Технологично училище Електронни системи към
Технически университет - София

ДИПЛОМНА РАБОТА

Тема: Мрежов анализатор с възможност за отдалечен анализ посредством AngularJS 2 клиент

Дипломант: Ивайло Арнаудов

Научен ръководител: Стоил Стоилов

София, 2017

Списък на съкращения

VPN	Virtual Private Network
IP	Internet Protocol

Съдържание

0.1 Компютърни мрежи 4
0.2 Приложения на компютърните мрежи 4
0.2.1 Приложения на компютърните мрежи в бизнеса 4
0.2.2 Приложения на компютърните мрежи в дома 5
0.3 Изисквания към мрежов анализатор 5

1 Методи и технологии за реализация на мрежов анализатор 7
1.1 Основни принципи и технологии за реализиране на мрежови анализатор . . 7
1.1.1 Софтуерни характеристики на компютърните мрежи 7
1.2 Съществуващи решения и реализации 10

2 Проектиране на структурата на мрежов анализатор 11
2.1 Функционални изисквания към мрежов анализатор 11
2.2 Съображения за избор на програмни средства и развойна среда 11
2.3 Проектиране на структурата на базата от данни 11

3 Програмна реализация на мрежов анализатор 12

4 Ръководство на потребителя 13

5 Заключение 14

A Изходен код 15

Библиография 16

Списък на фигурите 17

Списък на таблиците 18

Увод

0.1 Компютърни мрежи

Всеки от последните три века бива доминиран от някаква нова технология. Пример за това е ерата на механичните системи съпътстващи Индустриалната революция през XVIII век. За XIX век пък е характерен парния двигател. През XX век, ключовата технология е събирането, обработката и дистрибуцията на информация. С развитието ѝ човечеството става свидетел на инсталацията на глобални телефонни мрежи, изобретяването на радиото и телевизията, експоненциалния растеж на развитието на компютърната индустрия и, разбира се, Интернет. Като резултат от огромния технологичен прогрес в сферата на информационните технологии, през XXIв. разликите между съхраняване, транспортиране и обработка на информация изтъняват, а успоредно с това растат и изискванията на крайния потребител към комуникационните услуги.

Въпреки крехката възраст на компютърната индустрия (напр. в сравнение с автомобилната), тя прави значителен прогрес. През първите две десетилетия от съществуването им, компютърните системи са били силно централизирани. Университет или средно голяма фирма биха имали един или два компютъра, а по-големите институции - по няколко. Идеята за съществуването на малки устройства тип смартфон, които са взаимносвързани, е била по-скоро утопична.

Обединяването на компютрите и комуникациите оказва голямо влияние върху организацията на самите компютърни системи. Старият модел при който един компютър изпълнява заявките на цялата организация бива заменен от нов модел при който голямо количество отделни, но взаимносвързани компютри извършват обработка на дадена информация. Тези системи се наричат **компютърни мрежи**. Неформална дефиниция за компютърна мрежа е множество от автономни компютри, взаимносвързани (можещи да обменят информация помежду си) от една технология. Tanenbaum and Wetherall (2011)

0.2 Приложения на компютърните мрежи

0.2.1 Приложения на компютърните мрежи в бизнеса

Обикновено повечето компании имат голямо количество компютри, най-често по един за всеки служител. Изначално, те биха могли да работят в изолация един от друг, но в даден момент идва необходимост те да бъдат свързани с цел служителите да извършват работата си по-пълноценно чрез колаборация помежду си.

Един от основните проблеми, който решават компютърните мрежи е **споделянето на ресурси**. Целта е информацията да бъде достъпна от всеки в мрежата независимо от физическото му местоположение. Пример за това е група служители на организация, използващи един общ принтер — нито един от тях няма нужда от личен такъв, затова и решението е по-евтино, бързо и по-лесно за поддръжка от поддръжката на голямо количество принтери.

По-важен проблем, който решават компютърните мрежи е споделянето на

информация. Малките и средни компании са фундаментално зависими от дигиталната информация. Повечето компании имат записи за клиенти, за продукти, финансова информация и т.н. онлайн. По-малките компании са традиционно разположени в един офис, докато при по-големите интернационални компании компютрите и служителите са разпрострени в много държави. Това обикновено бива имплементирано чрез **Virtual Private Network (VPN)** с цел агрегация на индивидуалните мрежи на различни местоположения в една обща.

В допълнение, компютърните мрежи дават възможността да се използват вече изградената мрежова инфраструктура за телефонни разговори благодарение на технологията **Internet Protocol (IP) телефония**, или още известна като **Voice over IP (VoIP)**. Те също предоставят механизми за по-богати форми на виртуална комуникация – споделяне на екрана (**Desktop sharing**), видеоконференции, споделена обработка на документи, дори отдалечен мониторинг на пациенти. Компютърните мрежи отварят вратите и за нов бизнес модел, наречен електронна търговия (или **e-commerce**), който се развива с големи темпове в последните години и става де факто стандард при търговията от всякакъв тип. Tanenbaum and Wetherall (2011)

0.2.2 Приложения на компютърните мрежи в дома

В началото на компютърната индустрия причините за покупка на компютър от крайния потребител са се свеждали до нужда от обработка на текст и игри. През XXI век, най-голямата причина човек да се сдобие с персонален компютър е достъп до Интернет. Аналогично на компаниите, крайните потребители могат да достъпят отдалечена информация, да комуникират посредством **социалните мрежи**, да купуват продукти и услуги чрез e-commerce системи, да използват електронно банкиране, да споделят мултимедия и софтуер, да колаборират посредством **wiki** сайтове (напр. Wikipedia). В перспектива, използването на компютърни мрежи за подобряване на интеракцията между хората може да се окаже най-важното приложение.

Друго напоследък развиващо се приложение на мрежите е концепцията за **Internet of Things (IoT)**. Основната ѝ характеристика е че електронните устройства на крайните потребители се включват в компютърните мрежи; напр. душа в банята, който традиционно не е компютър, би могъл да записва какво количество вода е използвано и да праща информацията на приложение, което изчислява как водата да бъде използвана възможно най-ефикасно. Tanenbaum and Wetherall (2011)

0.3 Изисквания към мрежов анализатор

Анализа на пакети (**packet analysis**), още известен като **packet sniffing** или **protocol analysis**, описва процеса на заснемане и интерпретиране на данни в реално време (т.е в момента на преминаване през преносвателната среда) с цел по-задълбочено разбиране на процесите в мрежата. Анализа на пакети типично се изпълнява от програма, наречена мрежов анализатор или пакетен анализатор (**packet sniffer**). Sanders (2011) Изискванията към един такъв мрежов анализататор е да може да помогне на мрежовия администратор със следните задачи:

- Идентифициране на участниците в мрежата
- Идентифициране на кой или какво използва (bandwidth на български?)
- Идентифициране на моментите на максимално използване (load) на мрежата
- Идентифициране на потенциални атаки или злонамерена активност

Глава 1

Методи и технологии за реализация на мрежов анализатор

1.1 Основни принципи и технологии за реализиране на мрежови анализатор

Процеса на анализ на пакети включва кооперация между софтуера и хардуера и може да бъде разделен в следните три стъпки:

- **Събиране** В началната фаза на работата си, анализатора събира 'сурови', неинтерпретирани данни в двоичен вид директно от проводника. Типично, това става като съответно избрания мрежови интерфейс за анализ бива превключен в т.нар. **promiscuous mode**. В този режим, мрежовата карта може да 'слуша' за всевъзможен тип трафик по дадения мрежови сегмент, а не просто такъв, адресиран до станцията.

- **Конвертиране** В следващата фаза на работата си, анализатора конвертира събраните данни в разбираем формат за крайния потребител. Тук е мястото, където повечето добри анализатори спират с анализа. След тази стъпка, данните събрани от преносвателната среда са във вид, който може да бъде интерпретиран на много основно ниво; останалата по-голяма част от анализа се оставя на крайния потребител.

- **Анализ** В третата и финална фаза, мрежовият анализатор извършва реалния анализ на събраната и конвертирана информация. Анализатора взима събраните данни, отчита използвания мрежови протокол базирайки се на извлечените до момента данни и започва да анализира конкретните свойства на протокола.

С цел да бъде разбран процеса на работа, а и съответно модела на реализация на мрежов анализатор, е необходимо дефиниране на основните принципи на комуникация между компютърните системи.

1.1.1 Софтуерни характеристики на компютърните мрежи

Дизайнерите на първите компютърни мрежи са били строго концентрирани върху хардуерната имплементация, а софтуера е бил второстепенен проблем. В модерните компютърни мрежи този подход е грешен и не работи. Софтуерът в модерните компютърните мрежи е силно структуриран и е основополагащ за бързото им развитие в последните години.

Протоколни йерархии

За да се намали комплексността на решенията, повечето мрежи са организирани като стек от **слоеве** или **нива**, всеки изграден върху слоя под него. Броя на слоевете, имената на всеки от тях, съдържанието на всеки и функциите, които изпълнява са различни за различните мрежи. Целта на всеки слой е да предостави конкретни услуги на разположените по-високо от него в йерархията слоеве като им спестява детайлите около имплементацията на тези услуги.

Тази концепция е широко популярна в компютърните науки, още известна е като криене на имплементационни детайли, абстрактни типове от данни, енкапсулация на данни и обектно-ориентирано програмиране. Фундаменталната идея е че дадена част от софтуера (или хардуера) осигурява услуга на потребителите си, но скрива от тях детайлите на вътрешното си състояние и алгоритмите.

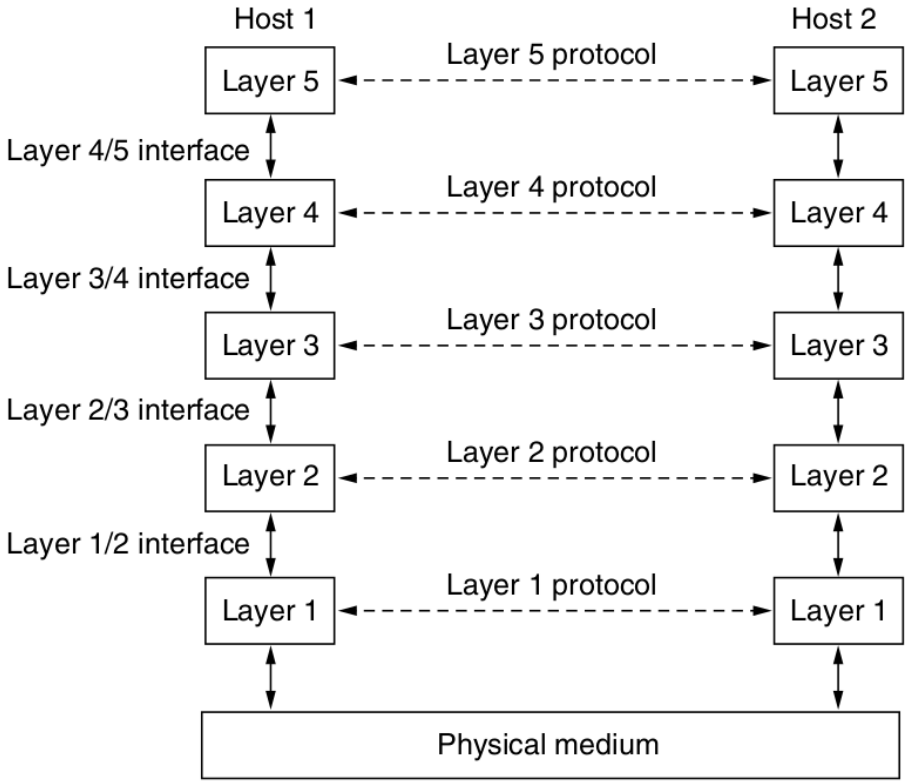
Когато слоят n на една машина е във връзка със слоят n на друга машина, правилата и конвенциите използвани в тази връзка се наричат **протокол на n-ти слой**. На практика, протокол е договореност между комуникиращите страни относно това как протичат процесите на комуникацията между тях. Tanenbaum and Wetherall (2011)

Популярна аналогия на протоколите са човешките езици — всеки един има конкретни правила за структура на изречението, за форми на глаголи, прилагателни и т.н. Както при езиците, с протоколите може да се дефинира маршрутизирането на пакети, иницииране на връзка, потвърждаването на приети данни и др. Протоколите могат да бъдат прости или комплексни. Някои от общите свойства, които традиционно споделят, макар и абстрактно представени тук, са:

- **Инициация на връзка** Дефинира кой инициира връзката, напр. клиентът или сървърът. При иницииране на връзка може да е необходима и допълнителна служебна информация преди да протече обмена на полезна информация.
- **Договаряне на параметрите на връзката** Дефинира процес, в който двете страни се разбират — дали връзката е криптирана, как се пренасят ключовете за декриптиране, какъв тип е връзката (full/half duplex) и др.
- **Форматиране на данните** Дефинира подредбата на данните, в каква последователност се обработват от приемащата страна и др.
- **Откриване на грешки и корекция на грешки** Дефинира какво се случва при загуба на данни, как едната страна на връзката реагира при загуба на отгвор от другата и др.
- **Терминиране на връзка** Дефинира как дадено крайно устройство сигнализира на друго че връзката е приключила, каква финална информация трябва да бъде предадена преди успешния край на връзката и др.

Традиционно, тези протоколи не 'живеят' сами, а са основополагащи за цялостния процес на комуникация. Например, на Figure 1.1 е представен пет слоен модел на комуникация между два софтуерни процеса. Реално данни не се предават директно от слой n на едната машина до слой n на другата: комуникацията е **виртуална** (означена с прекъснати линии на Figure 1.1). Вместо това, всеки слой предава данни и контролна информация на този под него докато не се достигне най-ниския слой. Под първия слой е физическият, т.е преносвателната среда през която реалната комуникация се случва. (означено с непрекъснати линии на Figure 1.1)

Между всяка съседна двойка слоеве има **интерфейс**. Този интерфейс дефинира какви операции и услуги долният слой предлага на горния. Интерфейсите между слоевете трябва



Фигура 1.1: Модел на пет слойна мрежа

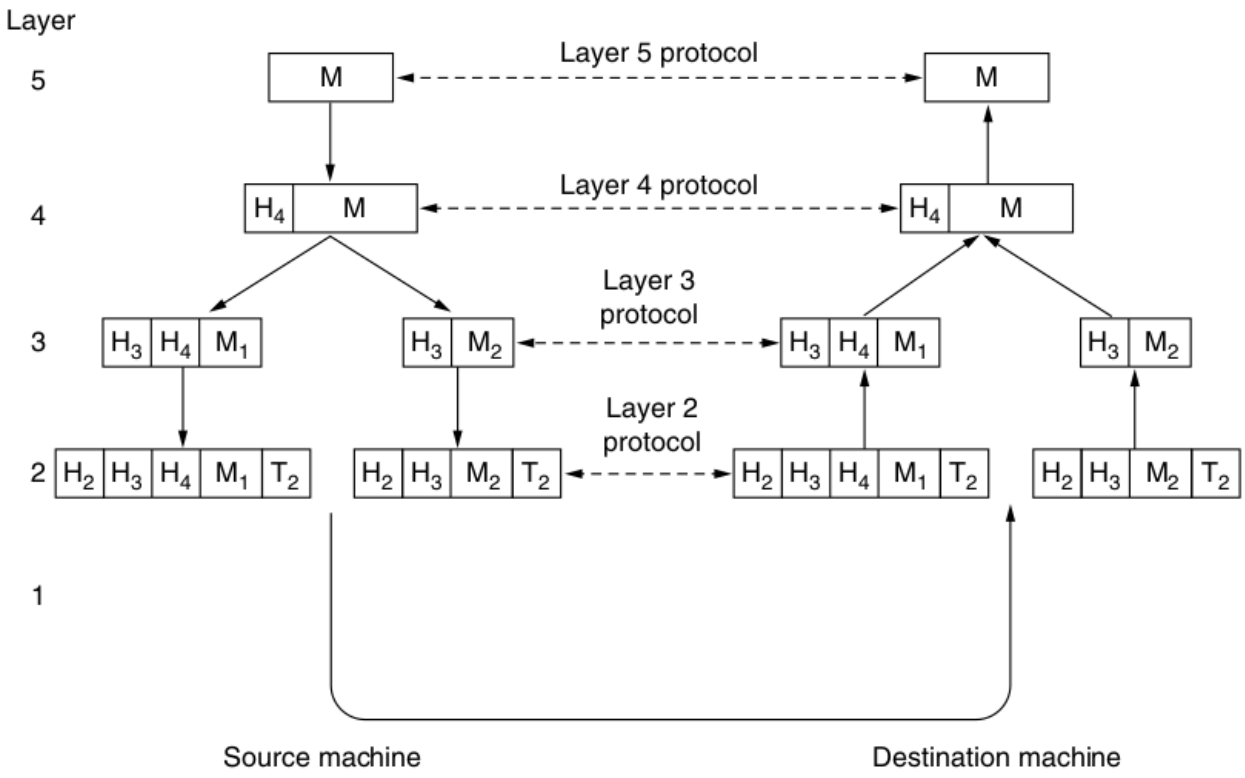
да бъдат ясно дефинирани. Това впоследствие би улеснило замяната на един слой с напълно различен протокол или имплементация (напр. смяна на телефонни линии със сателитни такива), защото единственото, което се очаква от новия протокол, е да предлага *точно* същото множество услуги на горния слой като стария. Аналогично, този механизъм позволява един протокол да се промени в даден слой без знанието на слоевете под и над него.

Основен механизъм на междуслойна комуникация. Капсулиране и декапсулиране.

Основния механизъм на междуслойна комуникация може да бъде описан с помощта на пет слойния модел представен във Figure 1.1. Типично, даден процес (т.е приложение) иска да изпрати съобщението M. От петия слой, съобщението бива предадено на четвъртия слой за трансмисия. Четвъртият слой слага т.нар. **header** в началото на съобщението и предава резултата към трети слой. Този header включва служебна информация, напр. адреси, за да може съответния четвърти слой на приемната машина да достави съобщението. Други примери за служебна информация могат да бъдат числови поредици (**sequence numbers**), често използвани когато слоят на по-ниско ниво няма функционалност за запазване на последователността на съобщенията, размери и времена.

В много мрежи, често няма граница относно размера на съобщения на четвърти слой, но почти винаги има такава относно размера от самият протокол на трети слой. Следователно, третият слой трябва да раздели идващите съобщения на по-малки единици — пакети, като успоредно с това добавя header към всеки пакет. В случая на Figure 1.2, съобщението M се разделя на две части: M_1 и M_2.

Третият слой аналогично предава пакетите на втория слой, който от своя страна освен че добавя header, добавя и **опашка** (**trailer**). Резултата се предава на първия слой, който се занимава с физическия пренос на данните. Този процес е още известен като **капсулация** (**encapsulation**). В приемащата страна, съобщението се декапсулира (**decapsulation**)



Фигура 1.2: Междуслойна комуникация. Капсулиране и декапсулиране.

като всеки header се отделя успоредно с "изкачването"на съобщението нагоре по слоевете. Нито един header за слоевете под n-тия не достига до n-ти слой. Едновременно с това, на Figure 1.2 ясно проличава виртуалната и реална комуникация, както и разликите между протоколи и интерфейси. Например, на четвърти слой процесите концептуално интерпретират комуникацията си като хоризонтална използвайки протокола на четвърти слой и биха имали функции от типа на send() и recieve(), въпреки че в реалност те комуникират със по-ниските слоеве през 3/4 интерфейса, а не директно с другата страна.

1.2 Съществуващи решения и реализации

Глава 2

Проектиране на структурата на мрежов анализатор

block diagrams?

2.1 Функционални изисквания към мрежов анализатор

what the analyzer should support

2.2 Съображения за избор на програмни средства и развойна среда

why c++ rocks, why angular rocks, show websocketpp benchmarks

2.3 Проектиране на структурата на базата от данни

??
explain orms / explain odb (if i integrate it) / show the db structure

Глава 3

Програмна реализация на мрежов анализатор

show sum code eh

Глава 4

Ръководство на потребителя

explaining how a sniffer works to noobs

Глава 5

Заключение

Приложение А

Изходен код

Библиография

C. Sanders. *Practical packet analysis: using Wireshark to solve real-world network problems*. No Starch Press, San Francisco, Calif, 2. ed edition, 2011. ISBN 978-1-59327-266-1. OCLC: 755869776.

A. S. Tanenbaum and D. Wetherall. *Computer networks*. Pearson Prentice Hall, Boston, 5th ed edition, 2011. ISBN 978-0-13-212695-3. OCLC: ocn660087726.

Списък на фигурите

1.1 Модел на пет слойна мрежа 9
1.2 Междуслойна комуникация. Капсулиране и декапсулиране. 10

Списък на таблиците