УДК 004.056+004.62+004.94
doi:10.31799/1684-8853-2025-5-22-34
EDN: AYCIBD

Научные статьи
Articles

Формальный фреймворк для OSINT-нарушителя и защитника

В. В. Грызунов[a], *доктор техн. наук, доцент, orcid.org/0000-0003-4866-217X, viv1313r@mail.ru*
[a]*Санкт-Петербургский университет Государственной противопожарной службы МЧС России им. Героя Российской Федерации генерала армии Е. Н. Зиничева*

Введение: *распространение OSINT в кибербезопасности выявило разрыв между практическими инструментами и отсутствием их теоретического осмысления. Существующие подходы не предлагают целостной модели процесса OSINT-анализа, что препятствует его формальной автоматизации, верификации и анализу угроз.* ***Цель:*** *разработать формальную модель OSINT-нарушителя в виде абстрактной машины, описывающей процесс разведки через графовое представление знаний, логику вывода и ресурсные ограничения.* ***Результаты:*** *предложена модель OSINT-нарушителя как абстрактной машины, оперирующей на итеративно расширяемом графе знаний. Задача OSINT формализована как поиск пути от публичных свойств к конфиденциальным. Сформулировано необходимое и достаточное условие раскрытия конфиденциального свойства, а понятие триангуляции определено через поиск независимых (непересекающихся) путей в графе. Анализ показал, что OSINT-атаки эксплуатируют «скрытый канал вывода», нарушая принципы классических моделей безопасности, например мандатной модели Белла — Лападулы. Введена формальная функция останова, объединяющая практические критерии завершения анализа: достижение цели, исчерпание ресурсов, убывающую отдачу и риск обнаружения. Формализовано понятие «независимые источники».* ***Практическая значимость:*** *теоретическая модель была транслирована в два прикладных фреймворка: в OSINT Kill Chain — пошаговую методологию для атакующих (Red Team) и в Blue Team Playbook — зеркальный фреймворк для защитников, описывающий аудит цифрового следа и минимизацию рисков.* ***Обсуждение:*** *в качестве направления для дальнейших исследований предложено использовать аппарат формальных грамматик для моделирования не только статических связей, но и динамических тактик, техник и процедур злоумышленника.*

Ключевые слова *— OSINT, разведка по открытым источникам, канал логического вывода, модель нарушителя, граф знаний, фреймворк, киберразведка, информационная безопасность, Red Team, Blue Team, атака на основе логического вывода, мандатная модель разграничения доступа.*

Для цитирования: Грызунов В. В. Формальный фреймворк для OSINT-нарушителя и защитника. *Информационно-управляющие системы*, 2025, № 5, с. 22–34. doi:10.31799/1684-8853-2025-5-22-34, EDN: AYCIBD
For citation: Gryzunov V. V. A formal framework for the OSINT attacker and defender. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 5, pp. 22–34 (In Russian). doi:10.31799/1684-8853-2025-5-22-34, EDN: AYCIBD

Введение

Разведка по открытым источникам (Open Source Intelligence, OSINT) стала важным инструментом в киберразведке и корпоративной безопасности. Рынок OSINT ожидает экспоненциальный рост с 14,85 млрд USD в 2024 г. до 49,39 млрд USD в 2029-м (https://www.thebusinessresearchcompany.com/report/open-source-intelligence-global-market-report). Являясь основой этапа Reconnaissance в Cyber Kill Chain, данные OSINT используются на последующих стадиях кибератак, например для создания эксплойтов или целевого фишинга. С другой стороны, применение OSINT в защите (https://mythreats.com/open-source-intelligence) помогло предотвратить более 75 % фишинговых атак и сократило среднее время обнаружения кибератак на 62 %.

Несмотря на это исследования в основном сосредоточены на извлечении данных, в них практически отсутствуют формальные модели самого аналитического процесса OSINT. Это создает пробел между инженерной реализацией и теоретическим описанием. Создание формализованной модели OSINT-нарушителя позволит автоматизировать и верифицировать работу Red Team и Blue Team, что снизит затраты на OSINT более чем в пять раз (https://business.privacybee.com/resource-center/the-double-edge-sword-of-osint-in-cybersecurity/).

Обзор литературы

Практически ориентированная MITRE ATT & CK описывает дискретные техники разведки, но не сам процесс. Продвинутые языки моделирования угроз [1] симулируют атаки внутри периметра, однако предшествующий этап OSINT-разведки и сам аналитик с его ресурсными ограничениями остаются неформализованными.

Систематический обзор 76 научных работ [2] демонстрирует активное применение графов знаний и онтологий для анализа киберугроз (Cyber Threat Intelligence, CTI). Однако основное внимание в проанализированных работах уделяется обработке уже существующих отчетов об

угрозах и структурированных данных. Процесс сбора и анализа первичной информации из открытых источников (OSINT), который предшествует CTI, а также сама деятельность OSINT-нарушителя как субъекта с практическими ограничениями остаются в значительной степени неформализованными.

Исследования по применению искусственного интеллекта в OSINT [3] сосредоточены на решении отдельных задач, таких как классификация данных, но не предлагают целостной модели самого расследования как ресурсозависимой деятельности. Даже узкоспециализированные работы концентрируются на прикладных методологиях, а не на обобщенной формализации. Так, работа [4] предлагает концептуальную схему расследования в даркнете, а исследование [5] демонстрирует методику отслеживания судов. Обе работы описывают практические подходы в конкретных кейсах, но не создают универсальной, ресурсозависимой модели аналитического процесса.

Подходы к автоматическому моделированию угроз [6] исходят из того, что модель атакуемой системы уже известна, но не объясняют, как злоумышленник получает информацию для ее построения с помощью OSINT. Аналогично, CTI-платформы [7] работают с уже готовыми индикаторами компрометации, оставляя за рамками фундаментальный процесс формирования этих знаний из разрозненных открытых данных.

Работы в смежных областях, таких как геоинформационные системы [8] или человеко-компьютерное взаимодействие [9], также не ставят своей целью формализацию OSINT-процесса как вычислительной задачи. Наконец, предыдущие исследования авторов [10, 11] носили описательный характер, каталогизируя угрозы, но оставляя открытым вопрос о строгой формализации самого процесса анализа.

Цель данной статьи — восполнить пробел между инженерными и научными знаниями путем формализации OSINT-нарушителя (аналитика) как абстрактной вычислительной машины с логикой inference, графовой моделью и критериями останова, что делает ее вкладом и в теорию, и в практику OSINT-анализа.

Формальная постановка задачи OSINT

Задача OSINT — из открытых свойств некоторых объектов из множества объектов (O) выделить открытые свойства интересующего объекта o^* и по ним восстановить искомые конфиденциальные свойства объекта o^*.

Определения.

1. Пусть A — универсальное множество всех возможных *атрибутов* (например, «никнейм», «email», «GPS-координата»), тогда свойством объекта o называется конкретная пара «атрибут-значение» $x = <k, v>$, где $k \in A$, а v — значение.

2. Пусть X — универсальное множество всех возможных *свойств* (атрибутов (k)-значений (v)). Для интересующего объекта o^* из множества объектов O (персоны, компании, системы), $o^* \in O$, его набор свойств $X_{o^*} \subseteq X$.

3. $X^{pub} \subseteq X$ — множество всех публичных свойств, т. е. свойств, доступных неограниченному кругу лиц.

4. $X^{conf} \subseteq X$ — множество всех конфиденциальных свойств, т. е. свойств, доступных ограниченному кругу лиц, конфиденциальность этих свойств нужно защитить, $X^{pub} \cap X^{conf} = \varnothing$.

Инструменты OSINT-нарушителя (аналитика)

Предикат P — это логическая функция, которая для любой пары свойств $(x, y) \in X \times X$ возвращает True или False, т. е. $P: X \times X \to \{\text{True, False}\}$. Предикат не характеризует «событие» в динамическом смысле, а устанавливает истинность статического, проверяемого факта о наличии связи между двумя свойствами.

У аналитика есть набор знаний и техник (Google, WHOIS, анализ метаданных и т. д.). Каждую такую технику можно формализовать как предикат.

Множество предикатов аналитика $\mathcal{P}$ — это конечное множество всех предикатов, которыми владеет аналитик: $\mathcal{P} = \{P_1, P_2, ..., P_k\}$, где P_i — это отдельный предикат.

Примеры предикатов из $\mathcal{P}$: 1) P_google(x, y): истинно, если поиск в Google по свойству x выдает в результате свойство y; 2) P_admin(x, y): истинно, если никнейм x указан как администратор сайта y; 3) P_whois(x, y): истинно, если WHOIS-запрос к домену x возвращает email y; 4) P_metadata(x, y): истинно, если в метаданных файла x содержится геотег y.

Набор $\mathcal{P}$ — это формализованный инструментарий аналитика.

Каждый предикат $P \in \mathcal{P}$ однозначно порождает бинарное отношение r, которое является подмножеством $X \times X$. Это отношение состоит из всех пар, для которых данный предикат истинен.

Формально, для каждого $P_i \in \mathcal{P}$ соответствующее ему отношение r_i определяется как $r_i = \{(x, y) \in X \times X \mid P_i(x, y)\}$.

Примеры порожденных отношений:

1) r_google = $\{(x, y) \in X \times X \mid \text{P\_google}(x, y)\}$;

2) r_admin = $\{(x, y) \in X \times X \mid \text{P\_admin}(x, y)\}$;

3) r_whois = $\{(x, y) \in X \times X \mid \text{P\_whois}(x, y)\}$.

Множество рабочих отношений аналитика $\mathcal{R}_A$ — это множество всех отношений, порожденных всеми предикатами из его инструментария $\mathcal{P}$:

$\mathcal{R}_A = \{r_i \mid P_i \in \mathcal{P}\}$, т. е.: $\mathcal{R}_A = \{\{(x, y) \in X \times X \mid P(x, y)\} \mid P \in \mathcal{P}\}$.

Отсюда вытекает порядок работы аналитика: 1) аналитик имеет конечный набор правил/инструментов ($\mathcal{P}$); 2) с помощью $\mathcal{P}$ он строит (порождает) конечный набор конкретных отношений ($\mathcal{R}_A$), нужных ему для работы.

Дано:

O — множество объектов исследования (персоны, компании);

$o^* \in O$ — целевой объект, интересующий аналитика;

$Y^{pub} = X_{o^*} \cap X^{pub}$ — множество всех открытых свойств целевого объекта o^*;

$Y^{known} \subseteq Y^{pub}$ — множество открытых свойств объекта o^*, известных аналитику на начальном этапе;

$x_c^* \in X^{conf}$ — искомое конфиденциальное свойство объекта o^*;

$\mathcal{P}$ — множество предикатов (инструментов), доступных аналитику;

$\mathcal{R}_A$ — множество всех рабочих отношений аналитика;

$R^{pub\_pub} = \{r \in \mathcal{R}_A \mid \forall(x, y) \in r, (x \in X^{pub} \wedge y \in X^{pub})\}$ — множество отношений, связывающих одни публичные свойства с другими (например, «никнейм — viv» -> «сайт — https://a-tree.ru»);

$R^{pub\_conf} = \{r \in \mathcal{R}_A \mid \forall(x, y) \in r, (x \in X^{pub} \wedge y \in X^{conf})\}$ — множество рабочих отношений, связывающих публичные свойства с конфиденциальными: $R^{pub\_pub} \cap R^{pub\_conf} = \varnothing$ (например, «номер телефона — + 79…» -> «ФИО — Иванов Иван»).

Требуется:

Найти такой набор свойств $(y_1, y_2, ..., y_n)$, где $y_1 \in Y^{known}$, и такую последовательность отношений $r_1, r_2, ..., r_n$, что:

$(y_1, y_2) \in r_1$, где $r_1 \in R^{pub\_pub}$;

...

$(y_{\{n-1\}}, y_n) \in r_{\{n-1\}}$, где $r_{\{n-1\}} \in R^{pub\_pub}$;

$(y_n, x_c^*) \in r_n$, где $r_n \in R^{pub\_conf}$.

Фактически, задача сводится к поиску пути в графе (G), где вершины — это свойства, а ребра — отношения.

Критерии завершения OSINT-анализа (точка останова)

OSINT-анализ останавливается при выполнении одного из условий.

1. Цель достигнута (Goal Reached): найдено искомое свойство x_c^* с достаточной степенью уверенности (assurance, $A(x_c^*)$). $A(x_c^*)$ может быть оценена на основе количества и независимости источников, подтверждающих вывод (триангуляция). Анализ останавливается, если $A(x_c^*) \geq A_{threshold}$, где $A_{threshold}$ — заранее заданный порог уверенности, который может измеряться с помощью лингвистической переменной (низкий, средний, высокий) или быть числом [0; 1].

2. Исчерпание зацепок (Lead Exhaustion): анализ останавливается, когда все доступные на данный момент публичные свойства $y \in Y^{pub}$ были полностью исследованы, но не привели к появлению новых релевантных свойств или путей к искомому x_c^*. Аналитик достиг «тупика»: все исходящие ребра из известных вершин графа G ведут к уже исследованным вершинам или к нерелевантной информации, и новых вершин для добавления в граф нет. Дальнейший пассивный поиск в рамках выбранных источников становится невозможным.

3. Исчерпание ресурсов (Resource Depletion): превышен лимит выделенного времени ($T_{spent} \geq T_{limit}$) или бюджета ($C_{spent} \geq C_{limit}$). Это наиболее частый практический ограничитель. C_{limit} может производиться не только в денежном выражении, но и через определение предельно допустимых затрат времени и рисков, которые организация готова понести ради достижения цели, что отражается в работах по анализу конфликтующих ценностей [12]: точности, безопасности, эффективности, прозрачности, приватности, доверия и др.

4. Закон убывающей отдачи (Diminishing Returns): стоимость получения следующего фрагмента информации превышает его потенциальную ценность. Пусть $\text{Value}(y_{new})$ — ценность новой информации y_{new}, а $\text{Cost}(y_{new})$ — затраты на ее получение. Анализ останавливается, если $(\text{Value}(y_{new})/\text{Cost}(y_{new})) \leq K_{threshold}$, где $K_{threshold}$ — заранее заданный коэффициент. Например, если для получения одного бита информации нужно потратить неделю работы, это неэффективно.

5. Повышение риска обнаружения (Detection Risk, $D_{detection}$): анализ переходит от пассивного (сбор публичных данных) к активному (взаимодействие в целях, например, отправки запроса на добавление в друзья). Это повышает $D_{detection}$. Анализ останавливается или аналитик меняет тактику, если $D_{detection} \geq D_{limit}$.

Сопоставление каждому допустимому действию конкретной величины риска выполняется, например, с помощью метода iSOFT [13], который предполагает на основе морфологического ящика последовательный поиск субстанциальных закономерностей (отношений r), связывающих величины (y, x_c^*) между собой. Либо аналитик обозначает риск открыть себя как риск нарушить свою «конфиденциальность», тогда в информационной системе аналитика применимы соответствующие подходы, в том числе с использованием лингвистической переменной для измерения риска: высокий, средний, красный и т. д. В рамках представленной в статье модели увеличение риска Risk_increase для каждой техники m может быть рассчитано как вероятность обнаружения этой техники, оцененная по методологии MAGIC [14], что позволяет перейти от абстрактного параметра D_{risk} к количественной

оценке вероятности инцидента на основе анализа «киберпозиции» организации. Исследование [15] предлагает упрощенный алгоритм: сочетание рейтингов риска, защитных мер и угроз. Подход позволяет самостоятельно вычислять вероятность события информационной безопасности и корректировать ее на основе показателей инфраструктуры. Авторы [16] используют модель с нечеткими правилами и энтропией безопасности. Статья [17] рассматривает расширенную модель FAIR, где применен гибридный подход: качественная оценка плюс конверсия в вероятности. Формализована схема агрегации вероятностей на основе метамодели.

При проведении OSINT возможны следующие векторы атак.

1. Поиск прямого или многошагового пути от известных публичных свойств к искомым конфиденциальным через общедоступные отношения (R^{pub_pub}, R^{pub_conf}).

2. Активное провокативное взаимодействие для раскрытия новых публичных или конфиденциальных свойств.

Утверждение о необходимом и достаточном условии OSINT

Конфиденциальное свойство x_c^* может быть однозначно восстановлено из набора открытых свойств $Z^{pub} \subseteq X^{pub}$ тогда и только тогда, когда существует отношение $r \in R^{pub\_conf}$ такое, что образ множества Z^{pub} при этом отношении является одноэлементным множеством, содержащим только x_c^*:

$$\exists r \in R^{pub\_conf} : r[Z^{pub}] = \{x_c^*\}.$$

Доказательство: Доказательство следует напрямую из определений.

1. (⇒) Необходимость. Пусть x_c^* однозначно восстановимо из Z^{pub}. Это означает, что: а) существует механизм (назовем его отношением r), который связывает Z^{pub} с x_c^*; б) результат восстановления является единственным (однозначным).

Из (а) следует, что существует пара $<z, x_c^*> \in r$ для некоторого $z \in Z^{pub}$, а значит, $x_c^* \in r[Z^{pub}]$. Из (б) следует, что для любого другого конфиденциального свойства $x' \neq x_c^*$ не существует связи с Z^{pub} через r. Это означает, что никакой другой элемент, кроме x_c^*, не может находиться в образе $r[Z^{pub}]$.

Следовательно, $r[Z^{pub}] = \{x_c^*\}$.

2. (⇐) Достаточность. Пусть существует отношение $r \in R^{pub\_conf}$ такое, что $r[Z^{pub}] = \{x_c^*\}$. По определению образа множества, это означает, что для элементов из Z^{pub} существуют связи в отношении r, и все эти связи ведут исключительно к одному элементу — x_c^*. Отсутствие в множестве $r[Z^{pub}]$ других элементов означает, что никакой другой результат, кроме x_c^*, не может быть получен. Следовательно, свойство x_c^* восстанавливается однозначно.

Что и требовалось доказать.

Замечания для практического применения доказанного утверждения.

1. Данное утверждение формулирует необходимость и достаточность в теоретическом смысле. Практическая осуществимость восстановления конфиденциального свойства x_c^* дополнительно требует наличия у OSINT-аналитика технологии, которая позволяет использовать отображение r согласно критериям завершения OSINT-анализа.

2. На практике отношение $r[Z^{pub}] \in R^{pub\_conf}$ обычно является многозначным, т. е. r: $[Z^{pub}] \rightarrow \rightarrow \{x_1^{conf}, x_2^{conf}, \ldots, x_i^{conf}\}$, следовательно, восстановление искомого конфиденциального свойства будет неоднозначным. То есть x_c^* восстанавливается с некоторой вероятностью/степенью уверенности.

Достоверность вывода о наличии конфиденциального свойства x_c^* существенно возрастает, если несколько различных и независимых публичных свойств $\{x_1^{pub}, x_2^{pub}, \ldots, x_j^{pub}\}$ указывают на него.

Поиск отношения r, которое является отношением «многие-к-одному» и в котором публичные свойства получены из независимых источников, является основной целью аналитической работы в OSINT и называется сходимостью доказательств (или триангуляцией):

$$r : x_c^* \in r[x_1^{pub}] \cap r[x_2^{pub}] \cap \ldots \cap r[x_j^{pub}].$$

Независимые источники OSINT

Два публичных свойства y_1 и y_2, указывающие на одно и то же конфиденциальное свойство x_c, считаются полученными из независимых источников, если процесс получения y_1 и процесс получения y_2 не имеют общих промежуточных свойств (узлов) или отношений (ребер) в графе расследования (G), за исключением самого целевого объекта.

Формально пусть:

$\text{Path}(y_1 \rightarrow x_c)$ — это цепочка (путь в графе) свойств (узлов) и отношений (дуг), которая ведет из y_1 в x_c;

$\text{Path}(y_2 \rightarrow x_c)$ — это цепочка, которая привела нас из y_2 в x_c.

Источники y_1 и y_2 считаются независимыми по отношению к x_c, если пересечение множеств элементов (свойств и отношений) двух путей является пустым:

$$\text{Nodes}(\text{Path}(y_1 \rightarrow x_c)) \cap \text{Nodes}(\text{Path}(y_2 \rightarrow x_c)) = \varnothing;$$

$$\text{Edges}(\text{Path}(y_1 \rightarrow x_c)) \cap \text{Edges}(\text{Path}(y_2 \rightarrow x_c)) = \varnothing,$$

где Nodes и Edges — функции, возвращающие множество узлов и ребер пути соответственно (кроме конечного узла x_c).

Таким образом, можно сказать, что два доказательства независимы, если они не опираются на одни и те же промежуточные факты.

Пример работы с независимыми источниками

Пусть необходимо установить ФИО (x_c) владельца Telegram-канала.

Зависимые источники: путь 1 через email goodboy@a-tree.ru из описания канала и путь 2 через тот же email, найденный в блоге по ссылке из поста, — зависимы. Они сходятся в промежуточном узле goodboy@a-tree.ru (Nodes(Path1)∩ ∩ Nodes(Path2) ≠ ∅). Достоверность вывода невысока.

Независимые источники: путь 1 через email goodboy@a-tree.ru и путь 2 через GPS-координаты из метаданных фото, взятого в Telegram-канале — независимы (Nodes(Path1)∩ ∩ Nodes(Path2) = ∅). Если пути ведут к одному ФИО, достоверность вывода значительно возрастает, что и является сутью триангуляции (рис. 1).

На основе изложенного формализуется модель OSINT-нарушителя.

Модель OSINT-нарушителя

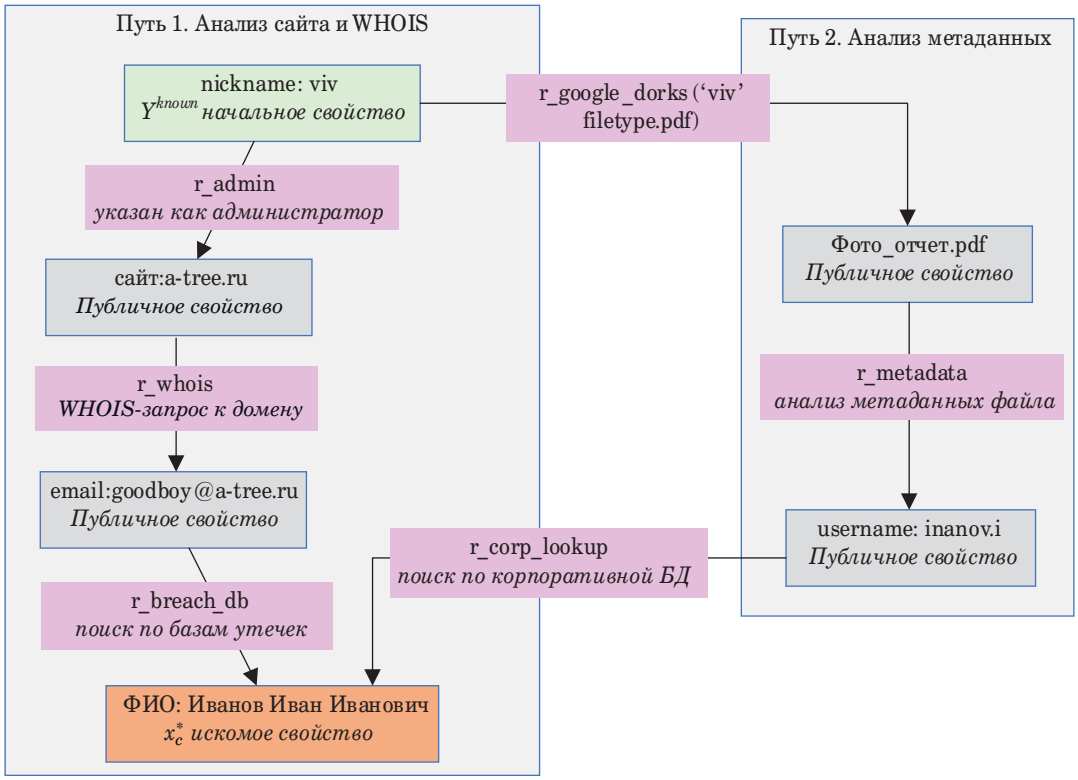
OSINT-нарушитель (далее — Аналитик) является абстрактной машиной, которая восстанавливает конфиденциальные данные на основе анализа общедоступных данных. Модель Аналитика определяется следующей семеркой:

$$A = (O, X, \mathcal{P}, S_i, M, F\_stop).$$

Здесь:

1. $S_i = <Y^{known}, C_{spent}, T_{spent}, D_{risk}, G_i>$ — состояние Аналитика на шаге i. Это кортеж, описывающий текущее состояние расследования, где $G_0 = Y^{known}$ — множество открытых свойств объекта o^*, известных Аналитику на начальном этапе, начальный граф G_0; C_{spent} — потраченный бюджет; T_{spent} — потраченное время; D_{risk} — текущий уровень риска обнаружения (0 для пассивных техник и >0 для активных); $G_i = (G_X(i), G_R(i))$ — граф знаний Аналитика, где $G_X(i)$ — вершины графа, известные Аналитику свойства на шаге i; $G_R(i) \subseteq G_X(i) \times G_X(i)$ — ребра графа, известные Аналитику отношения из R^{pub_pub} на шаге i.

2. M — множество функций (техник), доступных Аналитику для расширения своих знаний. Каждая техника $m \in M$ является функцией, ко-



■ ***Рис. 1.*** Пример графа знаний OSINT-аналитика
■ ***Fig. 1.*** Example of an OSINT analyst's knowledge graph

торая по известному свойству находит новые связанные с ним свойства и отношения, и имеет свою «стоимость».

m: $y \to (\Delta G_X, \Delta G_R$, Cost, Time, Risk_increase), где $y \in Y^{known}$; ΔG_X — множество новых найденных вершин (свойств); ΔG_R — множество новых найденных дуг (отношений); Cost — стоимость использования техники (например, плата за API-запрос); Time — время, затрачиваемое на использование техники; Risk_increase — увеличение риска обнаружения (0 для пассивных техник, >0 для активных).

Например, m_GoogleDorks(“example.com”, 0,17, 3, 0) возвращает множество пар связанных поддоменов и файлов, которые доступны по прямой ссылке на поддомене, стоимость — 0,17 единиц, длительность — 3 с, риска обнаружения нет.

Эта функция моделирует способность Аналитика находить реально существующие пары свойств, для которых истинен некоторый предикат из $\mathcal{P}$.

Конкретными примерами таких техник m являются многочисленные утилиты и сервисы, подробно каталогизированные в работе [18], такие как использование Google Dorks для поиска файлов (m_GoogleDorking), применение сервиса haveibeenpwned для проверки утечки email (m_HIBP_Check) или использование Shodan для поиска уязвимых устройств (m_Shodan_Scan). Предлагаемая модель позволяет абстрагироваться от конкретной реализации техник m, рассматривая их как эквивалентные «черные ящики», преобразующие одно известное свойство в множество новых.

Функция $m \in M$ — это, по сути, практическая реализация предиката $P \in \mathcal{P}$; $m(y)$ — это процедура, которая находит все z, для которых $P(y, z)$ истинно.

3. F_stop — функция останова. Булева функция, которая определяет, должен ли Аналитик прекратить расследование. Функция объединяет все критерии останова:

$$\text{F\_stop}(S_i, G_i, L) \to \{True, False\},$$

где $L = <C_{limit}, T_{limit}, K_{threshold}, A_{threshold}, D_{limi}>$ — кортеж лимитов и порогов, заданных для расследования.

Процесс останавливается, если F_stop возвращает True. F_stop возвращает True, если выполняется одно из следующих условий:

$$\text{F\_stop} = (\text{f\_goal} \vee \text{f\_resource} \vee \text{f\_risk} \vee \vee \text{f\_deadend} \vee \text{f\_diminishing}).$$

Здесь:

f_goal: $\exists\ x_c \in X^{conf}$: Confidence($x_c$, G_i) $\geq A_{threshold}$) — найдено искомое свойство x_c с заданной уверенностью $A_{threshold}$, Confidence(x_c, G_i) — функция, оценивающая уверенность в свойстве x_c на основе графа G_i (например, через количество независимых путей). Формирование функции Confidence является отдельной задачей и выходит за рамки данного исследования;

f_resource: $(C_{spent} \geq C_{limit}) \vee (T_{spent} \geq T_{limit})$ — ресурсы исчерпаны;

f_risk: $D_{detection} \geq D_{limit}$ — риск превышен;

f_deadend: $\forall\ y \in Y^{known}$, если (ΔG_X, ΔG_R, ...) = $m(y)$, то $\Delta G_X = \varnothing$ и $\Delta G_R = \varnothing$ — тупик расследования для всех известных свойств и всех доступных техник, т. е. ни одна из них не дает новых результатов;

f_diminishing: $\forall\ m \in M$: (Value(m) / Cost(t)) $< K_{threshold}$ — закон убывающей отдачи. Формирование функций Value и Cost является отдельной научной задачей и выходит за рамки данного исследования.

Следует подчеркнуть, что граф знаний G не является статической структурой. Он строится динамически: начиная с исходного набора известных свойств Y^{known}, Аналитик на каждом шаге итеративно добавляет в граф новые вершины (свойства) и ребра (отношения), обнаруженные с помощью техник из множества M.

Процесс работы Аналитика

Процесс является итеративным. На каждом шаге i:

1. Аналитик выбирает оптимальную или подходящую технику $m_{\text{opt}} \in M$ и свойство $y \in Y^{known}$ для применения (например, по критерию max(Value(m)/Cost(m))).

Применяет технику: (ΔG_X, ΔG_R, Cost, Time, risk_inc) = $m_{\text{opt}}(y)$ и обновляет свое состояние $S(i + 1)$:$G_X(i + 1) = G_X(i) \cup \Delta G_X$;$G_R(i + 1) = G_R(i) \cup \cup \Delta G_R$; $C_{spent}(i + 1) = C_{spent}(i)$ + Cost; $T_{spent}(i + 1) = = T_{spent}(i)$ + Time; $D_{risk}(i + 1) = D_{risk}(i)$ + risk_inc.

2. Вычисляет F_stop($S(i + 1)$, $G(i + 1)$, L). Если True, процесс завершается. Иначе, переход к шагу $i + 2$.

Пример применения разработанной модели

Процессный фреймворк OSINT-атаки (OSINT Kill Chain). OSINT Kill Chain — это последовательность этапов, которую проходит Аналитик для достижения цели. Данный термин используется по аналогии с классической моделью Cyber Kill Chain для описания последовательности этапов, специфичных именно для процесса разведки по открытым источникам, который в свою очередь является первым этапом комплексной атаки.

Этап 1. Целеполагание (Direction & Planning) — определение цели расследования и ограничений.

Определяется целевой объект $o^* \in O$ (например, компания «Ромашка» или персона «Иван Иванов»). Формулируется искомое конфиденциальное свойство $x_c^* \in X^{conf}$ (например, «реальный

IP-адрес веб-сервера» или «домашний адрес»). Фиксируются ограничения расследования $L = <C_{limit}, T_{limit}, K_{threshold}, A_{threshold}, D_{limit}>$, предоставленные заказчиком или установленные самим Аналитиком (бюджет, время, пороговая уверенность и т. д.). Проводится инвентаризация доступного инструментария M (набор техник), который порождает множество рабочих отношений $\mathcal{R}_A$.

Результат: четко сформулированная задача с измеримыми критериями успеха и ограничениями.

Этап 2. Сбор исходных данных (Initial Collection) — сбор всей изначально доступной информации о цели.

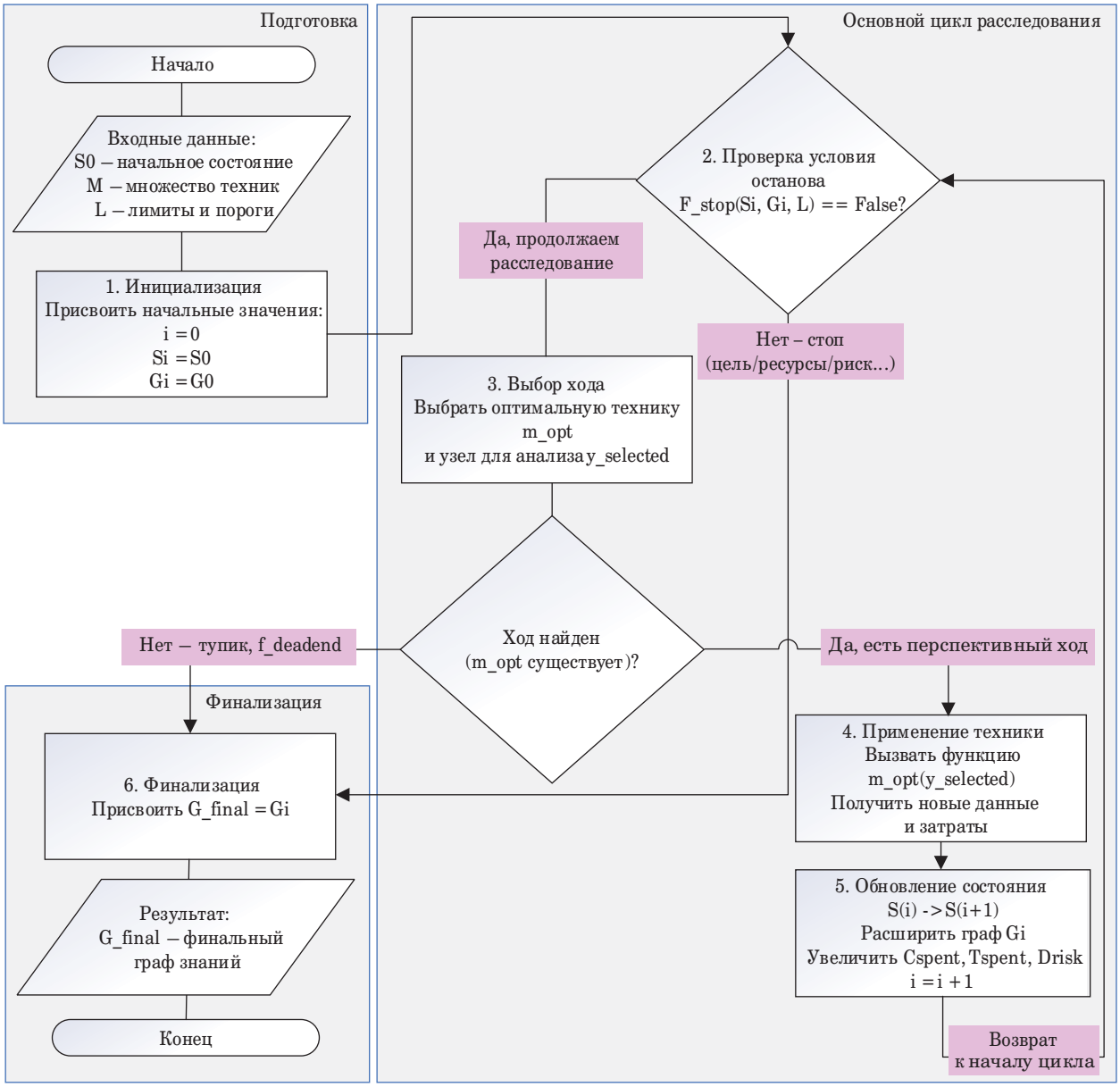
Формируется начальное множество известных публичных свойств $Y^{known} = G_0 = (G_X(0), G_R(0)) \subseteq X^{pub}$ (например, доменное имя сайта, известный никнейм, ФИО).

Результат: исходная точка для расследования — граф G_0.

Этап 3. Итеративное расширение (Iterative Expansion & Pivoting) — основной цикл расследования (рис. 2).

1. Аналитик использует известные данные для поиска новых, постепенно расширяя карту знаний (пивотинг, pivoting):

— выбор узла и техники: выбор свойства $y \in G_X(i)$ и оптимальной/подходящей техники



■ ***Рис. 2.*** Алгоритм основного цикла расследования (этап 3)

■ ***Fig. 2.*** Flow chart of the main investigation cycle algorithm (stage 3)

$m_{opt} \in M$ (например, по критерию максимальной ожидаемой ценности);

— применение техники: (ΔG_X, ΔG_R, Cost, Time, risk_inc) = $m_{opt}(y)$.

Пример конкретной техники m: m_GoogleDorking: Аналитик имеет свойство y = <domain, “romashka.ru”>. Он применяет технику m_GoogleDorking с запросом “site:romashka.ru filetype:pdf”.

Результат вызова:

$$\text{m\_GoogleDorking}(y) =$$

$$= (\Delta G_X, \Delta G_R, \text{Cost, Time, risk\_inc}),$$

где ΔG_X = {<url, “romashka.ru/docs/report_2023.pdf”>, <url, “romashka.ru/files/pricelist.pdf”>} — множество новых найденных свойств-ссылок; ΔG_R = {(<domain, “romashka.ru”>, <url, “romashka.ru/docs/report_2023.pdf”>), ...} — множество новых ребер в графе; Cost = 0 — бесплатный API; Time = 60 с — время на формулировку запроса и анализ выдачи; risk_inc = 0 — пассивный сбор, не вызывает подозрений.

2. Обновление состояния аналитика $S(i + 1)$: обновляется граф знаний $G(i + 1) = (G_X(i) \cup \Delta G_X, G_R(i) \cup \Delta G_R)$ и C_{spent}, T_{spent}, D_{risk}.

3. Проверка условия останова: вычисляется F_stop($S(i + 1)$, $G(i + 1)$, L). Если True — переход к этапу 4. Если False — возврат к шагу 1 данного этапа.

Результат: расширенный граф знаний G_{final}, содержащий достаточно информации для вывода, либо остановка процесса по одному из критериев F_stop.

Этап 4. Анализ и синтез (Analysis & Synthesis) — изучение построенного графа знаний для выявления скрытых связей и формирования выводов.

Аналитик ищет в графе G_{final} пути, ведущие от вершин из Y^{known} к вершинам, которые могут являться искомым x_c^*. Особое внимание уделяется триангуляции — поиску нескольких независимых путей, сходящихся в одной и той же вершине x_c^*. На основе количества и независимости путей вычисляется достоверность результата функцией Confidence(x_c^*, G_{final}).

Результат: гипотеза о значении x_c^* с оценкой ее достоверности.

Этап 5. Финализация (Finalization & Reporting) — оформление результатов расследования в виде отчета.

Отчет содержит искомое свойство x_c^*, оценку достоверности Confidence(x_c^*, G_{final}), а также сам граф расследования G_{final} (или его релевантную часть) в качестве доказательной базы. Граф наглядно демонстрирует, как именно был получен вывод, делая процесс прозрачным и верифицируемым.

Результат: документированный, аргументированный и воспроизводимый результат OSINT-анализа.

Фреймворк защиты от OSINT-угроз (Blue Team Playbook)

Данный фреймворк является зеркальным отражением OSINT Kill Chain и направлен на минимизацию рисков, связанных с действиями OSINT-нарушителя, путем управления своим публичным цифровым следом.

Этап 1. Аудит цифрового следа (Digital Footprint Auditing) — проактивный поиск и каталогизация публично доступной информации о своей организации.

Команда защиты (Blue Team) выступает в роли «этичного» OSINT-нарушителя, где объектом o^* является собственная организация. Цель — построить максимально полный граф знаний G_{self} о собственном публичном следе X^{pub}. Для этого команда выполняет этапы 1–4 из OSINT Kill Chain в отношении себя.

Результат: карта публичного цифрового следа организации, представленная в виде графа G_{self}.

Этап 2. Анализ уязвимостей (Inference Channel Analysis) — анализ собранной карты на предмет наличия опасных связей, позволяющих сделать нежелательные выводы.

Blue Team анализирует граф G_{self} в целях выявления ребер (отношений), принадлежащих множеству R^{pub_conf}, т. е. связей, ведущих от публичных свойств X^{pub} к внутренним, конфиденциальным X^{conf}. Например, наличие в графе G_{self} отношения $(y, x_c) \in$ r_VisualAnalysis $\in R^{pub\_conf}$, где y = <url_photo, ‘(публичная ссылка на фото сотрудника)’>, а x_c = <employee_id, ‘(конфиденциальный ID с его пропуска, видимый на фото)’>.

Результат: приоритизированный перечень выявленных «скрытых каналов вывода» (inference covert channels). Формально оценить опасность выявленного канала можно, применив подход, изложенный в работе [19].

Этап 3. Минимизация и разрыв связей (Minimization & Remediation) — устранение или ослабление найденных опасных связей для снижения рисков.

— Цель — сделать для атакующего поиск пути ко всем конфиденциальным свойствам x_c невозможным или слишком дорогим (увеличить Cost и Time, снизить Confidence). Разрыв отношения — прямое удаление одного из элементов связи. Например, удаление фотографии с пропуском или очистка метаданных из документа, что равносильно удалению ребра (y, x_c).

Пример конкретной техники m у защиты: m_MetadataScrubbing. Очистка метаданных из

файлов перед публикацией. Применение: в ходе этапа 2 был найден PDF-файл, в метаданных которого содержалось имя пользователя (x_c = <username, “ivanov.i”>). Blue Team применяет технику m_MetadataScrubbing к этому файлу.

Результат: отношение r_metadata, связывающее <url, “.../report_2023.pdf”> с <username, “ivanov.i”>, перестает существовать.

Для атакующего вызов аналогичной техники m_MetadataAnalysis вернет следующее: $\Delta G_X = \varnothing$ (новых свойств не найдено). Cost, Time будут потрачены впустую. Атакующий упрется в тупик (f_deadend) по этой ветке расследования.

— Минимизация X^{pub} — сокращение общего объема публикуемых данных (принцип need-to-know). Чем меньше вершин в X^{pub}, тем меньше у атакующего стартовых точек.

— Зашумление (Noise Injection) — создание большого количества ложных или неоднозначных связей. Это приводит к тому, что образ $r[Z^{pub}]$ становится многоэлементным множеством, и атакующий не может однозначно идентифицировать x_c^*.

Результат: снижение поверхности атаки (уменьшение размера и (или) связности графа G) для увеличения стоимости и сложности проведения OSINT-анализа для злоумышленника.

Этап 4. Периодический мониторинг и обучение (Continuous Monitoring & Training) — поддержание защищенности на должном уровне в динамичной среде.

— Мониторинг: регулярное (в идеале — автоматизированное) повторение этапа 1 для выявления новых, непреднамеренно опубликованных данных.

— Обучение: проведение тренингов с сотрудниками, которые являются объектами o, по безопасному применению информационных технологий. Цель — научить их не создавать новые опасные отношения $r \in R^{pub\_conf}$.

Результат: формирование в организации культуры управления информацией и обеспечение долгосрочной устойчивости к OSINT-угрозам.

Из предложенной модели OSINT-нарушителя следует возможность нарушения мандатной модели разграничения доступа.

Нарушение модели Белла — Лападулы (мандатной модели) OSINT-нарушителем

Следует подчеркнуть, что любая модель, включая мандатную модель Белла — Лападулы (Bell — LaPadula, BLP), имеет свои границы применимости и неизбежные ограничения. Основное ограничение BLP — фокус на контроле прямых потоков информации внутри замкнутой системы: модель эффективно предотвращает операции «чтение вверх» и «запись вниз», но не учитывает семантические связи и логические выводы, которые могут быть сделаны из данных, легально опубликованных за пределами ее периметра. OSINT-нарушитель, действуя как субъект, не описанный и не контролируемый моделью, эксплуатирует это ограничение, оставаясь для BLP невидимым. Он не нарушает правила доступа напрямую, но использует публичные данные, в том числе производные от конфиденциальной информации, опубликованные уполномоченными субъектами в рамках их прав, чтобы через цепочку логических выводов восстановить конфиденциальные свойства. Таким образом, хотя формально ни одно правило BLP не нарушается, ее основополагающий принцип конфиденциальности оказывается скомпрометирован за счет эксплуатации «скрытого канала вывода», существующего в открытом информационном пространстве вне зоны контроля модели.

Подробный пример нарушения мандатной модели

Сценарий: вычисление местоположения секретного совещания крупной технологической компании InnovateCorp, разрабатывающей секретный проект «Химера». Информация о проекте, включая место и время ключевых совещаний, хранится на внутреннем сервере и защищена BLP.

Формализация в терминах BLP и статьи.

1. Конфиденциальное свойство (объект с высоким уровнем секретности):

— свойство: x_c^* = <meeting_details, «Проект ‘Химера’: совещание в 15:00, 15.03.2025, в переговорной ‘Зенит’ на 25-м этаже главного офиса»>;

— объект в системе: файл chimera_meeting.ics на внутреннем календаре;

— уровень секретности объекта: Level(chimera_meeting.ics) = Secret.

2. OSINT-нарушитель (субъект с низким уровнем доступа):

— субъект: внешний Аналитик (OSINT-нарушитель);

— уровень допуска субъекта: Level(Analyst) = = Unclassified.

3. Правило BLP:

— согласно правилу No_Read_Up, Аналитику запрещена операция Analyst.read(chimera_meeting.ics), так как Level(Analyst) < Level(chimera_meeting.ics).

Итог: система BLP работает корректно. Прямой доступ к файлу невозможен.

Процесс OSINT-атаки через «скрытый канал вывода».

Аналитик не пытается взломать систему. Вместо этого он работает с общедоступными данными (объектами с уровнем Unclassified), чтобы логически вывести x_c^*.

Шаг 1. Сбор публичных свойств (Y^{known}).

Аналитик находит в открытых источниках три, на первый взгляд, не связанных между собой свойства:

1. Публичное свойство y_1 (социальные сети):

— ведущий разработчик InnovateCorp Анна публикует в своей профессиональной сети пост: *«Готовимся к большому дню в пятницу, 15 марта! Предстоит важная демонстрация для руководства. Держите за нас кулачки! #InnovateCorp #BigDay»*;

— формально: y_1 = <post_content, «демонстрация 15 марта»>. Level(y1) = Unclassified.

2. Публичное свойство y_2 (публичный API бронирования):

— у InnovateCorp есть публичная система бронирования переговорных комнат для встреч с внешними клиентами. Аналитик замечает, что 15 марта с 14:00 до 18:00 все переговорные комнаты на 25-м этаже (этаж руководства) помечены как «закрыты на техническое обслуживание», что нетипично для рабочего дня;

— формально: y_2 = <room_status, «25-й этаж, 15 марта, 14:00–18:00 — недоступен»>. Level(y_2) = = Unclassified.

3. Публичное свойство y_3 (публичный репозиторий кода):

— младший разработчик случайно отправил в публичный репозиторий на GitHub фрагмент кода с комментарием: // Срочный фикс для демо ‘Химера’, убрать до 15.03. Комментарий был быстро удален, но остался в истории коммитов;

— формально: y_3 = <code_comment, «демо ‘Химера’ до 15.03»>. Level(y_3) = Unclassified.

Шаг 2. Логический вывод (использование отношений R^{pub_conf}).

Теперь Аналитик использует свой инструментарий (предикаты $\mathcal{P}$), чтобы построить отношения (ребра в графе знаний) между этими публичными свойствами.

1. Отношение r_1 (временная корреляция):

— аналитик связывает пост Анны (y_1) и комментарий в коде (y_3);

— P_temporal_correlation(y_1, y_3) истинно, так как оба свойства указывают на одну и ту же дату (15 марта) и одно и то же событие («демонстрация»);

— вывод 1: «большая демонстрация» 15 марта — это демонстрация проекта «Химера».

2. Отношение r_2 (пространственно-временная корреляция):

— аналитик связывает свой вывод 1 с информацией о бронировании переговорных (y_2);

— P_spatial_correlation(«демо ‘Химера’ 15 марта», y_2) истинно. Логика: важное внутреннее мероприятие для руководства (вывод 1) является наиболее вероятной причиной аномального закрытия целого этажа, где находится руководство (y_2), в то же самое время;

— вывод 2: демонстрация проекта «Химера» пройдет 15 марта на 25-м этаже главного офиса.

3. Отношение r_3 (уточнение времени):

— анализируя корпоративную культуру InnovateCorp, например по прошлым публичным анонсам, Аналитик знает, что крупные демонстрации обычно назначают на середину дня, около 15:00, чтобы успеть собрать всех руководителей;

— P_heuristic_inference(y_2, «15:00») истинно. Блок времени с 14:00 до 18:00 с наибольшей вероятностью содержит встречу, начинающуюся в 15:00;

— вывод 3: наиболее вероятное время начала — 15:00.

Шаг 3. Синтез и получение конфиденциального свойства.

Аналитик объединяет выводы. Он *не нашел* файл chimera_meeting.ics, он вычислил его содержимое x_c^*:

(«демо ‘Химера’») + («15 марта») + («на 25-м этаже») + («~15:00») $\Rightarrow$ $x_c^{*\prime} \approx$ <meeting_details, «Проект ‘Химера’: совещание в 15:00, 15.03.2025, на 25-м этаже»>,

$x_c^{*\prime}$ практически идентичен конфиденциальному свойству x_c^*.

Таким образом:

1) Аналитик ни разу не нарушил правило No_Read_Up BLP. Все его действия: чтение поста в соцсети, просмотр публичного API, анализ истории коммитов на GitHub — были операциями read над объектами с уровнем Unclassified, что полностью разрешено для субъекта с уровнем Unclassified;

2) несмотря на то, что система BLP идеально защитила сам *объект* (файл chimera_meeting.ics), она не смогла защитить *информацию*, которую этот объект содержал;

3) уязвимость возникла не внутри системы, а в семантических связях между легитимно опубликованными данными. Совокупность публичных свойств {y_1, y_2, y_3} создала «скрытый канал вывода», по которому информация уровня Secret стала доступна субъекту уровня Unclassified через логические выводы с какой-то степенью уверенности.

Этот пример наглядно демонстрирует ключевую идею статьи: защита от современных OSINT-угроз требует не только контроля доступа к изолированным объектам данных, что делает BLP, но и управления отношениями между этими данными, т. е. аудита и минимизации цифрового следа организации, что и предлагается во фреймворке для Blue Team.

Даже усиленные реализации модели BLP, такие как в операционных системах Astra Linux [20] или Windows [21], неэффективны против OSINT-атак. Эти модели контролируют внутрисистемные информационные потоки, тогда как OSINT-нарушитель оперирует уже легитимно опубликованными данными извне периметра.

Таким образом, отношение r: $X^{pub} \to X^{conf}$ является уязвимостью типа «скрытый канал вывода». Защита от OSINT-атак — это управление этими скрытыми каналами: их разрыв, зашумление или мониторинг.

Обсуждение

В настоящей статье связь открытого и конфиденциального свойств рассматривается как бинарное отношение, которое в дальнейшем моделируется графом. Вообще говоря, это отношение может не быть бинарным. Тогда оно должно моделироваться гиперграфом. Требует отдельного изучения вопрос, что лучше для практического использования: усложнить модель OSINT-нарушителя или представить гиперграф в виде двудольного.

Помимо теории графов, которая является естественным аппаратом для моделирования сетевой структуры OSINT-данных, для описания самого процесса OSINT-атаки может быть применен аппарат формальных грамматик. В создаваемой формальной грамматике нетерминалами могли бы выступать атрибуты свойств (<Digital_ID>, <Physical_Location>), терминалами — конкретные значения атрибутов (‘user@a-tree.ru’, ‘55.75, 37.61’), а правилами вывода (<Digital_ID> -> <Real_Name>) — техники из множества M. Тогда «язык», порождаемый такой грамматикой, представляет собой множество всех возможных успешных сценариев атаки.

Такой подход позволяет формализовать не только статический цифровой след объекта, но и динамические тактики, техники и процедуры (Tactics, Techniques, and Procedures, TTPs), используемые OSINT-нарушителем. Например, можно построить грамматику, описывающую все валидные цепочки действий для подготовки к фишинговой атаке. Тогда задача аналитика сводится не просто к поиску пути в графе данных, а к поиску пути, который соответствует «синтаксису» успешной атаки, заданному грамматикой. При этом, учитывая сетевую природу OSINT-данных, теория графов остается основным инструментом моделирования, в то время как формальные грамматики могут служить мощным дополнением для верификации и классификации путей атак, что является перспективным направлением для дальнейших исследований.

Практическая реализация поиска путей в графе G_{final} может быть осуществлена с помощью стандартных алгоритмов. Например, волновой алгоритм (поиск в ширину) позволит найти кратчайшую цепочку вывода от публичных данных к конфиденциальным, в то время как муравьиный алгоритм позволяет эвристически моделировать поиск, учитывающий не только длину пути, но и дополнительные факторы, такие как ресурсные затраты и потенциальную полезность промежуточных данных.

Заключение

В данной работе предложена новая, практически ориентированная модель OSINT-нарушителя. В отличие от существующих подходов, разработанный фреймворк формализует работу OSINT-нарушителя как функционирование абстрактной машины, оперирующей на графе знаний.

Ключевыми результатами работы являются:

1. Формализация процесса: предложена строгая модель, включающая множество свойств, техник (предикатов), состояние аналитика и четкие критерии прекращения OSINT-анализа; сформулировано и доказано необходимое и достаточное условие OSINT. Это позволяет декомпозировать сложную деятельность OSINT-аналитика на последовательность формальных операций с последующей автоматизацией.

2. Связь с классической теорией информационной безопасности: продемонстрировано, что OSINT-атаки нарушают не букву, а принцип модели Белла — Лападулы (мандатной модели), эксплуатируя семантический «скрытый канал вывода».

3. Практическая апробация: теоретическая модель была успешно транслирована в два прикладных фреймворка: OSINT Kill Chain для атакующих и Blue Team Playbook для защитников, — что доказывает ее адекватность и практическую ценность.

Предложенный фреймворк закладывает теоретическую основу для дальнейших исследований в области противодействия угрозам, исходящим из открытых источников, и может служить базой для стандартизации процессов OSINT-анализа и обучения специалистов.

Литература

1. **Xiong W., Legrand E., Åberg O., Lagerström R.** Cyber security threat modeling based on the MITRE Enterprise ATT&CK Matrix. *Software and Systems Modeling*, 2022, vol. 21, no. 1, pp. 157–177. doi:10.1007/s10270-021-00912-3
2. **Bratsas C., Anastasiadis E. K., Angelidis A. K., Ioannidis L., Kotsakis R., Ougiaroglou S.** Knowledge graphs and semantic Web tools in cyber threat intelligence: A systematic literature review. *Journal of Cybersecurity and Privacy*, 2024, vol. 4, no. 3, pp. 518–545. doi:10.3390/jcp4030026
3. **Browne T. O., Abedin M., Chowdhury M. J. M.** A systematic review on research utilising artificial intelligence for open source intelligence (OSINT) applications. *International Journal of Information Security*, 2024, vol. 23, no. 4, pp. 2911–2938. doi:10.1007/s10207-024-00877-3
4. **Rajamäki J.** OSINT on the Dark Web: Child abuse material investigations. *Information & Security*, 2022, vol. 53, pp. 21–32. doi:10.11610/isij.5302
5. **Sage E. C.** Shining a light on AIS Blackouts with maritime OSINT. *Frontiers in Computer Science*, 2023, vol. 5. doi:10.3389/fcomp.2023.1185760
6. **Granata D., Rak M.** Systematic analysis of automated threat modelling techniques: Comparison of open-source tools. *Software Quality Journal*, 2024, vol. 32, no. 1, pp. 125–161. doi:10.1007/s11219-023-09653-6
7. **Martins C., Medeiros I.** Generating quality threat intelligence leveraging OSINT and a cyber threat unified taxonomy. *ACM Transactions on Privacy and Security*, 2022, vol. 25, no. 3, pp. 1–39. doi:10.1145/3505232
8. **Gryzunov V., Gryzunova D.** *Problems of Providing Access to a Geographic Information System Processing Data of Different Degrees of Secrecy.* Lecture Notes on Data Engineering and Communications Technologies, 2022, vol. 73, pp. 191–198. doi:10.1007/978-981-16-3961-6_17
9. **Mukhopadhyay A., Luther K.** OSINT clinic: Co-designing AI-augmented collaborative OSINT investigations for vulnerability assessment. *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025, pp. 1–22. https://doi.org/10.1145/3706598.3713283
10. **Романова Н. Н., Грызунов В. В.** Исследование методом расширенного систематического обзора литературы E-SLR проблемы обеспечения безопасности персональных данных при использовании OSINT. *Вестник Дагестанского государственного технического университета. Технические науки*, 2024, т. 51, № 3, с. 130–144. doi:10.21822/2073-6185-2024-51-3-130-144, EDN: KUEPUU
11. **Romanova N., Gryzunov V.** OSINT model of privacy violator. *2025 International Russian Smart Industry Conference (SmartIndustryCon)*, Sochi, Russian Federation, 2025, pp. 929–933. doi:10.1109/SmartIndustryCon65166.2025.10985976
12. **Riebe T., Bäumler J., Kaufhold M. A., Reuter C.** Values and value conflicts in the context of OSINT technologies for cybersecurity incident response: A value sensitive design perspective. *Computer Supported Cooperative Work (CSCW)*, 2024, vol. 33, no. 2, pp. 205–251. doi:10.1007/s10606-024-09498-w
13. **Грызунов В. В.** Формирование условия гарантированного достижения цели деятельности информационной системой на базе операторного уравнения. *Информатизация и связь*, 2022, № 4, с. 67–74. doi:10.34219/2078-8320-2022-13-4-67-74, EDN: NGLZEW
14. **Battaglioni M., Mas-Machuca C., Solé-Gimeno A., Sanchez-Rola I.** Magic: A method for assessing cyber incidents occurrence. *IEEE Access*, 2022, vol. 10, pp. 73458–73473. doi:10.1109/ACCESS.2022.3188589
15. **Badhwar R.** *Simplified Approach to Calculate the Probability of a Cyber Event*. The CISO's Next Frontier: AI, Post-Quantum Cryptography and Advanced Security Paradigms. Cham, Springer International Publishing, 2021, pp. 353–359. doi:10.1007/978-3-030-81534-7_17
16. **Cai W., Yao H.** Research on information security risk assessment method based on fuzzy rule set. *Wireless Communications and Mobile Computing*, 2021, vol. 2021, Article ID 9663520. doi:10.1155/2021/9663520
17. **Ekstedt M., Afzal Z., Mukherjee P., Hacks S., Lagerström R.** Yet another cybersecurity risk assessment framework. *International Journal of Information Security*, 2023, vol. 22, no. 6, pp. 1713–1729. doi:10.1007/s10207-023-00713-y
18. **Yamin M. M., Ullah M., Ullah H., Katt B., Hijji M., Muhammad K.** Mapping tools for open source intelligence with cyber kill chain for adversarial aware security. *Mathematics*, 2022, vol. 10, no. 12, p. 2054. doi:10.3390/math10122054
19. **Гайдамакин Н. А.** Модели и метрики осведомленности в конфиденциальной информации. Ч. 1. Потенциальная осведомленность. *Прикладная дискретная математика*, 2023, № 61, с. 86–103. doi:10.17223/20710410/61/5, EDN: ZNDISS
20. **Девянин П. Н.** Результаты переработки уровней ролевого управления доступом и мандатного контроля целостности формальной модели управления доступом ОС Astra Linux. *Труды Института системного программирования РАН*, 2023, т. 35, № 5, с. 7–22. doi:10.15514/ISPRAS-2023-35(5)-1, EDN: OVDQUI
21. **Козачок В. И., Козачок А. В., Кочетков Е. В.** Многоуровневая модель политики безопасности управления доступом операционных систем семейства Windows. *Вопросы кибербезопасности*, 2021, № 1(41), с. 41–56. doi:10.21681/2311-3456-2021-1-41-56

UDC 004.056+004.62+004.94
doi:10.31799/1684-8853-2025-5-22-34
EDN: AYCIBD

A formal framework for the OSINT attacker and defender

V. V. Gryzunov[a], Dr. Sc., Tech., Associate Professor, orcid.org/0000-0003-4866-217X, viv1313r@mail.ru
[a]Saint Petersburg University of State Fire Service of Emercom of Russia, 149, Moskovsky Pr., 197198, Saint-Petersburg, Russian Federation

Introduction: The proliferation of OSINT in cybersecurity revealed a gap between practical tools and their theoretical conceptualization. Existing approaches lack a comprehensive model of the OSINT analysis process, which hinders formal automation, verification, and threat analysis. **Purpose:** To develop a formal model of an OSINT attacker, conceptualized as an abstract machine that describes the intelligence process through graph-based knowledge representation, inference logic, and resource constraints. **Results:** We propose a model of an OSINT attacker as an abstract machine that operates on an iteratively expanding knowledge graph. The OSINT task is formalized as a pathfinding problem from public properties to confidential ones. We formulate the necessary and sufficient condition for disclosing a confidential property and define the concept of triangulation through the search for independent (non-intersecting) paths within the graph. The analysis shows that OSINT attacks exploite an "inference covert channel", thereby violating the principles of classical security models, such as the Bell – LaPadula mandatory access control model. In addition, we introduce a formal stopping function that integrates practical criteria for analysis termination: goal achievement, resource depletion, diminishing returns, and detection risk. The study formalizes the concept of "independent sources". **Practical relevance:** We translate the theoretical model into two applied frameworks. First, the OSINT Kill Chain provides a step-by-step methodology for attackers (Red Team). Second, the Blue Team Playbook offers a mirror framework for defenders, describing how to audit a digital footprint and minimize risks. **Discussion:** For future research, the study proposes using the framework of formal grammars to model not only static relationships but also the dynamic tactics, techniques, and procedures of an attacker.

Keywords – OSINT, open-source intelligence, inference channel, attacker model, knowledge graph, framework, cyber threat intelligence, information security, Red Team, Blue Team, inference attack, mandatory access control model.

For citation: Gryzunov V. V. A formal framework for the OSINT attacker and defender. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 5, pp. 22–34 (In Russian). doi:10.31799/1684-8853-2025-5-22-34, EDN: AYCIBD

References

1. Xiong W., Legrand E., Åberg O., Lagerström R. Cyber security threat modeling based on the MITRE Enterprise ATT & CK Matrix. *Software and Systems Modeling*, 2022, vol. 21, no. 1, pp. 157–177. doi:10.1007/s10270-021-00912-3
2. Bratsas C., Anastasiadis E. K., Angelidis A. K., Ioannidis L., Kotsakis R., Ougiaroglou S. Knowledge graphs and semantic Web tools in cyber threat intelligence: A systematic literature review. *Journal of Cybersecurity and Privacy*, 2024, vol. 4, no. 3, pp. 518–545. doi:10.3390/jcp4030026
3. Browne T. O., Abedin M., Chowdhury M. J. M. A systematic review on research utilising artificial intelligence for open source intelligence (OSINT) applications. *International Journal of Information Security*, 2024, vol. 23, no. 4, pp. 2911–2938. doi:10.1007/s10207-024-00877-3
4. Rajamäki J. OSINT on the Dark Web: Child abuse material investigations. *Information & Security*, 2022, vol. 53, pp. 21–32. doi:10.11610/isij.5302
5. Sage E. C. Shining a light on AIS Blackouts with maritime OSINT. *Frontiers in Computer Science*, 2023, vol. 5. doi:10.3389/fcomp.2023.1185760
6. Granata D., Rak M. Systematic analysis of automated threat modelling techniques: Comparison of open-source tools. *Software Quality Journal*, 2024, vol. 32, no. 1, pp. 125–161. doi:10.1007/s11219-023-09653-6
7. Martins C., Medeiros I. Generating quality threat intelligence leveraging OSINT and a cyber threat unified taxonomy. *ACM Transactions on Privacy and Security*, 2022, vol. 25, no. 3, pp. 1–39. doi:10.1145/3505232
8. Gryzunov V., Gryzunova D. *Problems of Providing Access to a Geographic Information System Processing Data of Different Degrees of Secrecy*. In: *Lecture Notes on Data Engineering and Communications Technologies*, 2022, vol. 73, pp. 191–198. doi:10.1007/978-981-16-3961-6_17
9. Mukhopadhyay A., Luther K. OSINT clinic: Co-designing AI-augmented collaborative OSINT investigations for vulnerability assessment. *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025, pp. 1–22. https://doi.org/10.1145/3706598.3713283
10. Romanova N. N., Gryzunov V. V. Research by the method of an extended systematical literature review E-SLR the problem of ensuring the security of personal data when using OSINT. *Herald of Dagestan State Technical University. Technical Sciences*, 2024, vol. 51, no. 3, pp. 130–144 (In Russian). doi:10.21822/2073-6185-2024-51-3-130-144, EDN: KUEPUU
11. Romanova N., Gryzunov V. OSINT model of privacy violator. *2025 International Russian Smart Industry Conference (SmartIndustryCon)*, Sochi, Russian Federation, 2025, pp. 929–933. doi:10.1109/SmartIndustryCon65166.2025.10985976
12. Riebe T., Bäumler J., Kaufhold M. A., Reuter C. Values and value conflicts in the context of OSINT technologies for cybersecurity incident response: A value sensitive design perspective. *Computer Supported Cooperative Work (CSCW)*, 2024, vol. 33, no. 2, pp. 205–251. doi:10.1007/s10606-024-09498-w
13. Gryzunov V. V. Formation of a condition for guaranteed achievement of the activity goal of an information system based on an operator. *Informatizatsiya i svyaz*, 2022, no. 4, pp. 67–74 (In Russian). doi:10.34219/2078-8320-2022-13-4-67-74, EDN: NGLZEW
14. Battaglioni M., Mas-Machuca C., Solé-Gimeno A., Sanchez-Rola I. Magic: A method for assessing cyber incidents occurrence. *IEEE Access*, 2022, vol. 10, pp. 73458–73473. doi:10.1109/ACCESS.2022.3188589
15. Badhwar R. *Simplified Approach to Calculate the Probability of a Cyber Event*. In: *The CISO's Next Frontier: AI, Post-Quantum Cryptography and Advanced Security Paradigms*. Cham, Springer International Publishing, 2021, pp. 353–359. doi:10.1007/978-3-030-81534-7_17
16. Cai W., Yao H. Research on information security risk assessment method based on fuzzy rule set. *Wireless Communications and Mobile Computing*, 2021, vol. 2021, Article ID 9663520. doi:10.1155/2021/9663520
17. Ekstedt M., Afzal Z., Mukherjee P., Hacks S., Lagerström R. Yet another cybersecurity risk assessment framework. *International Journal of Information Security*, 2023, vol. 22, no. 6, pp. 1713–1729. doi:10.1007/s10207-023-00713-y
18. Yamin M. M., Ullah M., Ullah H., Katt B., Hijji M., Muhammad K. Mapping tools for open source intelligence with cyber kill chain for adversarial aware security. *Mathematics*, 2022, vol. 10, no. 12, p. 2054. doi:10.3390/math10122054
19. Gaidamakin N. A. The model and metrics of awareness in confidentiali. Part 1. Potential awareness. *Applied Discrete Mathematics*, 2023, no. 61, pp. 86–103 (In Russian). doi:10.17223/20710410/61/5, EDN: ZNDISS
20. Devyanin P. N. The results of reworking the levels of role-based access control and mandatory integrity control of the formal model of access control in Astra Linux. *Proceedings of the Institute for System Programming of the RAS*, 2023, vol. 35, no. 5, pp. 7–22 (In Russian). doi:10.15514/ISPRAS-2023-35(5)-1, EDN: OVDQUI
21. Kozachok V. I., Kozachok A. V., Kochetkov E. V. Multi-level policy model access control security operating systems of the Windows family. *Cybersecurity Issues*, 2021, no. 1(41), pp. 41–56 (In Russian). doi:10.21681/2311-3456-2021-1-41-56