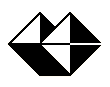
**ООО «NNN»**



УТВЕРЖДАЮ

Главный инженер ООО «»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Я.О.Сергеевич

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2025 г.

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

Документация по разработке приложения

г. Орск

2025 г.

|  |  |  |
| --- | --- | --- |
| № п/п | Перечень основных исходных данных и требований | Содержание данных и требований |
| **1 ОБЩИЕ СВЕДЕНИЯ** | | |
| 1.1 | Основание для проектирования | Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации», ФЗ № 184-ФЗ «О техническом регулировании»; другие нормативные документы по ИТ и ИБ. Составление ТЗ проводится по ГОСТ 34.602-89 (ТЗ на создание АС). Учитываются ГОСТ 34.201-89, ГОСТ 34.003-90, ГОСТ 34.601-90, ГОСТ 34.603-92, а также требования ФСТЭК (приказы по защите ГИС) и Роскомнадзора (методики по ПДн). При обосновании проекта учитываются аналитические отчёты (рост рынка NTA прогнозируется до $27 млрд к 2035 г. с CAGR ≈15%), что подчёркивает актуальность разработки.Начало формы  Конец формы  Начало формы  Конец формы  Начало формы  Конец формы  Начало формы  Конец формы |
| 1.2 | Заказчик |  |
| 1.3 | Местоположение проектируемых объектов | Решение устанавливается на узлах сети заказчика: персональных компьютерах, серверах и маршрутизаторах (в том числе на базе Linux/OpenWrt) в офисах и на удалённых площадках. Оборудование должно иметь доступ к отслеживаемому трафику (SPAN/TAP) и стабильное сетевое подключение. Предусмотрена работа и в виртуализированных средах (Docker/Kubernetes). Поддерживается доступ к системе управления через REST/API или веб-интерфейс из административной сети. |
| 1.4 | Вид работ | Разработка комплексной системы захвата и анализа сетевого трафика «Traffic Analyzer». Работы включают: системный анализ, проектирование архитектуры, программирование модулей (захват, обработка, ML-анализ), создание базы данных и интерфейсов, интеграцию, тестирование и внедрение. Также проводятся экспертиза и отладка системы в условиях заказчика. |
| 1.5 | Исполнитель | Выбор Исполнителя проекта осуществляется по результатам конкурса. |
| 1.6 | Срок реализации проекта | Источник финансирования — собственные средства Заказчика. Порядок финансирования определяется Заказчиком. |
| 1.7 | Источник финансирования и порядок финансирования. | Источник финансирования – собственные средства заказчика  Порядок финансирования определяется Заказчиком. |
| 1.8 | Стадийность проектирования | Проект разрабатывается по стадиям: ТЭО (технико-экономическое обоснование), проектная документация (архитектура, технический проект) и рабочая документация (код, инструкции). Формат документов – проектно-рабочая документация по ГОСТ 34.601-90. Для испытаний применяется ГОСТ 34.603-92 (приёмка ПО). Между стадиями проводятся согласования с заказчиком. |
| **2 НАЗНАЧЕНИЕ И ЦЕЛИ СОЗДАНИЯ ПЛАТФОРМЫ** | | |
| 2.1 | Назначение платформы | «Traffic Analyzer» предназначена для непрерывного мониторинга, захвата и анализа сетевого трафика в реальном времени. Модуль захвата пакетов реализован на Python с помощью библиотеки Scapy. NFStream (содержит DPI-модуль nDPI) обеспечивает агрегацию потоков и идентификацию приложений даже в зашифрованном трафике. Система автоматически классифицирует трафик по типам сервисов (Web, VoIP, DNS и др.), VPN (OpenVPN, WireGuard и др.) и обнаруживает аномальные/вредоносные активности с помощью обученных ML-моделей. Интерфейс взаимодействия – консольный CLI и/или веб-панель. Поддерживается распределённое хранение данных (например, с Elastic Stack или ClickHouse) для масштабируемости. |
| 2.2 | Цели создания платформы | Обеспечить централизованный сбор и консолидацию данных о сетевом трафике со всех точек сети («единое окно» мониторинга). Увеличить информированность о состоянии сети: своевременно выявлять угрозы, аномалии, узкие места. Оптимизировать пропускную способность и конфигурацию сети на основе детализированной аналитики. Поддержать принятие решений ИТ-отделом через формирование отчетов и визуализаций. Учитывая рост киберугроз и сложности современных сетей, создать инструмент для быстрой реакции на инциденты |
| 2.3 | Основной функционал для головной организации | Для администрации: генерация сводных отчётов по сети (по времени, сегментам, сервисам); дашборды с ключевыми метриками (загруженность, задержки, количество соединений); сравнительный анализ по филиалам/узлам; интеграция данных с SIEM/BI/CMDB через API; экспорт аналитики в CSV/JSON. Предусмотрено распределение прав доступа (на чтение/управление) и настройка расписания отчётов. |
| 2.4 | Основной функционал для конечного пользователя | Для сетевых инженеров: непрерывный мониторинг трафика с моментальным оповещением о превышении порогов (таймуты, объемы); детализация трафика по приложениям и пользователям; фильтрация и поиск по IP/портам; формирование on-demand отчётов в формате CSV/PDF; удобный CLI с подсказками и запросами к API для ad-hoc-анализа сети. |
| 2.5 | Объекты контроля | Объекты контроля – сетевые интерфейсы ПК и маршрутизаторов заказчика. Система собирает параметры трафика: IP-адреса источника/приёмника, номера портов, протоколы (TCP/UDP/ICMP и др.), TCP-флаги, объём и размер пакетов, интервалы между ними. С формируемыми на их основе потоками (flows) связаны признаки (пропускная способность, длительность потока, статистика пакетов), используемые для классификации. На выходе – категоризация трафика (например, Web-сервис, VPN-поток, вредоносный трафик). |
| 2.6 | Технические условия подключения | Стандартная Ethernet-сеть (10/100/1000 Мбит/с), поддержка IPv4/IPv6. Захват пакетов через raw-сокеты или libpcap (Scapy). NFStream обеспечивает DPI на уровне L7 (с nDPI. Не требуются специализированные устройства – система использует mirror-порт коммутатора или TAP. Связь с внешними сервисами (БД, SIEM) – по TCP/IP. Сеть заказчика должна позволять приложению слушать трафик (необходимо прописать права доступа или режим promisc). |
|  |  |  |
| **3 ХАРАКТЕРИСТИКИ ОБЪЕКТА** | | |
| 3.1 | Краткие сведения об объекте | Объект автоматизации – программно-аппаратный комплекс мониторинга сети. Состоит из устройств-захватчиков трафика (ПC/маршрутизаторы) и центрального сервера анализа. Основные функции: захват и агрегация трафика в потоки, извлечение признаков и классификация с помощью ML. Система обеспечивает сбор статистики сетевого поведения для последующей автоматической оценки. |
| 3.2 | Сведения об условиях эксплуатации объекта автоматизации | Эксплуатация – в нормальных условиях офисов/серверных (0–40 °C, влажность до 80%). Не требует специальных помещений. Нужно стабильное электропитание (220 В) и подключение к сети. Система рассчитана на круглосуточную работу: готова переносить потерю пакетов (≤1%), задержки (≤200 мс) и пиковые нагрузки (до миллионов пакетов/с) без сбоя. Рекомендуется ИБП для ключевого оборудования. |
| 3.3 | Контроль и учет трафика | Осуществляется на уровне пакетов и потоков. Система формирует статистику по каждому потоку: скорость передачи, количество соединений, объём данных. Используется DPI для подсчёта трафика по приложениям. Формируются журналы событий безопасности. По обнаруженным сервисам и аномалиям генерируются отчёты. Ведётся учёт трафика по зонам сети и по IP-адресам в допустимых пределах. |
| **4 ТРЕБОВАНИЯ К ПЛАТФОРМЕ** | | |
| 4.1 | Общие требования | Приложение работает в реальном времени (задержка анализа <1 с). Использует Python 3.8+ (желательно 3.10) на Linux/Windows. Кроссплатформенно: поддерживает основные дистрибутивы (Ubuntu, CentOS) и Windows Server. Оптимизировано для многоядерных CPU. Компоненты – с открытыми лицензиями (BSD/MIT/GPL). Код соответствует стандартам качества (PEP8, CI/CD). |
| 4.2 | Требования к структуре и функционированию приложения | Модульная архитектура: отдельные модули захвата (Scapy), обработки (NFStream), выделения признаков, классификации (sklearn, PyTorch), формирования отчётов и интерфейсов. Модули связаны через API/сообщения (например, Kafka). Используются стандартные форматы данных (JSON, CSV). Возможность гибкого расширения: подключение новых алгоритмов или интерфейсов без остановки системы. |
| 4.3 | Требования к компонентам анализа | Модульная архитектура: отдельные модули захвата (Scapy), обработки (NFStream), выделения признаков, классификации (sklearn, PyTorch), формирования отчётов и интерфейсов. Модули связаны через API/сообщения (например, Kafka). Используются стандартные форматы данных (JSON, CSV). Возможность гибкого расширения: подключение новых алгоритмов или интерфейсов без остановки системы. |
| 4.4 | Требования к способам и средствам связи | Сеть – Ethernet/Wi-Fi, стеки TCP/IP. Приложение передаёт данные API (REST/HTTPS) и по протоколу SSH. Интеграция с внешними системами через HTTP API и коннекторы (например, Kafka Producers, Elasticsearch Clients). Все каналы связи защищены TLS/SSL. Необходимы соответствующие сетевые права (открытые порты, VPN) для обмена данными. |
| 4.5 | Требования к режимам функционирования приложения | Режимы: 1) Непрерывный мониторинг трафика с автоматической обработкой входящих пакетов. 2) Анализ сохранённых файлов (PCAP) – оффлайн-режим. В обоих случаях система работает автоматом. При возникновении критических событий генерируются оповещения. Предусмотрена «горячая» перезагрузка – изменения конфигурации без остановки всех сервисов. |
| 4.6 | Требования по диагностированию | Предусмотрено логирование ошибок и событий (с метками времени, деталями модулей). Реализовано самотестирование: периодическая проверка статуса всех модулей (availability check), памяти, БД. Администратор может запросить статус системы (health check API). Детальный лог (INFO/ERROR) сохраняется для последующего анализа. При критических сбоях – уведомление на электронную почту или в консоль. |
| 4.7 | Перспективы развития, модернизации приложения | Система проектируется масштабируемой: может запускаться в кластере (несколько узлов-захватчиков с центральным анализом). Возможна доработка: добавление GUI-дашборда, расширение поддержки протоколов (QUIC, IoT), интеграция с ML-платформами. Архитектура допускает подключение новых моделей и алгоритмов без переработки основных модулей. Расширения разрабатываются как плагины. |
| 4.8 | Требования к надежности Платформы | Соответствие ГОСТ 24.701-86 (показатели MTBF, среднее время наработки). Целевой показатель безотказной работы – ≥99% времени. Предусматривается отказоустойчивость: автоматический перезапуск упавших сервисов, резервирование критичных компонентов (дублирование NIC, репликация БД). Регулярные тесты устойчивости (load-testing) подтверждают стабильность. |
| 4.9 | Требования безопасности | Защита от несанкционированного доступа: применяются шифрование каналов (TLS/SSL), аутентификация и ролевые права. Хранение паролей – в виде хешей. Для персональных данных (IP) реализуется анонимизация по стандартам ФЗ-152. Система соответствует ISO/IEC 27001 по ИБ[[8]](https://en.wikipedia.org/wiki/ISO/IEC_27001#:~:text=ISO%2FIEC%2027001%20is%20an%20information,national%20variants%20of%20the%20standard): регламентирована политика доступа, ведётся аудит. Реализованы механизмы обнаружения и предотвращения внедрения вредоносного кода. |
| 4.10 | Требования к эргономике и технической эстетике | Интерфейс удобен и интуитивен. CLI – с ясными сообщениями и помощью (help). Веб-интерфейс – с минималистичным, читаемым дизайном (таблицы, графики). Цвета используются функционально (подсветка аномалий). Соответствие руководствам по эргономике (ГОСТ Р 12.2.032). Документы – на русском, со структурой и терминологией согласно отраслевым стандартам. |
| 4.11 | Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов приложения | ПО сопровождается русскоязычной документацией (руководства пользователя и администратора). Обновления выпускаются через Git (релизы с описаниями изменений). Установка выполняется скриптом (pip) или контейнером Docker. Предусмотрены процедуры резервного копирования, архивирования данных и восстановления после сбоев. Поддержка пользователей – через систему заявок. |
| 4.12 | Требования к защите информации от несанкционированного доступа | Реализована многоуровневая защита: логин/пароль, LDAP/AD-интеграция по возможности. Ограничение доступа на уровне ОС и приложения. Журналы доступа к системе хранятся в зашифрованном виде. Связь между модулями – по защищённым каналам (SSH, VPN). Система устойчива к перебору паролей и внедрению SQL-инъекций (используются ORM и защищённые библиотеки). |
| 4.13 | Требования по сохранности информации | Регулярное резервное копирование БД и настроек. При недоступности хранилища трафик сначала буферизуется на диск, а затем отправляется (аналог Packetbeat, который сохранил данные при отказе). Полномасштабное восстановление из резервной копии производится без потери метрик. Арбитраж хранимых данных гарантирует целостность (контрольные суммы, репликация). |
| 4.14 | Требования к средствам защиты от внешних воздействий | Защита от физического и электрического воздействия: экранированные кабели, ИБП. Приложение устойчиво к сетевым помехам – сохраняет корректность при перепадах пропускной способности или кратковременных разрывах. При аварии системы возможен «чистый» выход с сохранением журнала. Средства защиты согласованы с нормами ГОСТ по EMC. |
| 4.15 | Требования по стандартизации и унификации | Используются открытые стандартные протоколы (Ethernet, IP, JSON, HTTP). Исходный код оформляется по PEP8/PyLint. Базы данных – SQL/NoSQL по стандартам. При необходимости могут быть подтверждены сертификатами соответствия (ГОСТ Р ИСО/МЭК 27001, FSTEC). Все сторонние компоненты – с открытым кодом и широко распространёнными лицензиями. |
| 4.16 | Требования к гарантийному обслуживанию | Гарантийный срок – 12 месяцев. В течение гарантии исправляются выявленные дефекты, выпускаются обновления безопасности. Реагирование на запросы заказчика – в течение 1 рабочего дня. Предоставляется поддержка через документацию и сервис заявок. Условия обслуживания фиксируются в договоре (SLA). |
| 4.17 | Требования к функциям, выполняемым Платформой | Платформа должна: захватывать пакеты с заданного интерфейса; агрегировать их в сетевые потоки; извлекать набор признаков (статистических и поведенческих) для каждого потока; классифицировать трафик с помощью ML-моделей (например, Random Forest, нейронная сеть); выявлять VPN-каналы и аномалии; генерировать отчёты (графики, CSV-файлы) и логи. Все функции доступны через API и интерфейсы системы. |
| 4.18 | Требования к точности проведения замеров | Точность классификации – не ниже 90% на контрольных наборах (чувствительность/полнота ≥85%). Верификация моделей проводится на общедоступных датасетах (например, CICIDS, ISCXVPN). Временные метки пакетов фиксируются с дискретностью 1 мс. Погрешность измерения скорости трафика – не более 1%. При тестировании результаты сравниваются с эталонными данными. |
| 4.19 | Требования к техническому обеспечению | ПО: Python 3.8+ (interpreter CPython), библиотеки Scapy, NFStream, pandas, NumPy, scikit-learn, PyTorch и др. Железо: x86 или ARM устройства с root-доступом, CPU 4+ ядра, оперативная память ≥8 GB, SSD-диск. ОС: Linux (рекомендуется Ubuntu 22.04 LTS) или Windows Server 2019+. Предусмотрены права администратора для захвата трафика (promiscuous mode). |
| 4.20 | Требования к электропитанию | Обычное электропитание 220 В/50 Гц. Для постоянной работы рекомендуется подключение к ИБП (исключает потерю данных при отключении питания). При развёртывании в мобильных условиях допускается питание от аккумуляторов: важно обеспечить питание минимум на несколько часов работы. |
| 4.21 | Требования к метрологическому обеспечению | Все измерения выполняются в единой системе СИ согласно ГОСТ Р 8.596-2002. Время фиксируется по синхронизированным часам (NTP). Модели тестируются на наборах с известными метками, результаты анализируются по стандартам статистики (ROC, AUC). Соответствие метрик (точность, полнота) проверяется в рамках валидации. |
| 4.22 | Требования к информационному обеспечению | Информационный ресурс – централизованная база данных потоков и статистики (например, ClickHouse). Все собранные данные хранится в структурированном виде. Реализован механизм архивации результатов в CSV/Parquet. Генерируются регулярные бэкапы отчётов. Предусмотрен интерфейс экспорта/импорта данных для внешних аналитических систем. |
| 4.23 | Требования к программному обеспечению | Программное обеспечение – лицензированное Python и свободные библиотеки. Все зависимости перечислены в requirements.txt. Поставляются установочные пакеты или контейнер Docker. ПО соответствует техническим регламентам и сопровождается исходным кодом. Документация по ПО (архитектура, API) включена в комплект поставки. |
| 4.24 | Состав и содержание работ по созданию Платформы | Работы включают: анализ требований и сбор данных, архитектурное проектирование, разработку модулей кода, создание тестовых сценариев, интеграция компонентов, тестирование (юнит, интеграция, нагрузочное), подготовку документации (Руководства пользователя/администратора). При каждом этапе – отчёты и ревью заказчиком, корректировка на основе замечаний. |
| 4.25 | Этапы создания платформы | 1) Анализ требований, обзор аналогов и подходов. 2) Проектирование системы (структуры модулей, протоколов взаимодействия). 3) Реализация прототипа и базовых функций (захват, анализ, классификация). 4) Обучение ML-моделей на собранных данных. 5) Полная реализация функционала, интеграция с БД и интерфейсами. 6) Тестирование и оптимизация. 7) Экспертиза (внешний аудит кодовой части). 8) Пилотное внедрение и опытная эксплуатация. |
| 4.26 | Порядок контроля и приемки Платформы | После завершения разработки проводятся приёмо-сдаточные испытания: проверка соответствия заявленным требованиям, функциональное и нагрузочное тестирование по утверждённым сценариям. Заказчиком оформляется акт приёма/сдачи. Затем следует опытная эксплуатация (pilot) – тестирование системы в реальных условиях и коррекция по результатам. |
| 4.27 | Требования к составу и содержанию работ по подготовке объекта к вводу в действие | Необходимо подготовить сеть: настроить зеркалирование портов (SPAN) или TAP на коммутаторах, обеспечить доступ приложения к нужным интерфейсам. На узлах установить ОС и Python с нужными правами. Обновить прошивки устройств. Проверить соединения с внешними базами данных и сервисами. |
| 4.28 | Требования к документированию | Документация разрабатывается по ГОСТ 34.201-89 и ГОСТ 19.101-77: ТЗ (данный документ), технический проект, руководства пользователя/администратора, отчёт об испытаниях. Всё на русском языке, с полным описанием функций, ограничений, процедур установки и обновления. Форматы и шаблоны документов соответствуют нормативам и корпоративному стандарту заказчика. |