Анализ сетевого трафика в реальном времени

Шумилов Н.Н.

инженер-программист АО «ОНИИП»

Актуальность работы

Оперативный мониторинг — ключ к безопасности и эффективности сетей.

Современные сети требуются быстрой видимости и реактивности.

Пассивные подходы минимизируют риск воздействия на рабочие системы.

Цель

Создание автономного программно-аппаратного решения для непрерывного мониторинга трафика в реальном времени.

Обеспечить прозрачный контроль без вмешательства в работу сети и без влияния на производительность каналов.

Анализ сетевого трафика

Это комплексный процесс сбора, корреляции и глубинной обработки передаваемых пакетов данных.

Мониторинг является незаменимым инструментом для диагностики «узких мест», оптимизации использования ресурсов и улучшения качества обслуживания (QoS).

5 Виды мониторинга

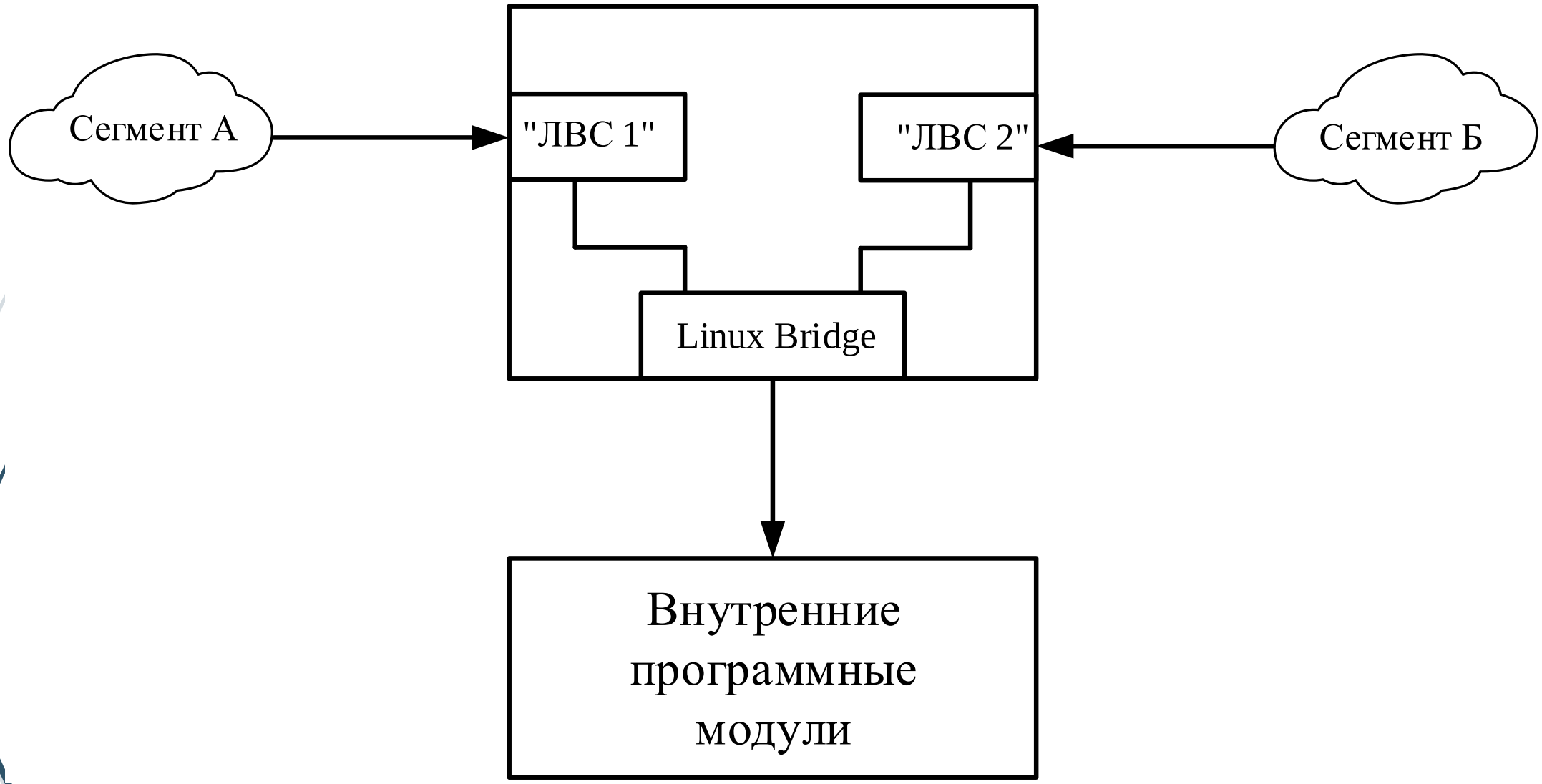
Характеристика	Пассивный мониторинг	Активный мониторинг
Методология	Анализ реального трафика	Генерация и отправка тестового трафика
Влияние на сеть	Отсутствует	Может создавать нагрузку и задержки
Сценарии использования	Непрерывный мониторинг, сбор долгосрочной статистики	Целевые проверки, аудит, проверка доступности
Недостатки	Не видит «бесшумные» устройства	Ресурсоемкий, может нарушать работу чувствительных систем
Данные	Реальные пользовательские данные и трафик	Синтетические пакеты и ответы от устройств
Источники	Зеркалирование портов, сетевой TAP	SNMP, WMI, ping, telnet, тестовые пакеты

Методы захвата трафика

- Port Mirroring (SPAN) — копирование трафика коммутатором.
- Network TAP — физический разрыв канала для 100% видимости.

Реализуемое устройство функционирует как мост между сегментами сети.

7 Аппаратная архитектура



8 Программная архитектура

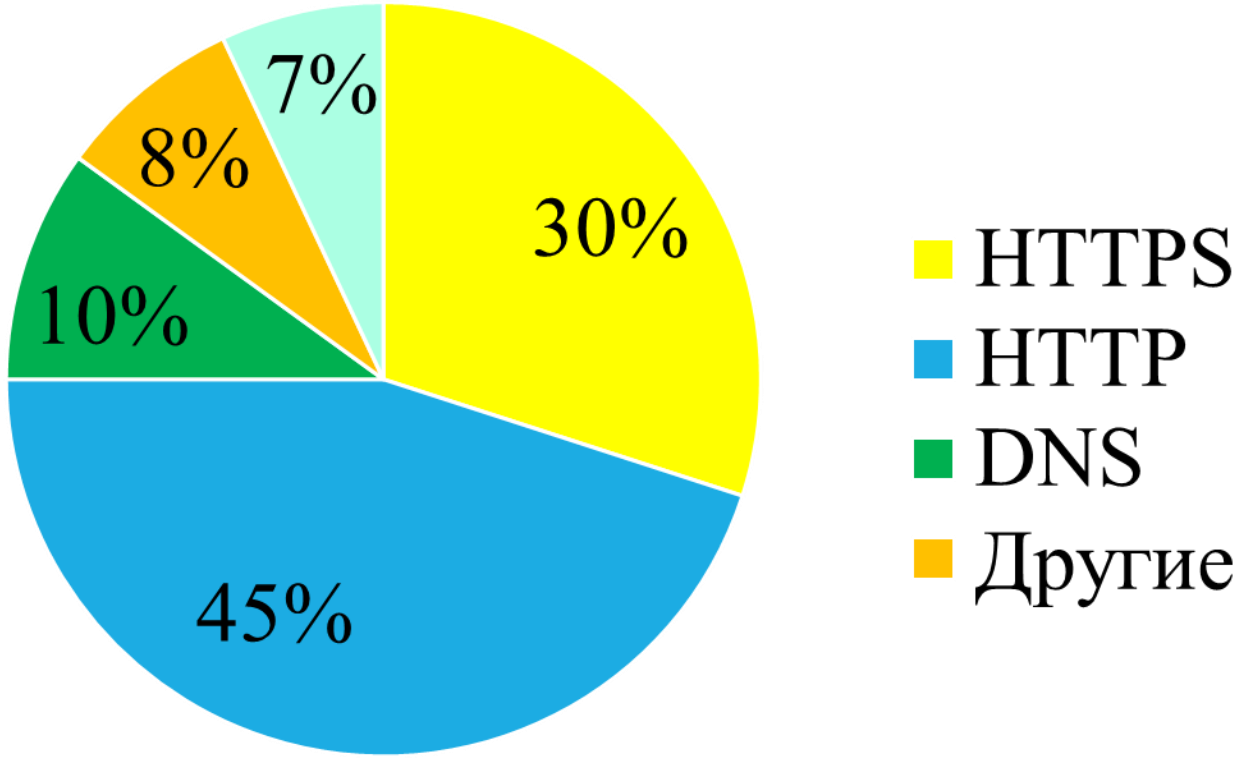
- Язык: C++ (многопоточное приложение).
- Захват пакетов: libpcap
- Потоки: захват/парсинг пакетов и измерение пропускной способности

DPI и классификация трафика

- Глубокое инспектирование для декодирования на всех уровнях OSI.
- Извлекаются: MAC, IP, порты, протоколы, метаданные приложений.
- Результат: потоковые записи, статистика

Выходные интерфейсы и интеграция

- HTTP-сервер, отдающий статистику в формате JSON.



Практические сценарии применения

- Выявление узких мест и оптимизация пропускной способности.
- Обнаружение подозрительных паттернов (сканирование, DoS, утечки данных).
- Долгосрочное планирование модернизации сети на основе реальных данных.

Результаты

- Разработан прототип программно-аппаратного комплекса в режиме моста.
- Реализован захват трафика, классификация потоков и сбор статистики в реальном времени.
- Предоставлен API (HTTP/JSON) для интеграции и визуализации.

Выводы

- Пассивный мониторинг в режиме моста обеспечивает прозрачный и надёжный сбор данных
- Использование libpcap и Linux даёт гибкость и переносимость
- Использование отечественной аппаратной базы повышает независимость

Спасибо за внимание