

SZOFTVER ÁLTAL DEFINIÁLT HÁLÓZAT – A JÖVŐ HÁLÓZATA

SOFTWARE DEFINED NETWORK – THE NETWORK OF THE FUTURE

Agg Péter András ^{1*}, Johanyák Zsolt Csaba ¹, Kovács Szilveszter ²

¹ Informatika Tanszék, GAMF Műszaki és Informatikai Kar, Pallasz Athéné Egyetem, Magyarország

² Általános Informatika Tanszék, Miskolci Egyetem, Magyarország

Kulcsszavak:

SDN
Szoftver által definiált hálózat
SDN rétegek
OpenFlow
Hálózatok

Keywords:

SDN,
Software-Defined Network
SDN planes
OpenFlow
Networks

Cikktörténet:

Beérkezett 2016. szeptember 26
Átdolgozva 2016. október 17.
Elfogadva 2016 október 20.

Összefoglalás

A számítógépes hálózatok megvalósítása során fontos szempont a kezelhetőség, a felügyelhetőség, és a konfigurálhatóság. A szoftver által definiált hálózatok (Software Defined Networks - SDN) paradigma biztosítja ezeket a jellemzőket. Az SDN különválasztja az adatsíkot és a vezérlő síkot, miáltal könnyebbé válik a hálózat megvalósítása. Az új megközelítés szerint a fizikai eszközök csak egyszerű adattovábbítási feladatot látnak el, míg a vezérlés központosításra kerül. Az OpenFlow protokoll alkalmazásával a hálózati eszközök csak a csomag küldését és fogadását végzik, az útvonalválasztás egy felsőbb réteg segítségével történik. Az SDN koncepcióval jelentős hatékonyság javulás érhető el az erőforrás gazdálkodás, az automatizálás és a biztonság területén is. Cikkünkben bemutatjuk az SDN rétegeit, funkcióit, és kitérünk a hagyományos hálózatokkal szembeni előnyökre. Bemutatjuk az SDN által használt OpenFlow protokollt is.

Abstract

In course of the design and implementation of computer networks important aspects are the usability, manageability, and configurability. The new paradigm of Software Defined Networks (SDN) ensures the fulfilment of these requirements. SDN separates the control plane and the data plane to facilitate network implementation. Thus the data plane of the physical devices performs only simple data transfer functionality, while the control is centralized. Using the OpenFlow protocol network devices are used for sending and receiving packages, while routing is performed in higher layers. The SDN network concept can improve significantly the efficiency of resource management, automation and security. This article presents the layers of SDN and their functionality as well as the OpenFlow protocol. Advantages compared to traditional networks are also discussed.

* Kapcsolattartó szerző. Tel.: +36-76-516-418
E-mail cím: agg.peter@gamf.kefo.hu

1. Bevezetés

Napjaink minden területét átjárja az informatika, ezen belül a számítógépes hálózatok használata. Alapvető elvárás a felhasználók szemszögéből, hogy hálózatuk gyors, megbízható és biztonságos legyen. A hagyományos hálózati infrastruktúrák hátrányainak, hibáinak feloldására komoly erőfeszítéseket tesznek a területtel foglalkozó szakemberek. Az alapvető gond az, hogy a hálózat konfigurációja nem szakadt el a fizikai alapoktól, így nem elég hatékony a folyamatok végrehajtása, illetve lassú az eszközök konfigurálása is.

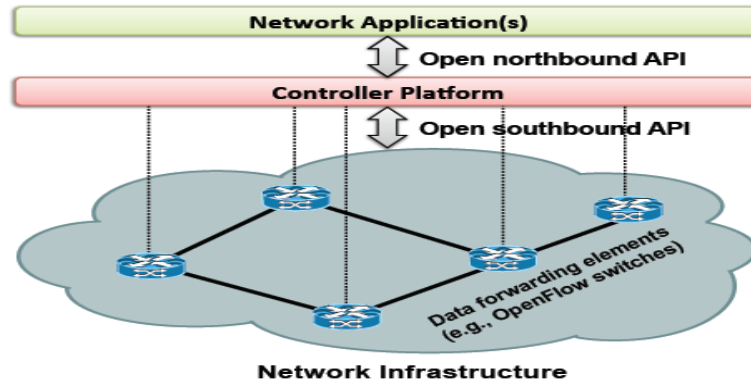
A hálózatok működéséhez feltétlenül szükség van mind hardver, mind szoftver elemekre. A használt aktív eszközök, mint például a forgalomirányítók és a kapcsolók, meghatározott protokollok segítségével biztosítják a megfelelő adatáramlást. Ennek kialakítása és konfigurálása annak ellenére magas szintű képzettséget igényel, hogy az IP hálózatokat az élet minden területén használjuk, mindenhol jelen vannak. Elengedhetetlen az eszközök maximális ismerete, hiszen vezérlésük erősen gyártó specifikus, ami nem könnyíti meg az irányítást. A mai szabványos hálózati kialakításra jellemző a központosítás hiánya és a vertikális integrálás, ami hátráltatja a hatékony működést, a biztonságos felügyelhetőséget, és a költségek radikális csökkentését. Ezen problémák kiküszöbölésére léteznek már megoldások, amelyek részben leegyszerűsítik a konfigurálást. Az OSI 2. rétegében [6] működő kapcsolók esetében például használható a virtuális helyi hálózat (virtual local area network - VLAN) ami az eszközök egy csoportját foglalja magába, és amelynek tagjai úgy képesek kommunikálni egymással, mintha azonos szegmensbe tartoznának annak ellenére, hogy fizikailag több különböző szegmensben helyezkednek el. Ehhez szükség van a CISCO által kifejlesztett VLAN Trunking Protocol-ra (virtuális trónkprotokoll - VTP) [12]. A VTP egy ügyfél-kiszolgáló alapú üzenettovábbító protokoll, amely lehetővé teszi, hogy egyszerre konfiguráljuk a hálózat VLAN konfigurációjának azon részét, amely egy VTP tartományhoz tartozik. A forgalomirányítóknál hasonló „központosított” irányítást (elsősorban biztonsági szempontból) valósíthatnak meg a hozzáférés-vezérlési listák (Access Control List ACL) [2], melyek megkönnyítik a rendszergazdák feladatait, de nem mentesíti őket az alól, hogy a helyi hardvernek megfelelő gyártófüggő beállításokat is alkalmazzák.

A hálózati infrastruktúra megújulását a szoftver által definiált hálózatok (Software-Defined Networks - SDN) [3] megjelenése jelentette. Az SDN-t támogató hálózatoknál az adatsík és a vezérlő sík elszakad egymástól, így a hálózati architektúra kívülről, programozási interfészekon (Application Programming Interface - API) keresztül automatizálhatóvá és programozhatóvá válik. A szerver oldali eszközök az API-k révén a hálózat funkcióihoz is hozzáférhetnek, ezáltal megvalósítható az újraprogramozás és a felhasználók igényeinek megfelelő beállítások. Továbbá lehetőség nyílik az automatizálásra is, melynek eredményeképpen csökkenteni lehet a humán erőforrásra szánt kiadásokat, illetve jobban kiküszöbölhetők az emberi hibákból eredő kockázatok, amelyek a hagyományos hálózatnál a legjelentősebb veszélyforrást jelentik. Az SDN koncepció segítségével gyors, hatékony alkalmazástelepítés, beüzemelés valósulhat meg, ami elősegíti, hogy megakadályozzuk az IT-infrastruktúrában a kimaradást, fennakadást, ami gyakorlatilag az informatika minden területén alapvető elvárás a felhasználók részéről.

Cikkünkben bemutatjuk az SDN rétegeit, funkcióit. Kitérünk a hagyományos hálózatokkal szembeni előnyökre, mind hardveres, mind szoftveres területen. Bemutatjuk az SDN által használt OpenFlow protokollt is.

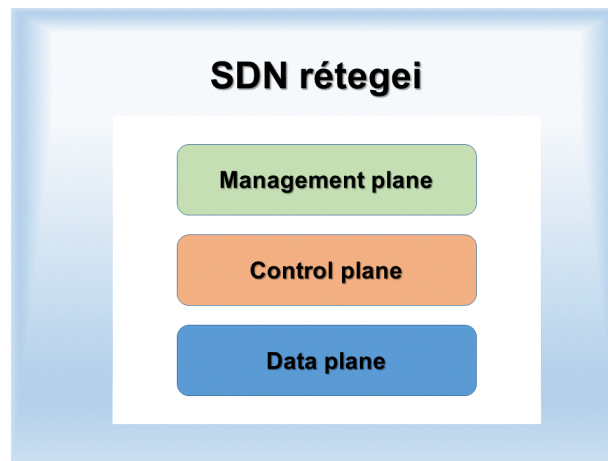
2. Software-Defined Networking (SDN) bemutatása

A szoftver által definiált hálózatok (SDN) [3] egy olyan feltörekvő hálózati paradigma, melynek az egyik legfontosabb jellemzője, hogy szétválasztja a vezérlési síkot (control plane) és az adatsíkot (data plane) (eszközsíkot) az eszközökben, így a hálózati eszközök csak továbbító eszközökként működnek, a vezérlés pedig központosításra kerül. Ez az elkülönítés biztosítja a megfelelő irányíthatóságot a programozható interfészekon keresztül. A legtöbb gyártó jelenleg még az úgynevezett hibrid megoldást alkalmazza [4], azaz OpenFlow [9][10] kompatibilis eszközöket állít elő, de nem engedi a teljes vezérlés átvételét az eszköztől. Az SDN gyakorlatilag elfedi a hardver elemeket a felsőbb szintek elől, mint az 1. ábrán látható.



1. ábra. SDN architektúra [3]

Az SDN három fő réteget definiál [3]. Ezek az adatsík (Data plane), a vezérlő sík (Control plane), és a menedzsmint sík (Management plane) (2. ábra.)



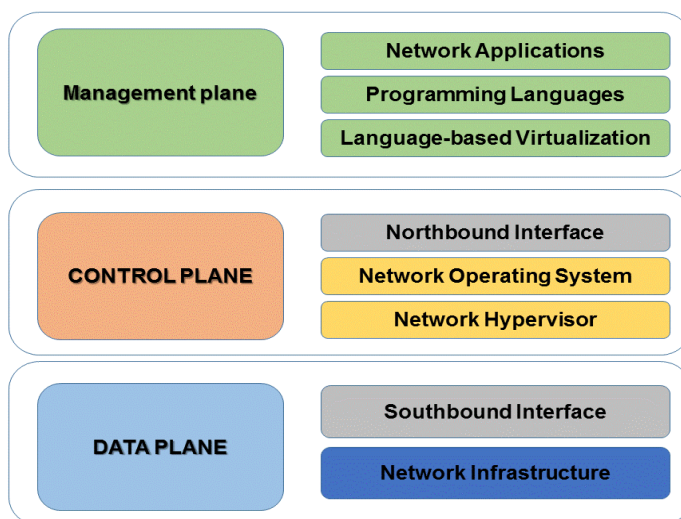
2. ábra. SDN rétegek

Az adatsík biztosítja a hálózati csomagok mozgatását az egyes hálózati szegmensek vagy végpontok között. Két fő részre bontható (3. ábra). A hálózati infrastruktúra (Network Infrastructure) és a déli interfész (Southbound interface). A hálózati infrastruktúra gyakorlatilag megegyezik az korábbi hálózatok hálózati eszközeivel, azzal a kiegészítéssel, hogy az eszközöknek rendelkezni kell egy a felsőbb rétegek felé nyitó lehetőséggel, ami által vezérelhetők lehetnek (pl.: OpenFlow protokoll). Ez a szabvány pontosan leírja, mit kell tenni a csomaggal [9]. A déli interfészek (Southbound API) hidat képeznek az adat és a vezérlő sík között. Bár erősen függenek a hardvertől, de az OpenFlow protokoll elterjedésének köszönhetően, a gyártók kompatibilissé tették az eszközeiket, ami nagymértékben megkönnyítette a hatékony hálózati infrastruktúra kialakítását. A déli interfészek meghatározzák, és kezelik azokat az információkat, amelyek fontosak hálózati operációs rendszerek számára (adatküldés port, link változás után, útvonalváltások jelzése, ismeretlen, kézbesítetlen csomagok hibái, stb.).

A vezérlő sík (szintén három részre osztható) (3. ábra) egyik legfontosabb alrétege a virtualizációs réteg. Az SDN-alapú hálózati virtualizáció [8] támogatja szabadon választott IP/MAC címzésrendszerek bevezetését, miközben megoldja az automatikus hálózati konfigurációs feladatokat is. A hálózat virtualizációja használja az OpenFlow protokoll adta lehetőségeket, mind a fizikai, mind a virtuális hálózati eszközök vonatkozásában. A hálózati operációs rendszer (Network Operating System) engedélyezi a dinamikus hozzáférést és a rendszeradminisztrációt, ezáltal a rendszergazda egy központi vezérlő konzolon keresztül alakíthatja a forgalmat úgy, hogy közvetlenül konfigurálja az adott hálózati eszközt. Az északi interfész (Northbound Interface) esetében ellentétben a déli interfésszel, ahol már az OpenFlow-t használják, még nagyon sok a nyitott kérdés, a szabványosítás még nem kiforrott.

A menedzsment sík három rétege (3. ábra) a nyelv alapú virtualizáció (Language-based virtualization), a programozási nyelvek (Programming language) és a hálózati alkalmazások (Network Applications). A menedzsment síkban magas szintű programozási nyelveket alkalmazhatunk létrehozva magasabb szintű absztrakciókat [1] annak érdekében, hogy egyszerűsítsük a feladatokat. Ezáltal a probléma-orientáltságra fókuszálva felgyorsíthatjuk az adatkommunikációt, és biztosíthatjuk az adatok áramlásának megbízhatóságát központi felügyelet mellett. Ebben a rétegben megoldhatjuk a szoftver modularizálását és a kód újrahasznosítását.

Fontos, hogy ez a réteg képes használni minden korábbi algoritmust, igaz nem feltétlenül az eddigi hálózati gyakorlattal azonos formában. Az OpenFlow lehetőséget nyújt az egyszerűbb, hatékonyabb hálózatkezelés megvalósítására.



3. ábra. SDN részletes architektúrta

3. OpenFlow

Az OpenFlow egy kommunikációs protokoll, amit a Stanford és a Berkeley egyetemek oktatói fejlesztettek ki annak érdekében, hogy lehetőséget nyújtson a hálózatok működésének felülírására, módosítására, egyfajta központosított vezérlés megvalósítására. Az OpenFlow protokoll alkalmazásával (adatsík déli interfésze használja, és kapcsolatot teremt a vezérlő síkkal) a hálózati eszközök csak a csomag küldését és fogadását végzik, az útvonalválasztás a felsőbb réteg segítségével történik.

Ez a viszonylag fiatal protokoll 2009. decemberében jelent meg az 1.0 verziószámmal. 1 flow (adatfolyam) táblával (tartalmazza, hogy mit kell tenni a bejövő csomaggal) [11] dolgozott és 12 csomagfejrész alapján illesztett. Napjainkban ennek a verzióknak a fejlesztése már nem zajlik. 2011 elején jelent meg az 1.1-es verzió, ami már több flow táblát kezel és biztosította például a VLAN támogatását. Még ebben az évben megjelent 1.2-es fejlesztés már támogatta az IPv6 használatát. A 2012 áprilisában megjelent OF 1.3 már támogatta a Provider Backbone Bridge-t (PBB) [7] amit IEEE 802.1ah [5] szabványként ismerünk. Az 1.4-es fejlesztés már biztosította az optikai portok támogatását, lehetőséget adott több flow tábla szinkronizálására. Napjainkban már 2.0 verzió fejlesztése zajlik.

4. Az SDN előnyei

4.1. Egyszerűbb hardver

Az SDN legfontosabb előnye az egyszerűbb hardver lehetősége. Azzal, hogy a gyártók egyre inkább támogatják az SDN eszközök gyártását és használatát, biztosítják, hogy a hardverben csak az adatsík marad, míg a vezérlő síkot, ami korábban minden eszközön jelen volt, leválasztották. Mivel az adatsík csak a forgalom továbbításával foglalkozik, az eszközök összetettsége nagyban csökkent, aminek közvetlen következménye az árcsökkenés. Ez mind az ipari, mind az oktatási környezetben jelentős költségvetés-csökkenést eredményezhet. Az üzemeltetési költségek szintűgy

csökkennek, hisz kevesebb lesz az áramfogyasztás azáltal, hogy nem kell külön vezérlő szoftvert futtatni az eszközön.

A fenntartási költségeken kívül a bekerülési költségek is csökkennek. Eddig az eszközök vételárának jelentős részét a gyártói szoftver licenzének megfizetése tette ki. Erre az SDN eszközöknél már nincs szükség, hisz az eszköz nem tartalmaz ilyen szoftvert. Fontos megjegyezni, hogy SDN környezetben a forgalomirányításért felelős intelligencia centralizált, egy SDN Controller-en keresztül történik. Ennek megfelelően a hagyományos számítógépes hálózatok esetében alkalmazott elosztott forgalomirányítással kapcsolatos intelligencia miatt alkalmazott komplex eszközökre sincs már szükség.

4.2. Erőforrás gazdálkodás

A hagyományos számítógépes hálózatok esetében, az OSI modell alkalmazási rétegében (7. réteg) futó szoftverek nincsenek tisztában azzal, hogy az őket kiszolgáló hálózaton hogyan alakul az aktuálisan elérhető szabad kapacitás. Ennek folyományaként gyakori lehet a hálózatok túlterheltsége. SDN hálózatoknál az OSI réteg-modell legfelső rétegében futó alkalmazások kommunikálni tudnak az SDN Controllerrel. Azáltal, hogy kapcsolatba tudnak lépni, információ továbbítható a hálózaton forgalmazni kívánt becsült adatmennyiségről, így a Controller figyelembe tudja venni ezt, és úgy tudja „konfigurálni” a hálózati eszközöket, hogy az aktuális forgalmi igénynek megfelelően legyenek kiszolgálva.

4.3. Automatizálás

Az SDN architektúra és megfelelő absztrakció felhasználásával olyan alkalmazások fejlesztése válik lehetővé, amelyek segítségével a fizikai hálózati infrastruktúra képes az azt igénybe vevő alkalmazások vagy felhasználók igényeihez alkalmazkodni és nem pedig fordítva. Ehhez a képességhez elengedhetetlenül szükségesek a déli illetve az északi interfészekben alkalmazott API-k. A déli (Southbound) API-kon tipikusan az OpenFlow protokollt értjük, amellyel az SDN Controller képes vezérelni az eszközöket. Ezért van szükség arra, hogy az eszközök OpenFlow kompatibilisek legyenek. Az északi interfész még nem teljesen kiforrott. A Northbound API olyan alkalmazás, mely az SDN Controller segítségével a hálózat vezérlését teszi lehetővé az alkalmazások számára.

4.4. Gyártói környezet, központi vezérlés

Az SDN előnye, hogy az olyan eszközök, melyek az OpenFlow protokollt képesek vezérelni az SDN Controller segítségével, vezérelni tudják bármelyik gyártó eszközét, amennyiben az eszközben implementálva van az OpenFlow protokoll megfelelő verziója. Elsődleges cél, hogy ne kelljen az eszközöket működésük szerint gyártónként külön csoportokba sorolni, kezelni (más szoftver, más vezérlési megoldás), hanem az SDN architektúrán alapuló hálózatot egy központi Controller segítségével tudják menedzselni, amelynek köszönhetően sokkal egyszerűbb lesz a hálózat megvalósításának folyamata.

4.5. Megbízhatóság, biztonság

Az SDN használatával megoldható az, hogy magasszintű programozási sémákat és szabályokat hozunk létre, melyeket a Controller fog lefordítani a fizikai hálózat függvényében, majd alkalmazni az OpenFlow protokoll segítségével. Az SDN Controller teljes felügyeletet gyakorol a hálózat fölött. A szükséges hozzáférési listák és biztonsági megszorítások érvényesítése lényegesen egyszerűbb és hatékonyabb feladat, mint a hagyományos számítógépes hálózatok esetében.

Ezen okok miatt az SDN bevezetése nagyvállalatok esetében komoly költségcsökkenést, dinamikusabb és hatékonyabb hálózatot, valamint kevesebb hibás konfigurációt és megbízhatóbb biztonsági előírásokat eredményezhet.

5. Hátrányok

Az SDN-nel kapcsolatban nehéz hátrányokat említeni. Természetesen, mint egy új, egyre jobban elterjedő paradigma, elsősorban az előnyök szempontjából vizsgálandó. Ha mindenképp

hátrányokat kell megemlíteni, akkor elsősorban az átállást említenénk. Nyilvánvalóan az eszközök olcsóbbak, hatékonyabbak lettek, de mind egy vállalat, mind egy oktatási intézmény esetében nagy költségvetési kiadást jelent az eszközök teljes vagy részleges cseréje. A részbeni eszközcsere vezérlési problémákat is okozhat. Az automatizálás szempontjából érdemes megemlíteni, hogy elképzelhető, hogy bizonyos fejlesztők SDN Controller-jei az OpenFlow protokoll mellett egy saját maguk által kifejlesztett protokollt is alkalmaznak. Ez a gyakorlat már ellentmond a gyártófüggetlenség elvének, ami az SDN kialakulásánál létfontosságú szempont volt. Az OpenFlow protokoll csak egy lehetséges mód az SDN architektúra bevezetésére, emellett gyártóspecifikus megoldások is elképzelhetőek a jövőben, ami szintén gondot okozhat a gyártófüggetlenség elvének követésében. A hagyományos hálózatok hátrányaként említettük a szakember szükségességét, azonban az SDN-nél is szükséges a magas fokú képzettség. Ennél a hálózati megoldásnál a centralizálás miatt elsősorban a programozás ismeretek fontosabbak.

6. Következtetések

Az SDN egy olyan új technológia, ami jelentősen megkönnyíti, felgyorsítja a hálózatok konfigurálását és működtetését. Az eddigi decentralizálás helyett a centralizálás jellemző rá, ami a rendszergazdák munkáját jelentősen egyszerűsítheti. Az SDN által használt rétegek, sokkal könnyebbé teszik a hálózat felügyeletét, ellenőrzését, miáltal az egyes eszközök konfigurációs beállításai automatizáltan jutnak érvényre. Az SDN folyamatos fejlődése maximálisan elősegíti a korszerű hálózatok kiépülését. Az SDN rétegstruktúrája lehetővé teszi, hogy az alsóbb rétegek elkülönüljenek a felsőbb rétegektől, ami biztosítja a költséghatékonyt, illetve a könnyebb felügyeletet. Cikkünkben bemutattuk az SDN előnyeit, amelyeket sok vállalat és oktatási intézmény hasznosíthat a közeljövőben. Véleményünk szerint sokkal több előnye, mint hátránya van ennek az új paradigmának. A hátrányok minden újítás esetében megjelennek, melyek leküzdése után, hosszú távon csak az előnyök kerülnek előtérbe. Nyilvánvaló, hogy az eszközök vásárlása megoldható, még ha hirtelen nagyobb teher is, de a későbbiekben a teljes bekerülési költség vonatkozásában maximálisan megtérülnek. A szakemberek képzése, alkalmazása miatt adódhat némi nehézség az átállás során, de hosszú távon ez a kérdés is könnyebben megoldható, mint a ragaszkodás a tradicionális hálózatokhoz.

7. Irodalomjegyzék

- [1] M. Casado, N. Foster, and A. Guha, "Abstractions for software-defined networks," *Commun. ACM*, vol. 57, no. 10, pp. 86–95, Sep. 2014.
- [2] Cisco IOS Security Configuration Guide, Release 12.2, http://www.cisco.com/c/en/us/td/docs/ios/12_2/security/configuration/guide/fsecur_c/sfcls.html [Hozzáférés: 2016.09.12.]
- [3] Fernando M. V. Ramos, D. Kreutz, Paulo Verissimo: Software-defined networks: On the road to the softwarization of networking https://www.researchgate.net/publication/284922281_Software-defined_networks_On_the_road_to_the_softwarization_of_networking [Hozzáférés: 2016. 09. 14.]
- [4] HP, "Hp SDN controller architecture," Hewlett-Packard Development Company, L.P., Tech. Rep., September 2013.
- [5] <http://www.ieee802.org/1/pages/802.1ah.html> [Hozzáférés: 2016. 09. 11.]
- [6] ISO standard 7498-1:1994: Information technology – Open Systems Interconnection – Basic Reference Model – The Basic Model [http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip) [Hozzáférés: 2016.05.08.]
- [7] http://www.juniper.net/techpubs/en_US/junos12.3/topics/concept/pbb-understanding.html [Hozzáférés: 2016. 09. 16.]
- [8] L. Kreeger, D. Dutt, T. Narten, and D. Black, "Network virtualization NVE to NVA control protocol requirements," Internet Draft, Internet Engineering Task Force, April 2014. [Online]. Available: <http://tools.ietf.org/html/draft-ietf-nvo3-nve-nva-cp-req-02>
- [9] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using OpenFlow: A survey," *IEEE Communications Surveys & Tutorials*, vol.16, no.1, pp.493,512, First Quarter 2014 <http://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1164&context=csetechreports> [Megtekintés: 2016.08.30.]
- [10] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.
- [11] <http://archive.openflow.org/documents/openflow-spec-v1.1.0.pdf> [Hozzáférés: 2016. 09. 04.]
- [12] Understanding VLAN Trunk Protocol (VTP) <http://www.cisco.com/c/en/us/support/docs/lan-switching/vtp/10558-21.html> [Hozzáférés: 2016. 09. 14.]