетах не всегда переносятся на реальные сети, поскольку распределение признаков, нагрузка, структура трафика и поведение пользователей могут существенно отличаться (Layeghy,2022).

Отсюда следует, что будущая программная реализация должна строиться не как простая модель классификации, а как комплексная система, включающая: модуль сбора данных, модуль предобработки, модуль формирования признаков, временную модель,

графовую модель, модуль оценки риска, модуль объяснения решений и интерфейс для аналитика.

Обоснование необходимости разработки

На основании проведённого анализа можно выделить несколько научно-практических противоречий.

Первое противоречие связано с тем, что большинство классических средств защиты работает реактивно, то есть фиксирует уже известный или уже произошедший факт атаки. Однако современным организациям требуется не только обнаружение, но и раннее предупреждение. Поэтому необходима модель, способная оценивать вероятность будущего инцидента.

Второе противоречие связано с тем, что многие ML-модели рассматривают сетевой поток как отдельный объект, не учитывая длинную историю событий. В реальности атака часто развивается постепенно, поэтому модель должна анализировать временные последовательности.

Третье противоречие связано с тем, что традиционные табличные признаки не отражают всей структуры сети. Узлы инфраструктуры связаны между собой, и изменение поведения одного узла может влиять на риск другого. Поэтому требуется графовое представление сети.

Четвёртое противоречие связано с тем, что высокоточные модели глубокого обучения часто являются «чёрным ящиком». Для практического применения в SOC необходимо объяснять решения модели.

Пятое противоречие связано с тем, что модели машинного обучения могут быть уязвимы к состязательным воздействиям. Поэтому при проектировании будущего программного обеспечения необходимо учитывать не только точность, но и устойчивость.

Шестое противоречие связано с тем, что многие научные работы ограничиваются экспериментами в Jupyter Notebook и не переходят к инженерной архитектуре программного комплекса. Для практического применения необходима архитектура, пригодная для интеграции с источниками логов, сетевыми сенсорами, SIEM и интерфейсом аналитика.

Эти противоречия формируют основание для постановки цели и задач исследования.

Цель исследования

Целью исследования является разработка математической, алгоритмической и архитектурной основы для последующей реализации программного обеспечения, предназначенного для моделирования, раннего выявления и прогнозирования угроз информационной безопасности на основе методов машинного обучения, глубокого обучения, графового анализа и объяснимого искусственного интеллекта.

Задачи исследования

Для достижения поставленной цели необходимо решить следующие задачи:

- Провести анализ современных научных источников, стандартов, датасетов и программных подходов в области обнаружения и прогнозирования угроз информационной безопасности.
- Исследовать ограничения сигнатурных, статистических и классических ML-подходов при обнаружении новых, многоэтапных и динамически изменяющихся атак.
- Сформировать математическое описание сетевого события в виде вектора признаков.

- Построить модель потока событий информационной безопасности как временной последовательности.
- Сформулировать вероятностную постановку задачи прогнозирования атаки на заданном временном горизонте.
- Представить сетевую инфраструктуру в виде динамического графа взаимодействующих узлов.
- Обосновать применение временных моделей, включая TCN, для анализа последовательностей событий.
- Обосновать применение графовых нейронных сетей, включая GCN, GraphSAGE и GAT, для учёта связей между узлами сети.
- Разработать математическую схему объединения временного и графового представления для оценки риска атаки.
- Определить функции потерь для задач классификации, прогнозирования риска и работы с несбалансированными классами.
- Рассмотреть подходы к проверке состязательной устойчивости модели с учётом допустимости сетевых признаков.
- Обосновать необходимость использования методов объяснимого искусственного интеллекта для интерпретации решений модели.
- Сформировать концептуальную архитектуру будущего программного комплекса, включающую сбор данных, предобработку, моделирование, прогнозирование, объяснение и визуализацию.
- Определить набор метрик для будущей оценки качества: Accuracy, Precision, Recall, F1-score, ROC-AUC, PR-AUC, False Positive Rate, lead time, latency, throughput и p95/p99 задержку.
- Подготовить основу для последующей реализации программного комплекса, интегрируемого с SIEM/SOC-инфраструктурой.

Список использованных источников

1. Anderson, J.P. (1980). *Computer Security Threat Monitoring and Surveillance*. Fort Washington: James P. Anderson Co.
2. Bai S., Kolter J.Z., Koltun V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. 2018.
3. Bishop C.M. Pattern Recognition and Machine Learning. Springer, 2006.
4. Breiman L. Random Forests // Machine Learning. 2001. Vol. 45. P. 5–32.
5. Brown T.B. et al. Language Models are Few-Shot Learners // NeurIPS. 2020. Vol. 33. P. 1877–1901.
6. Buczak A.L., Guven E. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection // IEEE Communications Surveys & Tutorials. 2016. Vol. 18, No. 2. P. 1153–1176.
7. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System // KDD. 2016. P. 785–794.

8. Denning D.E. An Intrusion-Detection Model // IEEE Transactions on Software Engineering. 1987. Vol. SE-13, No. 2. P. 222–232.

9. ENISA. Threat Landscape 2025. European Union Agency for Cybersecurity, 2025.

10. Ferrag M.A., Maglaras L., Moschoyiannis S., Janicke H. Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative Study // Information Security Journal. 2022. Vol. 31, No. 2. P. 108–139.

11. Goodfellow I.J., Shlens J., Szegedy C. Explaining and Harnessing Adversarial Examples // ICLR. 2015.

12. Hamilton W.L. Graph Representation Learning. Morgan & Claypool, 2020.

13. Hamilton W.L., Ying R., Leskovec J. Inductive Representation Learning on Large Graphs // NeurIPS. 2017. P. 1024–1034.

14. Hochreiter S., Schmidhuber J. Long Short-Term Memory // Neural Computation. 1997. Vol. 9, No. 8. P. 1735–1780.

15. IBM Security. Cost of a Data Breach Report 2025. IBM, 2025.

16. Kim J., Kim J., Thu H.L.T., Kim H. Long Short Term Memory Recurrent Neural Network Classifier for Intrusion Detection // ICUIMC. 2016. Article 94.

17. Kipf T.N., Welling M. Semi-Supervised Classification with Graph Convolutional Networks // ICLR. 2017.

18. Koroniotis N., Moustafa N., Sitnikova E., Turnbull B. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset // Future Generation Computer Systems. 2019. Vol. 100. P. 779–796.

19. Lo W.W., Layeghy S., Sarhan M., Gallagher M., Portmann M. E-GraphSAGE: A Graph Neural Network Based Intrusion Detection System for IoT // IEEE/IFIP NOMS. 2022. P. 1–9.

20. Madry A., Makelov A., Schmidt L., Tsipras D., Vladu A. Towards Deep Learning Models Resistant to Adversarial Attacks // ICLR. 2018.

21. MITRE. MITRE ATT&CK Enterprise Matrix.

22. Moustafa N. ToN_IoT Datasets: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems // IEEE Access. 2020. Vol. 8. P. 165130–165150.

23. Moustafa N., Slay J. UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems // MilCIS. 2015. P. 1–6.

24. Murphy K.P. Machine Learning: A Probabilistic Perspective. MIT Press, 2012.

25. Myneni S., Chowdhary A., Sabur A., Sengupta S., Agrawal G., Huang D., Kang M. DAPT 2020: Constructing a Benchmark Dataset for Advanced Persistent Threats // MLHat 2020. Springer, 2020. P. 138–163.

26. Neto E.C.P. et al. CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment // Sensors. 2023. Vol. 23, No. 13. Article 5941.

27. Scarselli F., Gori M., Tsoi A.C., Hagenbuchner M., Monfardini G. The Graph Neural Network Model // IEEE Transactions on Neural Networks. 2009. Vol. 20, No. 1. P. 61–80.

28. Shrikumar A., Greenside P., Kundaje A. Learning Important Features Through Propagating Activation Differences // ICML. 2017. P. 3145–3153.

29. Staudemeyer R.C. Applying Long Short-Term Memory Recurrent Neural Networks to Intrusion Detection // South African Computer Journal. 2015. Vol. 56, No. 1. P. 136–154.

30. Tavallaee M., Bagheri E., Lu W., Ghorbani A.A. A Detailed Analysis of the KDD CUP 99 Data Set // CISDA. 2009. P. 1–6.

31. Vaswani A. et al. Attention Is All You Need // NeurIPS. 2017. P. 5998–6008.

32. Veličković P., Cucurull G., Casanova A., Romero A., Liò P., Bengio Y. Graph Attention Networks // ICLR. 2018.

33. Verizon. Data Breach Investigations Report 2026. Verizon Business, 2026.

34. Warnecke A., Arp D., Wressnegger C., Rieck K. Evaluating Explanation Methods for Deep Learning in Security // IEEE EuroS&P. 2020. P. 158–174.

35. Yu W., Ge Z., Sun P., Wang J., Xu W. LogBERT: Log Anomaly Detection via BERT // IJCNN. 2021. P. 1–8.