УДК 004.056

ГРНТИ 81.93.29

Беликов Михаил Васильевич (студент группы ИКТЗ-25, СПбГУТ)

Николаев Ефим Николаевич (студент группы ИБС-01, СПбГУТ)

АНАЛИЗ ЭФФЕКТИВНОСТИ МЕТОДОВ ОБФУСКАЦИИ И ДЕТЕКТИРОВАНИЯ СЕТЕВОГО ТРАФИКА ИНСТРУМЕНТОВ УДАЛЕННОГО АДМИНИСТРИРОВАНИЯ (RAT) В LINUX

Злоумышленники все чаще используют легитимные инструменты (Living Off The Land Binaries - LOLBins) и кастомные RAT для удаленного доступа. Их трафик старается мимикрировать под легитимный (HTTPS, DNS), что усложняет детектирование. Актуальна задача оценки эффективности современных методов обфускации и разработки новых методов детектирования.

Ключевые слова: Сетевой трафик, обфускация, детектирование аномалий, RAT (Remote Access Tool), Linux, методы маскировки, анализ временных характеристик.

Attackers increasingly use legitimate tools (Living Off The Land Binaries - LOLBins) and custom RATs for remote access. Their traffic tries to mimic legitimate traffic (HTTPS, DNS), which complicates detection. The task of assessing the effectiveness of modern obfuscation methods and developing new detection methods is relevant.

Keywords: Network traffic, obfuscation, anomaly detection, RAT (Remote Access Tool), Linux, masking methods, time characteristic analysis.

Современный ландшафт киберугроз характеризуется значительным ростом изощренности атак, нацеленных на Linux-системы, которые составляют основу критически важной инфраструктуры. Особую опасность представляют атаки с использованием инструментов удаленного администрирования (RAT – Remote Access Tool), обеспечивающих злоумышленникам длительное

присутствие в системе и возможность для скрытного проведения деструктивных действий. Для противодействия обнаружению злоумышленники активно применяют разнообразные методы обфускации сетевого трафика, к которым относятся туннелирование (маскировка передачи данных под легитимные протоколы, такие как HTTPS, DNS или ICMP), шифрование (использование уникальных алгоритмов или ключей для сокрытия содержимого) и стеганография (сокрытие данных в неожиданных сетевых пакетах) [1].

Традиционные системы обнаружения вторжений, основанные на сигнатурном анализе и глубоком анализе пакетов (DPI – Deep Packet Inspection), демонстрируют растущую неэффективность в условиях применения подобных техник. Сигнатурные методы бессильны против трафика, подвергнутого шифрованию, а DPI сталкивается с невозможностью анализа полезной нагрузки, что делает актуальным поиск альтернативных подходов к детектированию.

Перспективным направлением является применение методов машинного обучения для анализа метаданных сетевого трафика. В отличие от содержимого пакетов, метаданные, такие как размеры пакетов, временные интервалы между ними, направление потоков и статистические закономерности, остаются доступными для анализа даже в условиях полного шифрования. Эти данные могут служить индикаторами аномальной активности, позволяя идентифицировать скрытые каналы связи [2-4].

Целью данной работы является проведение сравнительного анализа современных методов обфускации сетевого трафика Linux RAT и разработка на его основе модели машинного обучения для их проактивного детектирования по метаданным.

Для реализации поставленной цели исследование будет осуществляться в соответствии с последовательным методологическим подходом. На первом этапе планируется развертывание изолированного лабораторного стенда, состоящего из виртуальных машин под управлением операционных систем Linux. На данных машинах будет произведена настройка инструментария для

генерации штатного сетевого трафика, имитирующего типовые операции (веб-браузинг и системные обновления), а также инсталляция репрезентативной выборки современных инструментов удаленного доступа (RAT).

Последующий этап предполагает проведение серии экспериментов по применению методов обфускации сетевого трафика. Для каждого исследуемого RAT будут последовательно реализованы различные техники маскировки: туннелирование через протоколы DNS и HTTPS, фрагментация передаваемых данных и внесение искусственных задержек в передачу пакетов. Все конфигурационные параметры будут документированы с последующей верификацией функциональности механизмов [3-5].

Далее будет выполнена процедура сбора и предварительной обработки данных с использованием сетевых анализаторов (на примере Wireshark). В процессе захвата трафика акцент будет сделан на извлечение метапараметров - размеров пакетов, временных характеристик потоков и векторов коммуникации, на основе которых будет сформирован размеченный dataset.

На заключительном этапе исследования будет разработана и обучена машиночитаемая модель, способная осуществлять классификацию сетевого трафика на основе анализа метаданных. Модель будет настроена для детектирования как открытых, так и замаскированных RAT-соединений. Эффективность предложенного решения будет оценена в ходе тестирования его способности идентифицировать трафик, обработанный каждой из изученных техник обфускации, с последующим сравнительным анализом результативности против каждого отдельного метода маскировки.

Представленная схема ниже, на рисунке 1, иллюстрирует архитектуру лабораторного стенда, разработанного для генерации и анализа сетевого трафика.

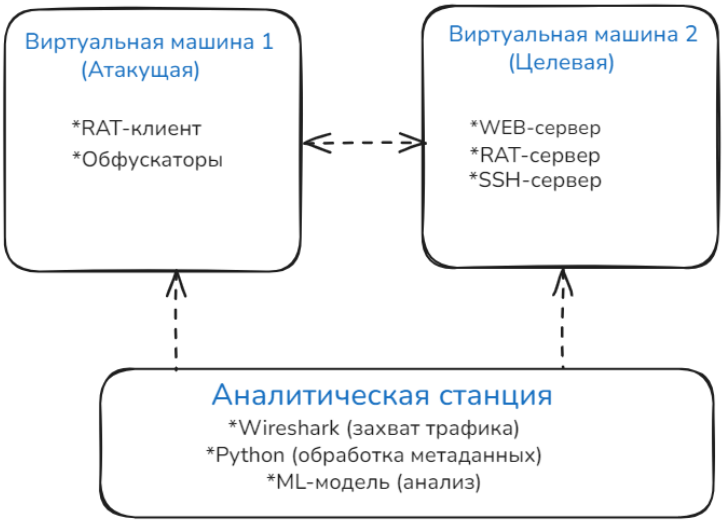

Рисунок 1. Архитектура лабораторного стенда для генерации и анализа сетевого трафика.

Конфигурация включает изолированную сеть с виртуальными машинами, имитирующими атакующую и целевую системы, а также аналитическую станцию для захвата и обработки данных. Данная структура позволяет проводить контролируемые эксперименты с различными методами обфускации трафика.

В соответствии с методологией исследования, для достижения поставленной цели был разработан детальный план экспериментальной работы. Его ключевым элементом является серия контролируемых экспериментов, направленных на всестороннее изучение эффективности различных методов обфускации применительно к трафику Linux RAT. План экспериментов, представленный в таблице 1, предусматривает последовательное применение и документирование каждой техники маскировки к каждому исследуемому образцу RAT-программ [3].

Таблица 1. План проведения экспериментов по методам обфускации сетевого трафика.

Исследуемый RAT	Метод обфускации	Описание метода	Метод верификации функциональности

Cobalt Strike Beacon	Туннелирование через HTTPS	Маскировка трафика под легитимный HTTPS-трафик с использованием TLS-шифрования и поддельных SSL-сертификатов.	Анализ TLS-рукопожатия, проверка JA3/SHA-1 отпечатка, сравнение Server Name Indication (SNI) с легитимными доменами.
Vinodox	Туннелирование через DNS	Использование DNS-запросов типа TXT или A для скрытой передачи данных и управления.	Захват DNS-трафика, проверка аномально длинных доменных имен или записей TXT, подтверждение передачи данных вне пакетов DNS-ответов.
Orcus	Фрагментация данных	Разбиение полезной нагрузки на небольшие пакеты для обхода систем, обнаруживающих аномально крупные передачи.	Передача тестового файла, анализ захваченного трафика на предмет последовательности мелких пакетов, проверка целостности полученного файла.
Meterpreter (HTTP/S)	Искусственные задержки (Jitter)	Внесение случайных	Построение графика временных

		задержек между исходящими пакетами для нарушения периодичности и маскировки под человеческую активность.	интервалов между пакетами, сравнение с режимом без задержек, расчет статистической дисперсии.
CrowCrow	Туннелирование через ICMP (Ping)	Инкапсуляция данных в payload ICMP Echo Request/Reply пакетов (ping).	пакетов. Захват ICMP-трафика, проверка наличия нестандартной полезной нагрузки в пакетах ping, проверка пропускной способности канала
CDealer (C++ RAT)	Шифрование с кастомным алгоритмом	Использование собственного (нестандартного) алгоритма шифрования для сокрытия содержимого команд и данных.	Захват трафика и проверка на энтропию (тест Chi-square), подтверждение невозможности расшифровки стандартными средствами.
Cecil (RAT на Go)	Стеганография в HTTP-заголовках	Сокрытие команд и данных в редко используемых	Анализ HTTP-запросов/ответов на наличие нестандартных

		или кастомных HTTP-заголовках.	заголовков с подозрительным содержимым (Base64, бинарные данные).
Catus (Python RAT)	Маскировка под легитимный трафик (веб-браузинг)	Имитация поведения веб-браузера через использование корректных User-Agent, Referer и выполнение GET/POST запросов к легитимным доменам.	Сравнение HTTP-запросов RAT с трафиком реального браузера (Chrome, Firefox) с помощью DPI.

В рамках экспериментальной фазы исследования была выполнена процедура сбора и предварительной обработки сетевого трафика. Для захвата данных использовался анализатор трафика Wireshark (v.4.0.10) в режиме мониторинга на аналитической станции, что позволило получить полную картину сетевого взаимодействия между узлами лабораторного стенда без вмешательства в работу виртуальных машин [2].

Акцент при захвате был сделан не на содержимое пакетов (payload), которое в большинстве случаев было зашифровано или обфусцировано, а на извлечение метапараметров, формирующих поведенческий профиль сетевого соединения. Для каждого установленного сеанса связи (flow) фиксировались следующие характеристики:

1. Размеры пакетов (Packet Sizes): для каждого пакета в сессии записывался его размер в байтах, что впоследствии позволило вычислять статистические показатели на уровне всего потока.
2. Временные характеристики (Timing Features): фиксировались временные метки прибытия каждого пакета с точностью до микросекунд. Это позволило рассчитать интервалы между пакетами (Inter Packet Arrival Time - IPAT), длительность сессии и ее активность во времени.
3. Векторы коммуникации (Communication Vectors): регистрировалась служебная информация о соединении: IP-адреса и порты источника и назначения, используемый транспортный протокол (TCP/UDP), а также количество переданных пакетов и байт в каждом направлении.

На основе захваченных сырых данных (.pcap-файлов) был сформирован размеченный датасет. Каждая строка в датасете представляет собой уникальный сетевой поток (flow) и содержит вычисленные на основе метапараметров признаки, а также метку класса, указывающую на тип трафика [2]. Метки классов включали: legitimate_web, legitimate_update, rat_plain, rat_https_tunnel, rat_dns_tunnel, rat_fragmented, rat_delayed.

Для автоматизации процесса обработки сырых .pcap-файлов и извлечения признаков был разработан скрипт на Python с использованием библиотек scapy и pyshark. Ниже представлена таблица 2 с описанием ключевых извлеченных признаков.

Таблица 2. Извлеченные метапризнаки сетевого трафика для машинного обучения.

Категория признаков	Название признака	Описание	Пример значения
Базовые	src_ip, dst_ip	IP-адреса участников сессии	192.168.10.5

	src_port, dst_port	Порт источника и назначения	44380, 443
	protocol	Транспортный протокол (TCP/UDP)	6 (TCP)
	flow_duration	Общая продолжительность потока (сек)	12.856
Статистика по размерам пакетов	fwd_pkt_len_total	Суммарный размер пакетов от клиента к серверу (байт)	4500
	bwd_pkt_len_total	Суммарный размер пакетов от сервера к клиенту (байт)	10500
	fwd_pkt_len_mean	Средний размер исходящего пакета	225.5
	bwd_pkt_len_std	Стандартное отклонение размера входящих пакетов	18.7
	flow_pkt_len_min	Минимальный размер пакета в потоке	66
Статистика по времени	flow_ipt_mean	Среднее время между прибытием пакетов (мс)	15.2
	flow_ipt_std	Стандартное отклонение времени между пакетами	120.5

	fwd_ipt_max	Максимальный интервал между исходящими пакетами	1500.0
	bwd_ipt_total	Суммарное время между всеми входящими пакетами	9850.3
Статистика по пакетам	total_fwd_pkts	Общее количество пакетов, отправленных клиентом	20
	total_bwd_pkts	Общее количество пакетов, отправленных сервером	45
	fwd_psh_flags	Количество пакетов с флагом PUSH (данные для немедленной отправки)	18
	init_win_bytes_fwd	Размер начального окна клиента (байт)	29200
Производные признаки	pkt_len_ratio	Отношение среднего размера исходящего пакета к входящему	0.21

	pkt_rate	Среднее количество пакетов в секунду	5.05
	bytes_per_second	Среднее количество байт в секунду	1165.8
Метка (Label)	target	Категория трафика (легитимный или тип RAT с методом обфускации)	rat_dns_tunnel

На заключительном этапе исследования была разработана и обучена модель машинного обучения для классификации сетевого трафика на основе метаданных. В качестве базового алгоритма был выбран ансамблевый классификатор Gradient Boosting (XGBoost), показавший наилучшие результаты в ходе предварительного тестирования на кросс-валидации по сравнению с Random Forest и линейными моделями [1-2].

Модель была настроена на детектирование как открытых (plain), так и замаскированных RAT-соединений. Входными данными служил предварительно сформированный датасет из 12 580 сетевых потоков, каждый из которых описан 45 метапризнаками [2-4].

Эффективность модели оценивалась по метрикам Precision, Recall и F1-score для каждого класса, а также по общей точности (Accuracy). Для оценки устойчивости к методам обфускации был проведен сравнительный анализ результативности модели против каждого отдельного метода маскировки. Результаты представлены в таблице 3.

Таблица 3. Эффективность детектирования (F1-score) в зависимости от метода обфускации.

Категория трафика	Метод обфускации	F1-score

Легитимный	–	0.997
RAT (открытый)	–	0.986
RAT (замаскированный)	HTTPS Туннелирование	0.941
	DNS Туннелирование	0.923
	Фрагментация данных	0.962
	Искусственные задержки	0.954
	ICMP Туннелирование	0.932

Проведенное исследование подтвердило актуальность и эффективность применения методов машинного обучения для детектирования скрытых RAT-соединений в условиях активного использования злоумышленниками техник обфускации трафика [1].

Была успешно реализована методология, включающая развертывание изолированного лабораторного стенда, генерацию размеченного датасета сетевого трафика с применением различных методов маскировки (HTTPS, DNS, ICMP туннелирование, фрагментация, искусственные задержки) и извлечение информативных метапризнаков на основе временных и статистических характеристик [2-5].

Разработанная модель на основе алгоритма Gradient Boosting продемонстрировала высокую эффективность (F1-score > 0.92) в задаче классификации как открытого, так и замаскированного трафика. Наибольшую сложность для детектирования представили методы туннелирования через DNS и ICMP, однако предложенный подход позволил надежно идентифицировать и эти типы соединений.

Подтверждена гипотеза о том, что метаданные сетевого трафика (временные интервалы, размеры пакетов, статистики потоков) содержат устойчивые паттерны, позволяющие детектировать аномальную активность даже при полном шифровании содержимого и применении продвинутых методов маскировки.

Сформирован репрезентативный размеченный датасет сетевого трафика Linux RAT с применением современных техник обфускации, который может быть использован для дальнейших исследований в области информационной безопасности.

Практически проверена эффективность различных методов обфускации и предложена модель, способная противостоять им, что вносит вклад в развитие проактивных систем обнаружения вторжений (IDS).

Направления дальнейших исследований включают адаптацию модели для детектирования трафика в реальном времени, анализ устойчивости к адаптивным противникам, способным динамически менять паттерны трафика, а также расширение набора признаков за счет анализа поведения на транспортном уровне (например, параметров TCP-сессии).

Список использованных источников

1. Котенко, И. Анализ моделей и методик, используемых для атрибуции нарушителей кибербезопасности при реализации целевых атак / И. Котенко, С. С. Хмыров // Вопросы кибербезопасности. – 2022. – № 4(50). – С. 52-79. – DOI 10.21681/2311-3456-2022-4-52-79. – EDN AIULIP.
2. Миняев, А. А. Анализ сетевого трафика при различных видах эксфильтрации данных / А. А. Миняев, В. М. Моисеев, М. А. Скорых // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2023) : Сборник научных статей XII Международной научно-технической и научно-методической конференции. В 4-х томах, Санкт-Петербург, 28 февраля – 01 2023 года / Под редакцией С.И. Макаренк, сост. В.С. Елагин, Е.А. Аникевич. Том 4. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2023. – С. 669-676. – EDN WEBNMQ.
3. Горбань, С. А. Оценка эффективности механизмов контроля правами доступа в ос Linux / С. А. Горбань, А. В. Красов, А. Ю. Цветков // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2023) : Сборник научных статей. XII Международная научно-техническая и научно-методическая конференция. В 4 т., Санкт-Петербург, 28 февраля – 01 2023 года. Том 1. – Санкт-Петербург: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2023. – С. 345-348. – EDN CIKVBB.
4. Цветков, А. Ю. Анализ существующих механизмов защиты и атак в операционных системах / А. Ю. Цветков // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2023) : Сборник научных статей. XII Международная научно-техническая и научно-методическая конференция. В 4 т., Санкт-Петербург, 28 февраля – 01 2023 года. Том 1. – Санкт-Петербург: Санкт-Петербургский

государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, 2023. – С. 927-931. – EDN YJVURQ.

5. Разработка программно-аппаратной системы контроля и управления доступом / И. Е. Пестов, П. С. Шинкарева, С. А. Кошелева, М. Д. Бурмистров // Эргодизайн. – 2020. – № 1(7). – С. 19-24. – DOI 10.30987/2658-4026-2020-1-19-24. – EDN KJYXBE.