02.02.03 – Фундаментальная информатика и информационные технологии

Выполнил:
студент 4 курса Э. А. Ковтун

Научный руководитель:
кандидат физико-математических наук, доцент кафедры информатики и вычислительного эксперимента К. Ю. Гуфан

Система инспекции сетевого SSL/TLS трафика и имитации плохих каналов передачи данных для тестирования межсетевого взаимодействия приложений по протоколу TCP

Институт ММиКН им И.И. Воровича, Южный Федеральный Университет

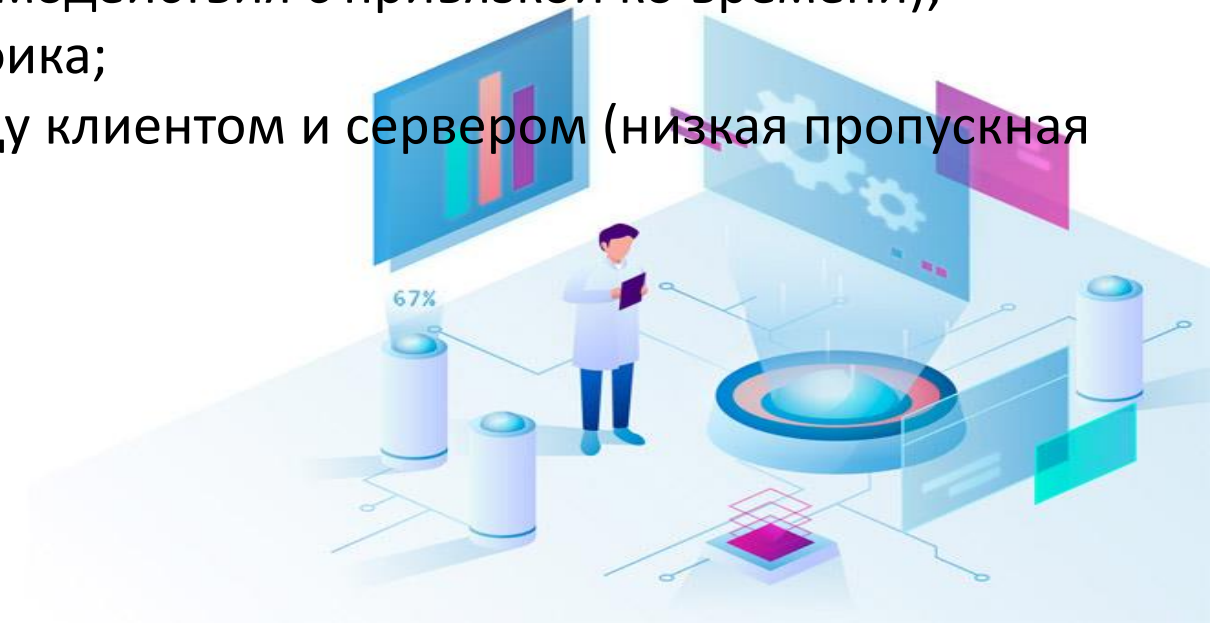
1

Постановка задачи

1. Исследовать к подходы к тестированию сетевого взаимодействия вычислительной системы, построенной на базе клиент-серверной архитектуры

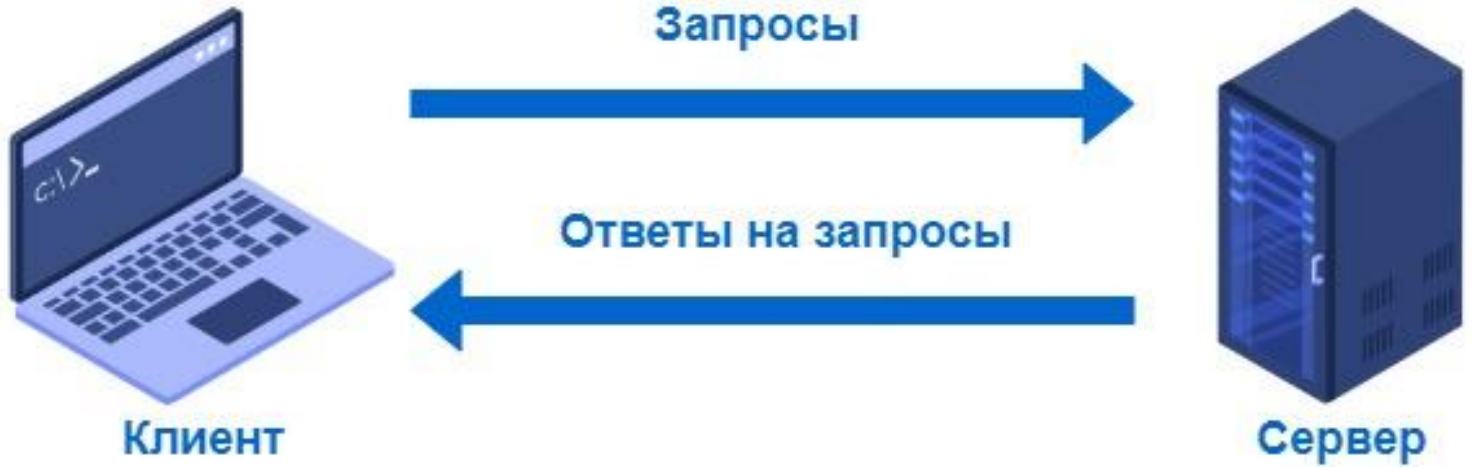
2. Разработать программное обеспечение, позволяющее:

- журналировать сеансы взаимодействия выбранных пар клиентов и серверов (с возможностью ведения протокола их взаимодействия с привязкой ко времени);
- выполнять инспекцию зашифрованного трафика;
- имитировать проблемы в канале связи между клиентом и сервером (низкая пропускная способность, задержки и т.д.).



2

Клиент-серверная архитектура



Клиент (заказчик услуг) – программное обеспечение, инициирующее сетевое взаимодействие. На основании пользовательских команд запрашивает данные у сервера или просит его произвести необходимые вычисления на основе отправляемых ему данных и вернуть результат вычисления.

Сервер (поставщик услуг) – программное обеспечение, принимающее набор команд, на основание правил собственного API, и производящее их исполнение.

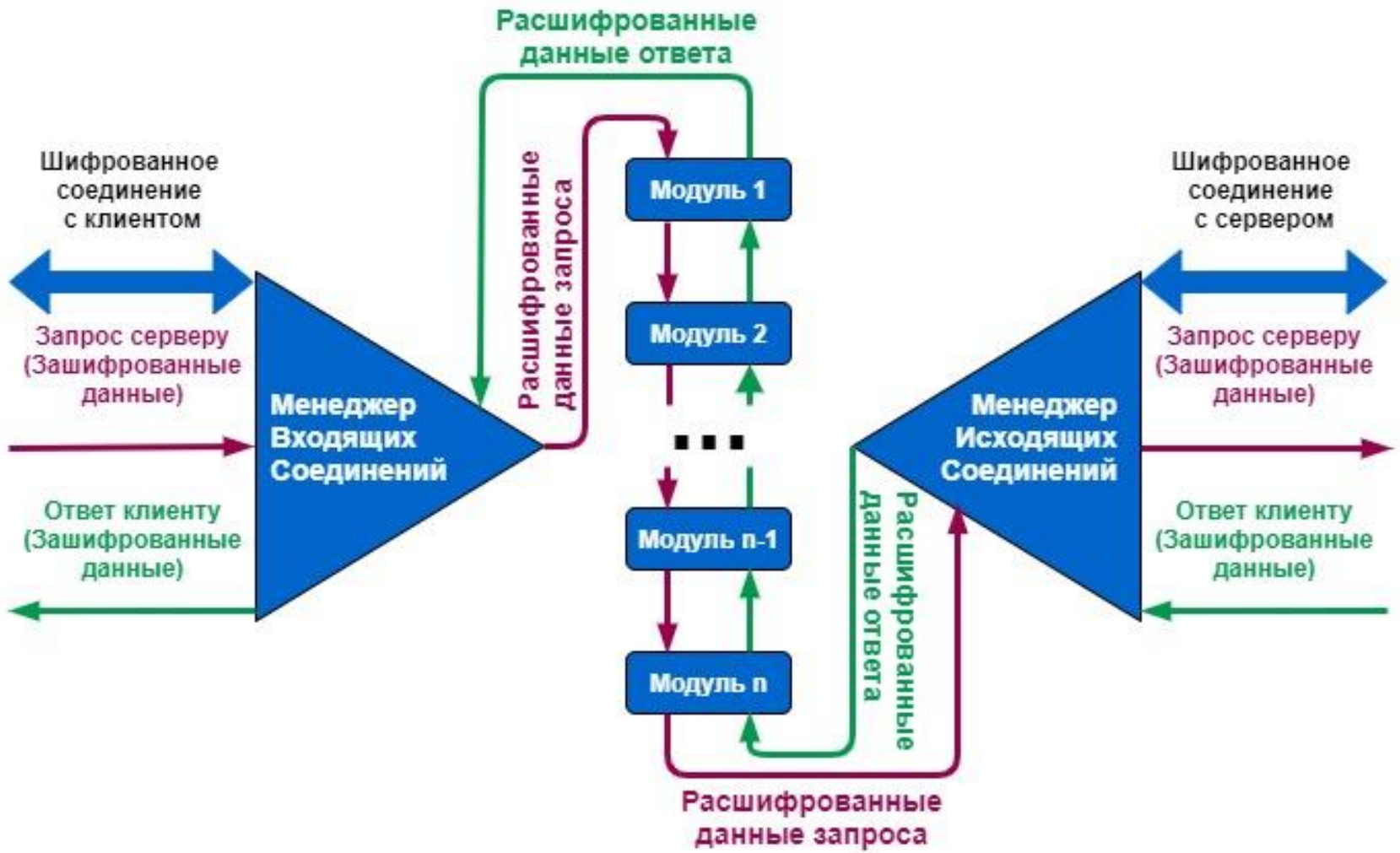
3

Промежуточный прокси-сервер

Промежуточный сервер (прокси-сервер) выполняет роль посредника, позволяя клиентам осуществлять косвенные запросы (принимать и передавать их через прокси-сервер) к целевому серверу и принимать от него ответы.

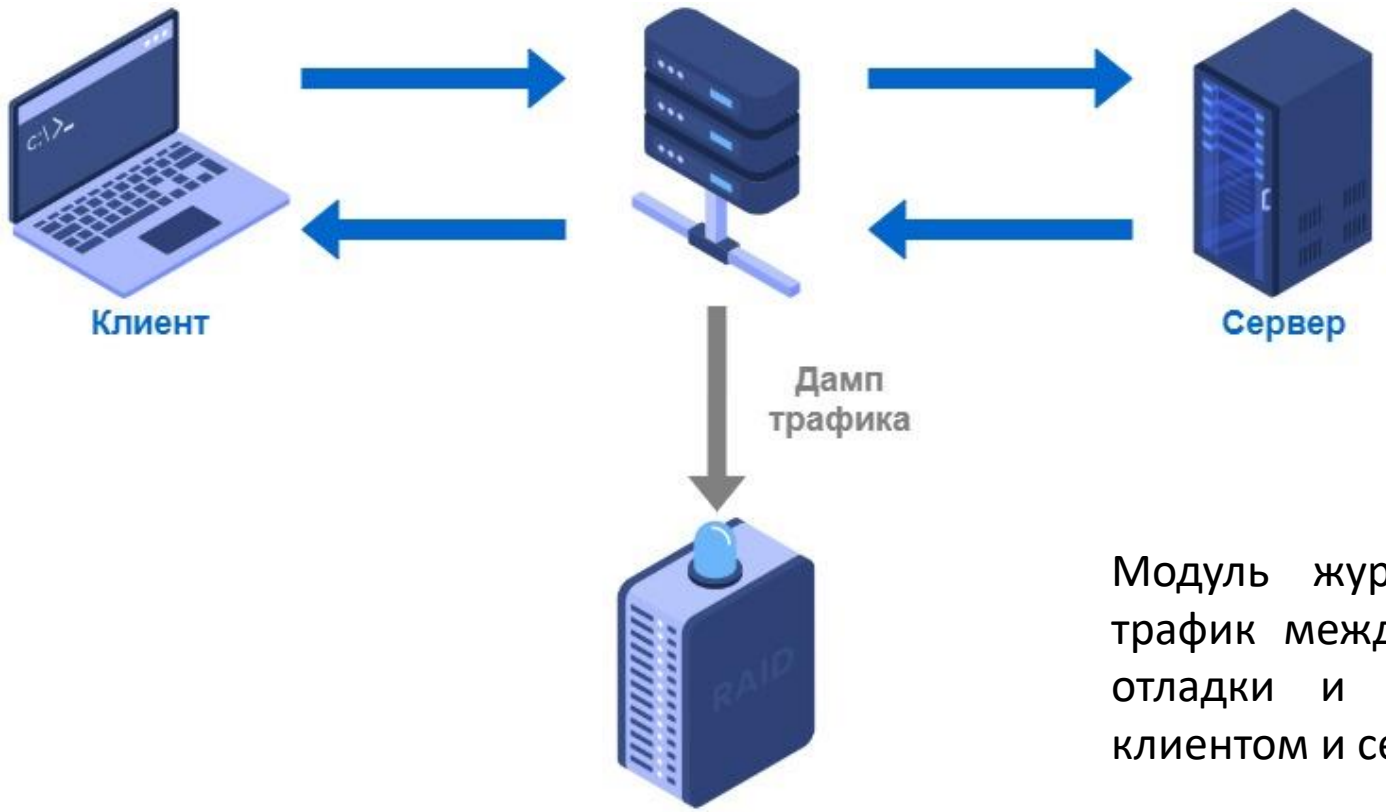
SOCKS – сетевой протокол, позволяющий прозрачно обмениваться данными между клиентом и сервером через SOCKS-прокси-сервер. Прокси-серверы, взаимодействующие с клиентами по протоколу SOCKS, обычно производят передачу данных (в обоих направлениях) без их анализа или модификации.

Разработка архитектура прокси-сервера



5

Журналирование сетевого трафика



Модуль журналирования сохраняет весь проходящий трафик между клиентом и целевым сервером с целью отладки и анализа проблем взаимодействия между клиентом и сервером.

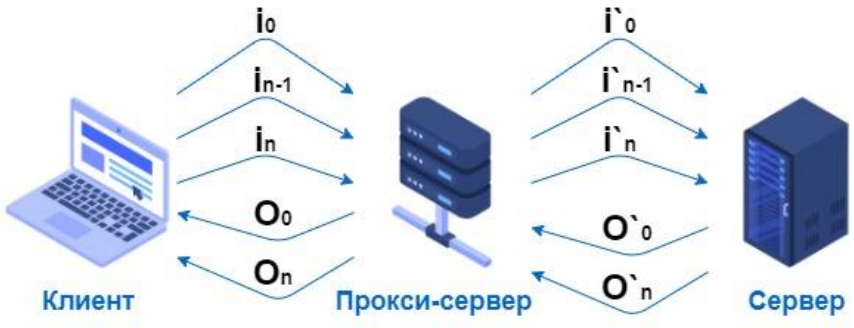
6

Инспекция сетевого SSL/TLS трафика

Стандартный режим работы SOCKS прокси-сервера при организации подключения TCP-сессий с шифрованием трафика по протоколам SSL/TLS.
Данные не анализируются и не изменяются.

Специальный режим работы SOCKS-прокси-сервера с инспекцией SSL/TLS трафика.
Данные расшифровываются для анализа.

Имитация низкой пропускной способности сети



Используется использует алгоритм "протекающего ведра" для ограничения суммарной полосы пропускания в каждом направлении, равномерного её распределения между всеми активными TCP-сессиями.

Время с начала отсчета, мс	В буфере / отправлено	Номер соединения / трафик, кбайт		
		№ 1, данные	№2, данные	№3, данные
0	в буфере	100	150	200
	отправлено	20	21	21
100	в буфере	80	129	179
	отправлено	21	21	22
200	в буфере	59	108	157
	отправлено	21	22	22
300	в буфере	38	86	135
	отправлено	20	20	20
400	в буфере	18	66	115
	отправлено	18	23	23
500	в буфере	0	43	92
	отправлено	0	30	31
600	в буфере	0	13	61
	отправлено	0	13	49
700	в буфере	0	0	12
	отправлено	0	0	12
800	в буфере	0	0	0
	отправлено	0	0	0

Результаты работы

1. Выполнено исследование подходов к обеспечению тестирования и отладки сетевого взаимодействия в программно-аппаратных системах, построенных на принципах клиент-серверной архитектуры;
2. Изучены детали реализации протокола SOCKS версии 5 и принципы работы прокси-серверов, функционирующих по протоколу SOCKS;
3. Изучены принципы работы протоколов шифрования сетевого трафика SSL/TLS, широко применяемых в сети Интернет;
4. Изучены методы обеспечения «разрыва» канала передачи данных, создаваемого протоколами SSL/TLS;
5. Разработана модульная архитектура прокси-сервера;
6. Изучены базовые и сетевые возможности фреймворка Qt и принципы асинхронного сетевого взаимодействия;
7. Разработано программное средство – SOCKS-прокси-сервер, предназначенное для тестирования и отладки сетевого взаимодействия клиент-серверных приложений;
8. Проведено тестирование разработанного программного средства в реальной сети.

Благодаря использования асинхронных подходов для организации сетевого взаимодействия прокси-сервер обеспечивает:

- высокую сетевую производительность даже при работе в однопоточном режиме;
- возможность одновременной работы с множеством (до 400-600) сетевых соединений с включенной инспекций и управлением скоростью передачи.

Система инспекции сетевого SSL/TLS трафика и имитации плохих каналов передачи данных для тестирования межсетевого взаимодействия приложений по протоколу TCP

github.com/EKovtun/LittleSocks