**ТЕХНОЛОГИЧНО УЧИЛИЩЕ “ЕЛЕКТРОННИ СИСТЕМИ”**

**към ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ**

**ДИПЛОМНА РАБОТА**

Тема: Изграждане на MPLSVPN мрежова архитектура

Дипломант: Научен ръководител:

*Александър Павлов Христо Войнски*

СОФИЯ

2020

**Използвани съкращения**

* IP – Internet Protocol
* OSI – Open Systems Interconnection
* LAN – Local Area Network
* WAN – Wide Area Network
* MAC – Media Access Control
* BGP – Border Gateway Protocol
* eBGP – Exterior Border Gateway Protocol
* iBGP – Interior Border Gateway Protocol
* MPLS – Multiprotocol Label Switching
* LDP – Label Distribution Protocol
* RSVP - Resource Reservation Protocol
* VRF – Virtual Routing & Forwarding
* IS-IS – Intermediate System to Intermediate System
* QoS – Quality of Service
* TCP – Transmission Control Protocol
* IPv4 – Internet Protocol version 4
* IGP – Interior Gateway Protocol
* EGP – Exterior Gateway Protocol
* IGRP – Interior Gateway Routing Protocol
* EIGRP – Enhanced Interior Gateway Routing Protocol
* OSPF – Open Shortest Path First
* NET – Network Entity Title
* RIP – Routing Information Protocol
* RIPng - Routing Information Protocol next generation
* MAC – Media Access Control
* PDU – Protocol Data Unit
* LSPs – Link State PDUs
* IIHs – IS-IS Hello PDUs
* CSNPs - Complete Sequence Number PDUs
* PSNPs - Partial sequence number PDUs
* VPN – Virtual Private Network
* RFC – Request For Comment
* AD – Administrative Distance
* AS – Autonomous System
* UDP – User Datagram Protocol
* MED – Multiple Exit Discriminator
* TTL – Time-To-Live
* NHLFE - Next Hop Label Forwarding Entry
* ILM - Incoming Label Map
* DIS – Designated Intermediate System

**Увод**

В днешно време компютърните мрежи са изключително необходими и с всеки изминал ден мрежовата свързаност нараства. Хората не биха могли да комуникират помежду си, ако не съществуваха мрежовите устройства. С нарастването на компютърните технологии нараства и нуждата за сигурност, ефикасност и качество на интернет услугите. Големите компании и организации, зависими от Интернет свързаността, тези които поддържат и изпълняват изискванията от своите клиенти, се нуждаят от множеството технологии и протоколи, които предоставят ефикасност, качество, сигурност и съвременно обменят информация помежду си.

За да могат доставчиците да предоставят на клиентите си своите услуги най-ефикасно и най-качествено, се използват множество техологии и методи. Често използвана технология при корпоративните решения е Multi-Protocol Label Switching (MPLS). Една от най-често срещаните услуги на доставчиците, са виртуални частни мрежи (VPN) от втори и трети слой на OSI модела. Чрез тях се постига бързодействие, надеждност и улеснение при поддръжката на частните мрежи на клиентите, като се предоставя и възможност за пренасяне на различен трафик, спрямо изискванията на компанията.

Главната цел на дипломната работа е да се изгради мрежова архитектура, която съдържа мрежата на доставчик на интернет услуги и два негови клиента, които са представени като две копмании, всяка от които има централен офис и клон офис на различни географски разположения. Доставчикът предоставя на всеки от клиентите отделна виртуална частна мрежа, чрез която се свързват двата офиса на двете отделни компании, като едната компания няма достъп до мрежата на другата. Създава се корпоративно решение, използвайки едни от най-използваните и ефикасни технологии и протоколи – BGP, IS-IS, MPLS и се имплементират политики за контрол на качеството на услуги. Изграждането на мрежовата архитектруа е на базата на мрежовия симулатор – GNS3, а тестването и работоспособността на мрежата се осъществява чрез командите *ping, traceroute* и анализатор на трафик *Wireshark.*

**Първа глава: Технологии, протоколи и стандарти, използвани в MPLS VPN мрежовата топология**

* 1. **Интернет протокол версия 4 – IPv4**

Интернет протокол версия 4 (IPv4) е четвъртата ревизия на Интернет протокола (IP) и е широко използван протокол при комуницкациите в различните типове мрежи. Протоколът служи за индентификацията на всяко едно устройство свързано в мрежата. Той работи на Мрежовия слой на OSI модела и освен че се използва при статична маршрутизация, се използва и от протоколите за динамична маршрутизация като OSPF, EIGRP, RIP, IS-IS за избиране на най-оптималния път през който интернет пакета да премине за да достигне своята дестинация. Без IP адреси, устройствата не биха могли да комуникират помежду си и да изпращат данни един на друг. Протоколът е същественото в инфраструктурата на мрежите в целия свят.

Когато устройство се свърже към дадена мрежа, на него му се възлага уникален логически адрес, чрез който може да бъде достъпено от другите устройства в мрежата и чрез който може да комуникира с тях. Логическият адрес се дели на две части: хост част и мрежова част. Мрежовата част посочва мрежовия сегмент към който спада хоста, а хост частта - неговия адрес в тази мрежа. Адресацията се предоставя като 32 битов адрес, който е разделен на четири октети, чрез точки. Всеки октет е формиран от осем бита. За различаване на мрежовата част и хост частта на логически адрес се използва мрежова маска (subnet mask). Тя е 32 битова и е съставена от „1“ и „0“. Частта на мрежовата маска, която е съставена изцяло от „1“ определя мрежовата част, а частта съставена изцяло от „0“ – хост частта.

IP адресната класация се разделя на: Class A, B, C, D, E. Всеки клас има различен брой на възможните хостове и подмрежи и различна мрежова маска.

Разпределението, предназначението и детайлите на всеки клас са показани на Фиг.1.1.

*Фиг.1.1. Класове адреси*

От класове A, B, C освен публични адреси има и частни адреси, които са със собствени мрежови маски. Адресните пространства, резервирани за частна адресация са следните: 10.х.х.х/8 (Class А), 172.16.х.х/16 (Class B), 192.168.х.х/24 (Class C) [2]. За да се използват адресните пространства най-ефикасно и да няма излишък от неизползвани адреси, поради голямата мрежова маска, се използва метод наречен VLSM (Variable Length Subnet Mask). Чрез този метод мрежовата маска се разделя на подмаски с дължина зададена според изисквания брой хостове.

 На фиг.1.2. е показан заглавната част на IP пакета заедно с нейните полета.

*Фиг.1.2. IPv4 заглавна част (header)*

Заглавната част на IPv4 съдържа следните полета:

* **Version –** Първото поле указва коя версия на IP е пакета. Понеже само IPv4 ползва това поле, винаги стойността на това поле ще бъде 4 бита.
* **Header Length –** Това 4 битово поле указва каква е дължината на заглавната част (header) на IP пакета с 32 битови инкрементации. Минималната стойност на това поле е 5 или представено в битове „0101“, а максималната стойност е 15 или представено в битове „1111“.
* **Type of Service –** Това поле се използва за QoS (Quality of Service) или така нареченото „Качество на услугите“. Предоствени са 8 бита за това поле, чрез които може да се маркира какъв трафик пренася този пакет.
* **Total Length –** Това 16 битово поле указва целия размер на IP пакета (заглавна част и данни). Минималният размер е 20 байта, ако няма данни (Data), и максималният размер е 65,535 байта.
* **Identification –** Поле, което се използва за различаване на фрагментирани пакети от различни дейтаграми.
* **IP Flags –** Полето се състои от 3 бита, които се използват за фрагментация: Първият бит винаги е със зададена стойност „0“ ; Вторият бит се нарича DF (Don’t Fragment) бит, който показва че съответния пакет не трябва да бъде фрагментиран.
* **Fragment Offset –** В случай на фрагментирани IP пакети, това поле съдържа изместването от началото на IP пакета. Това поле се използва за сглобяване на фрагментираните IP пакети.
* **Time to Live –** Всеки път, в който IP пакет премине през маршрутизатор, стойността на това поле се намалява с 1. Когато това поле достигне стойност „0“, маршрутизаторът ще отхвърли съответния пакет и ще изпрати „ ICMP: time exceeded” съобщение към подателя, изпратил този пакет. Time to Live полето има 8 бита и се използва за предотвратяване на маршрутизиращ цикъл.
* **Protocol –** В това 8 битово поле се указва кой протокол е енкапсулиран в IP пакета, чрез стойност, например TCP (Transmission Control Protocol) има стойност 6 и UDP (User Datagram Protocol) има стойност 17.
* **Header Checksum –** В това 16 битово поле се съхранява „checksum” на заглавната част на IP пакета. Стойността се изчислява използвайки алгоритъм който преминава през всички полета на заглавната част на IP пакета. Получателят на пакета използва това поле, с цел да провери ако съществуват грешки в заглавната част.
* **Source Address –** Това поле указва 32 битов адрес на IP източника (подателя).
* **Destination Address –** Това поле указва 32 битов адрес на IP дестинацията (получателя).
* **IP Option –** Това поле не е често използвано, не е задължително и има променлива дължина според използваните опции. Когато се използва това поле, стойността на полето „Header Length“ се увеличава.
* **Data –** Това поле съдържа заглавната част и данните получени от протоколите от транспортния слой.
  1. **Статична и динамична маршрутизация**

За да може един маршрутизатор да изпраща пакети до други мрежи, той трябва да има попълнена маршрутизираща таблица, за да знае къде да изпрати съответната информация. Маршрутизиращата таблица на всеки маршрутизатор съдържа пътища до дадена дестинация, научени чрез различни начини. Освен пътищата научени като директно свързани пътища (път, който е научен от маршрутизатора чрез физическа свързаност с друго устройство), съществуват пътища, които са ръчно конфигурирани, така наречените статични пътища, и пътища които се научават с помощта на маршрутизиращи протоколи – динамични пътища.

**1.2.1. Статична маршрутизация**

Статичната маршрутизация е процес на ръчно конфигуриране, от мрежовият администратор, на пътища в маршрутизиращата таблица. Тези пътища не се променят след като се конфигурират, освен ако не се промени от мрежови администратор. Статичните пътища не се влияят от възникнали промени в мрежата, затова не изисква допълнителни ресурси за да научи промените. Ако освен статичен път има и други пътища към една дестинация, ще се предпочете статичния път, защото той е с по-голям приоритет за маршрутизатора - има AD равен 1. Може да се зададе статичен път по подразбиране (Default static route), през който минава всеки пакет, за който не е намерен път в маршрутизиращата таблица, през който да премине към своята дестинация.

**1.2.2. Динамична маршрутизация**

Динамичната маршрутизация е по-комплексен начин за маршрутизация, осъществяваща се на базата на маршрутизиращи протоколи, които променят маршрутизиращата таблица според промените настъпващи в мрежата. Преизчислението на пътищата се случва постоянно и поради това този тип маршрутизация изисква много повече ресурси отколкото статичната. Когато се появи промянa в дадена мрежа, се изпраща съобщение на маршрутизатора за да специфицира тази промяна, да си обнови и прекалкулира динамичните си пътища и да изпрати информация за променените си пътища на другите маршрутизатори в мрежата, за да попълнят промените в своята таблица. Преизчисленията на пътищата става на базата на специфичен алгоритъм, според зависи от маршрутизиращия протокол. Съответно използването на голяма част ресурси се счита за най-големия недостатък на динамичната маршрутизация.

**1.2.3. Външни и вътрешни маршрутизиращи протоколи**

* **Външни протоколи (EGP) –** Протоколи, които обменят информация и функционират между две или повече автономни системи. Този тип протоколи спада към една единствена категория – „Path-Vector“.
* **Вътрешни протоколи (IGP) –** Протоколи, които функционират в рамките на една автономна система. Те се разделят на три категории: Link-State, Distance-Vector, Hybrid.

**1.2.4. Link-State маршрутизиращи протоколи**

Този тип маршрутизиращи поддържат топологична таблица, в която се съдържа цялата информация за мрежовата топология и за всеки маршрутизатор, който използва „Link-state“ маршрутизиращ протокол. Тези маршрутизатори изпращат информация за своите директно свързани пътища и техните състояния. Тази информация се изпраща под формата на „multicast” съобщение. Когато топологичната таблица е завършена, всеки маршрутизатор изчислява най-добрите пътища за всяка мрежа. „Link-state“ протоколите са базирани на „Shortest Path First“ (SPF) алгоритъма за изчисление. Този алгоритъм е познаван също като „Dijkstra” алгоритъм. Чрез този алгоритъм, когато някое състояние на път се промени, маршрутизаторите си изпращат актуализиращо съобщение (Link-State Advertisement) помежду си и пътищата се преизчисляват отново.

Този тип протоколи са по-малко податливи на маршрутизиращи „loops“ отколкото са Distance-Vector протоколите. От друга страна, изискват повече процесорна мощ и памет. IS-IS и OSPF са част от Link-State протоколите.

**1.2.4. Distance-Vector маршрутизиращи протоколи**

Това са протоколи, които използват алгоритъм за изчисление на метрика и цена на базата на разстоянието до друга мрежа – през колко маршрутизатора ще премине един пакет или колко на брой скокове (hops) ще направи. Първоначално се изпраща „broadcast“ съобщение, което има за цел да обяви присъствието на нов маршрутизатор в мрежата. Всеки от маршрутизаторите, използващ Distance-Vector протокол, научава за нови пътища през своите съседи. Цялото действие на динамичните протоколи от този тип зависи от информацията, идваща от съседите на даден маршрутизатор. Те не знаят за състоянието на мрежата, познават само съседните маршрутизатори. Изпращат се периодични съобщения с цялата маршрутизираща таблица и времето им на сходимост е голямо. RIPv1, RIPv2, EIGRP (този протокол се смята за хибриден, защото обединява спецификациите на няколко категории протоколи), IGRP са част от протоколите от този тип.

**1.2.5. Path-Vector маршрутизиращи протоколи**

Това са протоколи, които са подобни на Distance Vector протоколите, но не се разчита на следенето на дължината на път, а по-скоро се анализира даден път (AS път). Често тези протоколи се използват за маршрутизиране в среди, където няма постоянна метрика. Устройството, през което минава пакета, се добавя към AS пътя и се праща към следващите маршрутизатори. Добавеният път се анализира, за да се провери валидността му. Border Gateway Protocol е единствения протокол, който спада към Path-Vector категорията. На Фиг.1.3 са означени категориите протоколи.

****

*Фиг.1.3. Категории протоколи за динамична маршрутизация*

**1.2.6. Border Gateway Protocol**

Border Gateway Protocol (BGP) е Path-vector маршрутизиращ протокол, позволяващ осъществяването на връзката между отделните автономни системи (AS). Автономните системи са група от маршрутизатори, които са под управлението на една или повече компании с еднакви политики за маршрутизатори, като използват вътрешни протоколи за маршрутизация за комуникация помежду си и външни за комуникация с другите автономни системи. Има два вида BGP – internal BGP (iBGP) за вътрешна маршрутизация и external BGP (eBGP) за външна маршрутизация. При имплементация на eBGP, маршрутизаторите трябва да са директно свързани, защото TTL (Time to Live) параметърът BGP съобщенията е със стойност 1, което означава че съобщението може да стигне само до един маршрутизатор, след като е изпратено. Aко e пусната функционалността „multi-hop“, TTL стойността е по-голяма от 1. На Фиг.1.4 е означена свързаността между три автономни системи, посредством eBGP и iBGP.

*****фиг.1.4. Свързаност между три автономни системи*

Протоколът има четири версии, като последната версия има функциолността да пренася и IPv6 трафик. Поддържа се Classless Inter Domain Routing (CIDR), което позволява да се използват мрежи с променлива маска на мрежата (subnet mask), като BGP поддържа механизъм за агрегиране на мрежите до класово разделяне на IP адресите. По този начин се спестяват изчеслителни ресурси и се разтоварва трафика от мрежата. BGP работи с Transport Control Protocol (TCP) за транспортен протокол за обмен на съобщения и работи на порт 179. Маршрутизаторите използващи BGP се наричат „BGP speakers“, а тези, установили TCP сесия, се наричат „peer” маршрутизатори. Първоначално, при стартиране на мрежата с BGP, всички маршрутизатори разменят пълните си маршрутизиращи таблици и след това започват да изпращат съобщения, ако е настъпила промяна в таблиците.

В зависимост от броя на автономните системи, има възможност BGP да не е препоръчително да се използва. Топологиите биват:

* **Single-homed** – това е топология, при която автономна система има връзка само с една друга автономна система. За предпочитане е да се използват статични пътища. Предимството на тази топология е че е изгодна откъм ценови аспект. Големият недостатък обаче е че липсва резервираност, понеже е само една връзка и ако се появи проблем и тя спре да функционира, няма да има друг път по който да се осъществи свързаност. На Фиг.1.5. е показана single-homed топология.

*Фиг.1.5. Single-homed топология*

* **Multi-homed** – това е топология, при която автономна система има връзка с повече от една автономна система. При това положение е подходящо да се използва eBGP. Основно предимство на тази топология е, че за разлика от single-homed, има по-добра резервираност поради наличието на повече от 1 връзка между автономните системи. На Фиг.1.6. е показана multi-homed топология.

*Фиг.1.6. Multi-homed топология*

BGP използва четири типа съобщения, в зависимост от състоянието на мрежата:

* **OPEN** – първото съобщение, което се изпраща, след като се установи TCP сеисията. Чрез него се изпраща информация за версията на протокола, номера на автономната система, времето на изчакване (Hold time), в което е нужно съседния маршрутизатор да отговори, както и адреса на BGP speaker.
* **KEEPALIVE** – изпраща се от отсрещната страна, след като се приеме OPEN съобщението. Този тип съобщение се изпраща в интервал от 60 секунди по подразбиране, за да поддържат връзката.
* **UPDATE** – използва се, за да се обменя маршрутизираща информация. Изпращат се мрежовия адрес и дължината на subnet маската, информация за пътя и недостъпните пътища.
* **NOTIFICATION** – съобщение което се изпраща към всички BGP маршрутизатори в автономната система в случай на грешка, като за всеки тип грешка има номер, указан в съобщението.

При имплементация на BGP е препоръчително и най-добре да се използва „loopback“ интерфейс (логически интерфейс, който е постоянно активен, освен ако не бъде деактивиран от мрежовият администратор), с който се гарантира, че мрежовия адрес на маршрутизатора ще е постоянно достъпен.

**1.2.6.1. Border Gateway Protocol Path Attributes**

За избирането на най-добър път до отдалечената автономна система, при използването на BGP, се гледат атрибутите, които имат даден път до избраната дестинация. Те се разделят на четири категории:

* **Well-known mandatory** – това са параметри, които задължително трябва да се разбират от всички маршрутизатори, използващи BGP. Те трябва да се съдържат във всяко едно BGP UPDATE съобщение.
* **Well-known discretionary** – това са параметри, които задължително трябва да се разбират от всички маршрутизатори, използващи BGP, но е възможно и да не се включват в BGP UPDATE съобщенията.
* **Optional transitive** – това са параметри, за които не е задължително да се разбират от всички маршрутизатори, използващи BGP, но трябва да се приемат и да се предадат на съседните маршрутизатори.
* **Optional non-transitive** – това са параметри, за които не е задължително да се разбират от всички маршрутизатори, използващи BGP, но може да изпусне BGP UPDATE съобщението и да не го предаде на съседните маршрутизатори.

Маршрутизаторите, използващи BGP, взимат решението за това по какъв начин определена информация да достигне дестинацията си. Решението се взима на базата на параметрите от четирите категории. Даден маршрут се избра по определен ред на атрибутите на пътя. Съществуват десет стандартни атрибута, но се практикуват само седем, които са най-често използваните при имплементация на BGP:

* **Local preference (LOCAL PREF)** – използва се при пренос на информация вътре в автономната система. Указва най-добрият път за изход на автономната система. Този атрибут се конфигурира от мрежовият администратор и действа на базата на приоритет. Пътят с най-висок приоритет бива избран за основен BGP път към определената мрежа, през който да премине трафикът, а всички други за резервни пътища, които се активират, когато основният отпадне. Колкото е по-голяма стойността на приоритета, толкова по-предпочитан пътя.
* **AS\_PATH** – това е задължителен атрибут (от категория well-known mandatory), който указва през кои автономни системи преминава информацията. Този атрибут се използва, за да се предотвратят цикли в мрежата.
  + 1. **IS-IS**
    2. **BGP**
       1. **Route Reflection**
       2. **MPBGP**

1. **MPLS**
2. **LDP**
3. **VRF**
4. **QoS**

**Втора** **глава: Проектиране на физическата реализация на MPLS VPN мрежова архитектура**

* 1. **Основни изисквания към мрежовата топология**
     1. **Изграждане на MPLSVPN архитектура за две различни компании**
     2. **Използване на протокол LDP за дистрибуция на лейбъли**
     3. **Употреба на IS-IS протокол в MPLS мрежата**
     4. **Използване на BGP и MPBGP за реализация на динамична маршрутизация**
     5. **Използване на Route Reflector в MPLS мрежата**
     6. **Осигуряване на резервираност на всеки от клиентите на ниво доставчик на услугата**
     7. **Внедряване на политика за качество на услугата - Quality of Service**
     8. **Симулация на мрежовото решение на GNS3**
  2. **Описание на мрежовата топология**
  3. **Адресация на мрежовата топология**

**Трета** **глава: Симулация на MPLS VPN мрежовата архитектура на GNS3**

* 1. **Основна конфигурация на мрежовите устройства**
  2. **Конфигуриране на протокол за динамична маршрутизация IS-IS**
  3. **Конфигуриране на протокол за динамична маршрутизация BGP**
     1. **Конфигуриране на тип iBGP топология Route Reflector**
     2. **Конфигуриране на протокол за мултифункционална динамична маршрутизация MPBGP**
  4. **Конфигуриране на протокол за дистрибуция на етикети LDP**
  5. **Конфигуриране на протокол за комутация на етикети MPLS**
  6. **Конфигуриране на VRF таблица**
  7. **Конфигуриране на политики за качество на услугите QoS**

**Четвърта глава: Тестване на работоспособността на мрежовата архитектура**

* 1. **Тестване на свързаността между отдалечено свързаните компании**
  2. **Тестване на функционалността на протокола BGP**
  3. **Тестване на функционалността на протокола IS-IS**
  4. **Тестване на функционалността на протокола MPLS**
  5. **Тестване на функционалността на протокола LDP**
  6. **Тестване на функционалността на VRF**
  7. **Тестване на функционалността на QoS**

**Заключение**

**Използвана литература**

* <http://www.steves-internet-guide.com/ipv4-basics/>
* <https://protechgurus.com/how-does-isis-protocol-work-explained/>
* <https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_isis/configuration/15-mt/irs-15-mt-book/irs-ovrw-cf.html#GUID-64960B8C-24DB-4929-A7BD-7C308120A650>
* <https://www.networkworld.com/article/2297171/network-security-mpls-explained.html>
* <https://www.metaswitch.com/knowledge-center/reference/what-is-label-distribution-protocol-ldp>
* <https://www.cisco.com/c/en/us/td/docs/ios/mpls/configuration/guide/12_2sr/mp_12_2sr_book/mp_ldp_overview.html#wp1354663>
* <https://www.imperva.com/blog/bgp-routing-explained/>
* <https://www.cloudflare.com/learning/security/glossary/what-is-bgp/>
* <https://www.fir3net.com/Networking/Protocols/what-is-a-bgp-route-reflector.html>
* <http://www.networkers-online.com/blog/2009/02/bgp-route-reflector-basics/>
* <https://searchunifiedcommunications.techtarget.com/definition/QoS-Quality-of-Service>
* <https://www.plixer.com/blog/what-is-vrf-virtual-routing-and-forwarding/>
* <https://tools.ietf.org/html/rfc4271>
* <https://tools.ietf.org/html/rfc3031>
* <https://tools.ietf.org/html/rfc3813>
* <https://tools.ietf.org/html/rfc5036>
* <https://tools.ietf.org/html/rfc2547>
* <https://tools.ietf.org/html/rfc5777>

**Съдържание**