

ANALISIS PERANCANGAN ARSITEKTUR JARINGAN SMK NEGERI 3 KOTA BEKASI MENGGUNAKAN METODE NDLC DENGAN PEMANFAATAN SOFTWARE-DEFINED NETWORKING

Lucky Anis Januar Sigit¹, Agung Susilo Yuda Irawan², Didi Juardi³

^{1,2,3}Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang

Email:¹lucky.anis19027@student.unsika.ac.id

ABSTRACT

This study aims to implement the concept of Software-Defined Networking or SDN using an Open Network Operating System or ONOS controller using the GNS3 platform in government-owned educational institutions, namely SMK Negeri 3 Bekasi City. Implementation of the SDN concept can facilitate and streamline the work of network engineers in managing a computer network infrastructure. The method used in this study is the Network Development Life Cycle or NDLC. The ONOS controller used in this study functions as a central network controller that has been integrated with various APIs, the result is that network management can become easier and simpler. Then, monitoring is carried out using Wireshark, suitable for monitoring data packets on networks that have been integrated with the SDN concept. The topology used has also received changes to simplify the SMKN 3 Kota Bekasi network.

Keywords: GNS3, NDLC, ONOS, SDN, SMK Negeri 3 Kota Bekasi, Wireshark.

ABSTRAK

Penelitian ini memiliki tujuan untuk mengimplementasikan konsep Software-Defined Networking atau SDN menggunakan controller Open Network Operating System atau ONOS dengan penggunaan platform GNS3 pada instansi pendidikan milik pemerintah yaitu SMK Negeri 3 Kota Bekasi. Pengimplementasian konsep SDN ini dapat memudahkan serta mengefisienkan pekerjaan network engineer dalam mengelola suatu infrastruktur jaringan komputer. Metode yang dipakai dalam penelitian ini adalah Network Development Life Cycle atau NDLC. Controller ONOS yang digunakan dalam penelitian ini, berfungsi sebagai pusat pengendali jaringan yang sudah terintegrasi dengan berbagai macam API, hasilnya adalah manajemen jaringan dapat menjadi lebih mudah dan simpel. Lalu, monitoring yang dilakukan menggunakan Wireshark, cocok untuk pengawasan terhadap paket-paket data yang ada pada jaringan yang sudah terintegrasi dengan konsep SDN ini. Topologi yang digunakan juga mendapatkan perubahan untuk melakukan penyederhanaan dalam jaringan SMKN 3 Kota Bekasi.

Kata Kunci: GNS3, NDLC, ONOS, SDN, SMK Negeri 3 Kota Bekasi, Wireshark.

Riwayat Artikel :

Tanggal diterima : 20-08-2023

Tanggal revisi : 21-08-2023

Tanggal terbit : 22-08-2023

DOI :

<https://doi.org/10.31949/infotech.v9i2.6536>

INFOTECH journal by Informatika UNMA is licensed under CC BY-SA 4.0

Copyright © 2023 By Author



1. PENDAHULUAN

1.1. Latar Belakang

Penggunaan internet saat ini yang semakin banyak serta berkembangnya berbagai macam perangkat jaringan yang dikeluarkan oleh berbagai vendor penyedia perangkat jaringan, seperti router, switch, access point, dan perangkat jaringan lainnya, membuat pengelolaan jaringan menjadi semakin sulit dan kurang efisien penggunaan sumber dayanya. Begitupula topologi-topologi jaringan konvensional yang ada saat ini banyak yang belum bisa berjalan seiringan dengan teknologi baru pada perangkat jaringan, sehingga semakin sulit untuk menunjang pemakaian internet yang semakin masif. Itulah yang membuat para peneliti harus menemukan suatu konsep arsitektur jaringan baru yang dapat mengatasi berbagai permasalahan tersebut.

Pada konsep pendekatan rancangan arsitektur jaringan konvensional yang telah kita ketahui dan telah banyak digunakan pada saat ini juga masih ditemukan beberapa kekurangan yang sangat berpengaruh terhadap jalannya arus lalu lintas atau traffic jaringan yang digunakan, seperti biaya operasional yang tinggi, kerumitan dalam melakukan manajerial atau pengelolaan jaringan, permasalahan dalam network scalability, arsitektur yang tidak terpusat atau desentralisasi, dan lain sebagainya (Yasmin, et al., 2019). Untuk itulah para peneliti menemukan suatu konsep pendekatan rancangan arsitektur jaringan baru yaitu Software-Defined Networking sebagai salah satu arsitektur jaringan komputer yang dapat mengatasi permasalahan jaringan yang ada tersebut dengan menawarkan beberapa kemudahan dengan sistem pengendali yang terpusat pada bagian controller sebagai control plane yang berfungsi untuk melakukan pengaturan, pengelolaan, dan monitoring terhadap seluruh traffic yang ada pada jaringan (Hidayat, et al., 2018).



Gambar 1 SMKN 3 Kota Bekasi

SMKN 3 Kota Bekasi sendiri memiliki wilayah yang cukup luas dan unik. Terdiri dari 4 gedung utama yang masing-masing memiliki minimal 2 lantai dengan jarak antar gedung yang

cukup jauh. Dengan kondisi geografis pada SMK Negeri 3 Kota Bekasi yang memiliki keunikan dan kerumitan tersebut, pengelolaan dan pengkonfigurasi konvensional yang saat ini digunakan menjadi kurang efisien dari segi waktu dan tenaga. Jika dilihat juga pada topologi SMK Negeri 3 Kota Bekasi saat ini, terdapat pemisahan antara dua ISP dengan dua router yang berbeda membuat resource yang digunakan tidak efektif. Kurangnya sumber daya manusia dan tempat pun menjadi salah satu faktor yang sangat mempengaruhi topologi yang ada saat ini. Untuk itu diasumsikan bahwa SMK Negeri 3 Kota Bekasi membutuhkan suatu arsitektur jaringan baru yang dapat menjadi solusi bagi efisiensi pengelolaan jaringan di SMK Negeri 3 Kota Bekasi yaitu Software-Defined Networking.

Jika dilihat pada penelitian yang telah diteliti sebelumnya oleh (Yasin, 2020), Pengujian pada penelitian ini menghasilkan kesimpulan bahwa dampak serangan DDoS pada RAM, delay, dan packet loss terhadap GNS3 lebih besar dibandingkan dengan serangan DDoS terhadap Mininet. Pemantauan yang dilakukan oleh peneliti menggunakan beberapa aplikasi monitoring, seperti Wireshark, sFlow, dan NetFlow pada Mininet dan GNS3, menunjukkan bahwa serangan ini berdampak pada resources yang ada di software OpenVSwitch dalam penggunaan CPU dan RAM, akibatnya software OpenVSwitch mengalami kegagalan dalam pengoperasian untuk sementara waktu karena sistem melakukan restart secara otomatis.

Penelitian lain yang dilakukan oleh (Pratama, dan Wikantyasa, 2019), mengenai pengimplementasian serta analisis pada simulasi Quality of Services (QoS) serta performance device menggunakan controller ONOS dan tools Iperf3. Adapun hasil kesimpulan yang didapat dari penelitian tersebut antara lain, peneliti berhasil mendapatkan keterangan nilai packet loss, throughput, bandwidth, serta jitter tiap host dari pengujian topologi buatan yang dibuat berdasarkan bahasa pemrograman Python dengan menggunakan API Mininet. Peneliti melakukan pengujian kedua dengan melakukan analisis pada perangkat Openflow switch dengan menyaring packet pada traffic sehingga berhasil memperoleh data packet send and received, dan byte send and received.

Penelitian terdahulu yang dicantumkan diatas menunjukkan bahwa arsitektur jaringan Software-Defined Networking dapat mengefisienkan pengelolaan serta pengkonfigurasi perangkat-perangkat jaringan di dalam pengimplementasian jaringan komputer.

1.2. Tinjauan Pustaka

a. Jaringan Komputer

Jaringan komputer merupakan kumpulan dari dua atau lebih komputer yang dapat saling berkomunikasi satu sama lain menggunakan berbagai macam perangkat jaringan tambahan atau

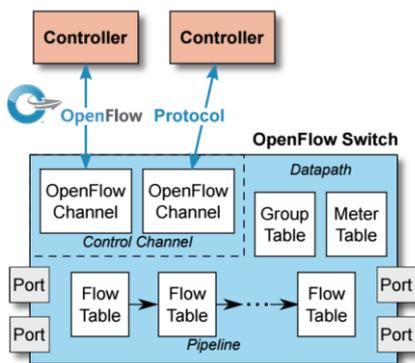
perangkat-perangkat jaringan yang dipakai sebagai suatu sistem komunikasi di dalam jaringan (Pamungkas, et al., 2018). Dalam jaringan komputer, ada beberapa perangkat-perangkat jaringan yang umumnya digunakan untuk menghubungkan perangkat satu dengan perangkat yang lain, diantaranya adalah modem, access point, hub atau switch, Network Interface Card (NIC), dan router (Micro, 2012).

b. Software-Defined Networking

Software-Defined Networking atau SDN merupakan salah satu konsep baru dalam memudahkan pembangunan, perancangan, pengelolaan, serta pengimplementasian jaringan yang saat ini sudah semakin kompleks, dengan konsep dasar memisahkan dua bagian network yaitu bagian data plane dan bagian control plane. (Risdiyanto, et al., 2016). Software-Defined Networking memiliki konsep dasar pengefisienan dengan memisahkan dua bagian network yaitu bagian forwarding plane atau data plane sebagai pengeksekusi keputusan dalam jaringan (decision execution), dan bagian control plane sebagai pembuatan keputusan dalam jaringan (decision making) (Narayana, 2019).

c. Openflow

OpenFlow merupakan standar protokol yang bertugas untuk menghubungkan control plane dan data plane pada arsitektur Software-Defined Networking dengan memberikan izin untuk melakukan akses secara langsung ke data plane melalui perangkat jaringan seperti switch dan router baik secara physical maupun secara virtual (Melissa, 2018).



Gambar 2 Struktur OpenFlow

d. Open Network Operating System (ONOS)

Open Network Operating System atau ONOS merupakan salah satu controller yang ada pada arsitektur SDN dalam bentuk sistem operasi pada jaringan yang bersifat open-source atau terbuka, dan diutamakan untuk network operator (Eueung Mulyana, 2016). ONOS juga merupakan salah satu SDN Controller yang memiliki tampilan GUI (Graphical User Interface) untuk memudahkan operator dalam melakukan konfigurasi ataupun pengelolaan pada perangkat yang mendukung protokol SDN OpenFlow (Saragi, et al., 2017).

e. GNS3

Graphical Network Simulator-3, atau yang biasa disebut juga GNS3, merupakan suatu aplikasi yang dapat digunakan oleh pengguna untuk mensimulasikan jaringan komputer secara kompleks, nyata dan mendekati aslinya (Abdullah, 2017).

f. Wireshark

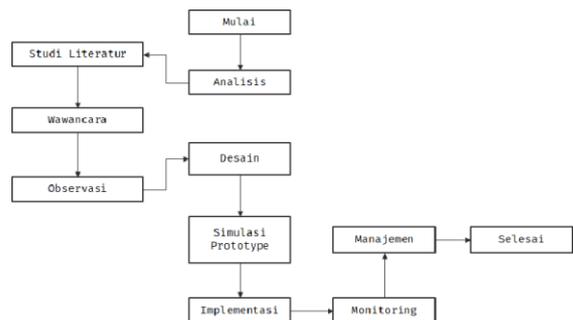
Wireshark merupakan salah satu aplikasi yang dapat digunakan oleh network administrator untuk melakukan monitoring terhadap kualitas jaringan yang sedang dikelola (Maulana, et al., 2021). Wireshark akan melakukan analisis terhadap paket data yang terdapat dalam jaringan yang sedang beroperasi, dan akan langsung dilakukan pensortiran informasi, baik informasi paket data secara singkat maupun detail (Mahmud, et al., 2019).

1.3. Metodologi Penelitian

Perancangan, pembangunan, serta pengelolaan sistem jaringan yang dilakukan menggunakan metode Network Development Life Cycle (NDLC) yang memiliki 6 tahapan, mulai dari Analysis, Design, Simulation Prototyping, Implementation, Monitoring, dan Management. Untuk metode pengumpulan data yang digunakan pada penelitian ini adalah metode observasi dan metode wawancara.

Observasi yang dilakukan adalah dengan mengunjungi SMK Negeri 3 Kota Bekasi secara langsung untuk melihat perangkat-perangkat jaringan yang sedang digunakan, mulai dari jenis perangkat, merk perangkat, dan media transmisi yang digunakan. Observasi juga dilakukan guna melihat topografi dan denah lokasi yang akan dilakukan implementasi secara virtual. Pengamatan jalannya sistem jaringan yang ada pada SMK Negeri 3 Kota Bekasi sebelum dilakukannya penelitian juga dilakukan.

Wawancara yang dilakukan adalah dengan mewawancarai guru-guru di jurusan Teknik Komputer dan Jaringan serta teknisi-teknisi jaringan yang ada pada bagian ICT SMK Negeri 3 Kota Bekasi secara langsung.



Gambar 3 Alur Metodologi Penelitian

a. Analisis

Tahapan ini melingkupi 3 kegiatan, yaitu studi literatur, observasi, dan wawancara. Tahapan analisis ini bertujuan untuk menganalisis kebutuhan

yang dibutuhkan oleh pengguna dan peneliti untuk mendukung serta menjadi referensi dalam penelitian yang sedang dijalankan.

b. Desain

Perancangan desain dilakukan untuk merancang alur topologi jaringan sesuai dengan observasi yang telah dilakukan pada SMK Negeri 3 Kota Bekasi. Perancangan desain topologi jaringan dilakukan dengan menggunakan tools seperti Microsoft Visio, ataupun perancangan desain secara manual sesuai dengan aturan dan simbol-simbol yang berlaku.

c. Simulasi

Simulasi dilakukan untuk mensimulasikan topologi yang telah dirancangan. Simulasi dilakukan secara sederhana menggunakan Cisco Packet Tracer untuk menghubungkan jaringan-jaringan lokal maupun jaringan yang terhubung ke internet. Pada simulasi ini juga dilakukan pengkonfigurasi secara sederhana untuk melihat alur koneksi pada jaringan.

d. Implementasi

Implementasi dilakukan untuk membangun jaringan yang sudah disimulasikan. Implementasi dilakukan secara virtual menggunakan alat bantu software GNS3. Pengimplementasian secara virtual dilakukan dengan alasan tidak adanya dana untuk membeli perangkat yang dibutuhkan.

e. Monitoring

Monitoring merupakan langkah yang harus dilakukan jika suatu jaringan sudah selesai dibangun atau diimplementasikan. Langkah ini dilakukan untuk mengawasi jalannya arus lalu lintas jaringan yang telah dibangun. Tools yang digunakan untuk mengawasi arus lalu lintas paket data jaringan adalah Wireshark.

f. Manajemen

Tahapan ini merupakan tahapan paling akhir yang dilakukan untuk mengelola jaringan, menambah ataupun mengurangi perangkat yang terpasang, memperbaharui kebijakan dalam jaringan, mengurangi ataupun menambah bandwidth jaringan, dan sebagainya..

2. PEMBAHASAN

Tahapan simulasi dilakukan menggunakan tools Cisco Packet Tracer dari Cisco, aplikasi ini akan mensimulasikan topologi jaringan yang telah didesain. Implementasi dilakukan secara virtual menggunakan tools GNS3 dengan perangkat-perangkat jaringan yang tersedia pada tools tersebut.

Controller yang digunakan sebagai pengendali pada konsep sistem Software-Defined Networking adalah ONOS atau Open Network Operating System yang memiliki tampilan GUI agar memudahkan operator jaringan dalam

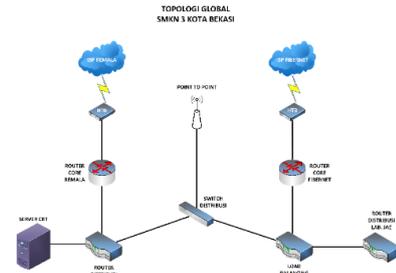
melakukan konfigurasi terhadap sistem SDN yang sedang dijalankan.

2.1. Tahap Analisis

Analisis berguna untuk mengumpulkan data-data yang dibutuhkan dalam perancangan arsitektur jaringan Software-Defined Networking menggunakan controller ONOS (Open Network Operating System) yang akan memudahkan peneliti untuk mengkonfigurasi jaringan komputer yang baru. Analisis akan dibagi menjadi 2 tahap, diantaranya:

a. Tahap yang sedang berjalan

Tahapan ini menjelaskan mengenai kekurangan yang ada pada topologi jaringan konvensional di SMK Negeri 3 Kota Bekasi. Tahap analisis dilakukan dengan memberikan pertanyaan-pertanyaan teknis seputar infrastruktur jaringan yang ada pada SMK Negeri 3 Kota Bekasi kepada teknisi yang ada di lokasi.

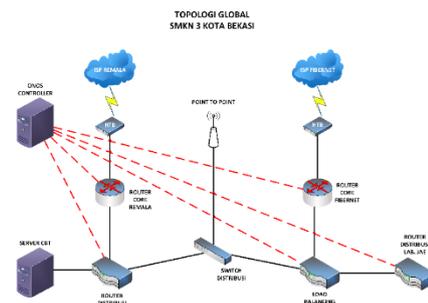


Gambar 4 Topologi Global SMKN 3 Kota Bekasi

Terdapat ketidakefisienan yaitu terdapat 2 ISP yang terkoneksi pada 2 router yang berbeda di tempat yang berbeda. Secara teknis, topologi ini berfungsi dengan baik, tetapi waktu yang dibutuhkan untuk mengkonfigurasi perangkat jaringan, khususnya router, pada tempat yang berbeda dan saling berjauhan satu sama lain menjadi sangat tidak efisien bagi operator jaringan.

b. Tahap perancangan jaringan baru

Setelah mengamati dan menganalisis kebutuhan pada arsitektur jaringan yang ada, akan dilakukan penambahan arsitektur jaringan berbasis Software-Defined Networking menggunakan kontroler ONOS sebagai pusat pengendali dan manajemen dari masing-masing perangkat jaringan utama, seperti router.



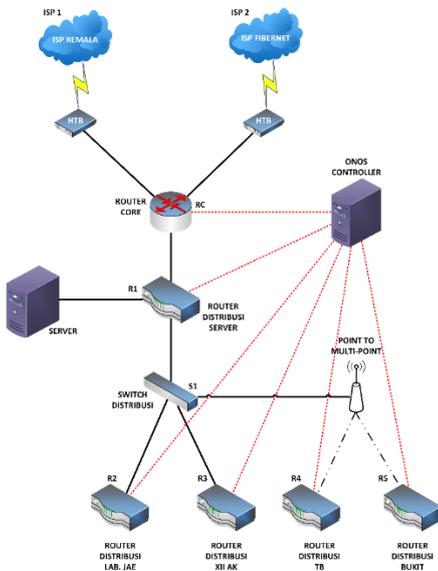
Gambar 5 Rancangan Arsitektur Jaringan SMKN 3 Kota Bekasi

2.2. Tahapan Desain

Tahapan ini dilakukan untuk membangun suatu topologi jaringan agar proses penempatan dan implementasi arsitektur jaringan dapat dengan mudah dipahami. Topologi yang ada akan menjadi gambaran hal-hal yang dibutuhkan pada penelitian ini. Tahapan ini dibagi menjadi 3 perancangan, antara lain:

a. Perancangan desain sistem dengan ONOS controller

Desain topologi yang digunakan akan mengalami perubahan, yaitu penambahan perangkat antena directional point-to-point yang akan menghubungkan beberapa router yang saling berjauhan. Rancangan topologi SMKN 3 Kota Bekasi yang baru memiliki 1 router core untuk 2 ISP yang digunakan. Topologi ini dirancang sesuai dengan analisis kebutuhan yang telah didapatkan dengan pengumpulan informasi dan data melalui wawancara serta observasi pada SMKN 3 Kota Bekasi.



Gambar 6 Rancangan Topologi Global SMKN 3 Kota Bekasi

b. Perancangan pengalaman IP address

Pengalaman IP address adalah suatu tahapan yang penting karena pengalaman ini berguna untuk mengenali setiap perangkat jaringan agar dapat saling berkomunikasi satu sama lain. Pengalaman IP address dilakukan mulai dari perangkat ISP sampai kepada host. Dibawah ini merupakan tabel yang menunjukkan pengalaman IP address pada sistem jaringan ini.

Tabel 1 Pengalaman IP Address

No	Nama	Port	Address	Keterangan
1	Router Core (RC)	1	200.20.45.2/30 (Cisco Packet Tracer)	Dari ISP 1
			210.20.45.2/24 (Mikrotik)	

		2	210.20.45.2/30 (Cisco Packet Tracer)	Dari ISP 2
			192.168.150.2/24 (Mikrotik)	
		3	192.168.1.1/24	Ke R1
2	Router Distribusi Server (R1)	1	192.168.1.2/24	Dari RC
		2	192.168.10.0/24	Ke R2, R3, R4, R5
		3	192.168.20.1/24	Ke Server
3	Router Distribusi Lab. Jae (R2)	1	192.168.10.2/24	Dari R1
		2	192.168.11.0/24	Ke Hosts
4	Router Distribusi XII AK (R3)	1	192.168.10.3/24	Dari R1
		2	192.168.12.0/24	Ke Hosts
5	Router Distribusi TB (R4)	1	192.168.10.4/24	Dari R1
		2	192.168.13.0/24	Ke Hosts
6	Router Distribusi Bukit (R5)	1	192.168.10.5/24	Dari R1
		2	192.168.14.0/24	Ke Hosts

c. Perancangan Kebutuhan Hardware dan Software

Hardware dan software yang sesuai untuk digunakan dalam membangun sistem arsitektur jaringan berbasis Software-Defined Networking ini akan sangat mempengaruhi kinerja sistem jaringan yang telah dirancang. Meskipun penelitian dilakukan secara virtual, kebutuhan hardware dibutuhkan sebagai saran jika penelitian ini ingin dilakukan secara riil. Dibawah ini merupakan perangkat keras dan perangkat lunak yang diperlukan, antara lain.

Tabel 2 Kebutuhan Perangkat Jaringan

No	Nama Perangkat	Spesifikasi
1	Router RB1100	13 buah port gigabit ethernet
		PowerPC MPC8544 network processor
		2 buah switch chip
		2 buah port auto bypass
		Casing 19" 1U rackmount
2	Router RB750	5 buah port ethernet 10/100
		Prosesor Atheros 400MHz
		Penggunaan SOHO
3	Cisco Catalyst 2950-24	24 Ethernet Ports 100BASE-T
		Managed - Command Line Interface
		Multicast management
4	Router CCR1036	12 port Gigabit ethernet, dan 4 SFP
		8GB RAM
		Tilera-Tile-Gx 36 Core CPU @ 1.2Ghz
5	Ubiquiti NanoStation M5 Antenna	2-Tx 2-Rx MIMO (Multiple Input Multiple Output)
		16 dBi
		32 MB SDRAM, 8 MB FLASH
		15 km max range
		2 LAN ports

		5470 - 5825 MHz Operating Frequency
		Throughput 150+ Mbps
6	TP-Link TL-WA901N Access Point	450 Mbps wireless transmission
		Range up to 30 meter
		Power over Ethernet Injector included
		IEEE 802.11n advanced MIMO

Untuk perangkat lunak (software) yang akan disebutkan pada tabel di bawah ini, merupakan perangkat lunak yang digunakan untuk menunjang konfigurasi dan implementasi virtual sesuai dengan tujuan penelitian ini.

Tabel 3 Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Keterangan
1	Sistem Operasi Komputer	Windows 11 Home
2	SDN Controller	Open Network Operating System
3	Software Simulasi	Cisco Packet Tracer
4	Software Desain Topologi	Microsoft Visio 2021
5	Software Implementasi Virtual	GNS3
6	Software Monitoring	Wireshark
7	Software Virtual Machine	Oracle VM VirtualBox

Spesifikasi perangkat komputer yang akan disebutkan pada tabel dibawah ini, merupakan perangkat komputer yang digunakan untuk menunjang jalannya kebutuhan spesifikasi yang direkomendasikan oleh seluruh perangkat lunak pada penelitian ini.

Tabel 4 Spesifikasi Perangkat Komputer

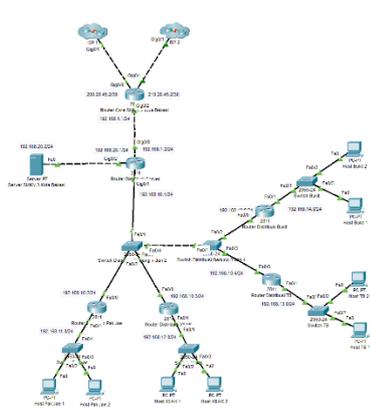
No	Spesifikasi	Keterangan
1	Prosesor	Intel Core i3 12100F
2	RAM	16 GB
3	Kartu Grafis	Radeon RX6500XT
4	Penyimpanan	740 GB

2.3. Tahapan Simulasi

Tahapan ini memiliki kegunaan untuk melakukan pengecekan terhadap kinerja atau performa awal dari desain yang telah dibuat dan akan dibangun berdasarkan dengan analisis pada subjek penelitian. Tahapan simulasi ini memiliki 2 bagian yaitu instalasi dan konfigurasi topologi secara umum, serta melakukan konfigurasi menggunakan controller. Simulasi dilakukan menggunakan aplikasi Cisco Packet Tracer berdasarkan pada konfigurasi umum Cisco untuk melakukan pengecekan terhadap ketersediaan koneksi yang akan dibangun sebelum dilakukan implementasi.

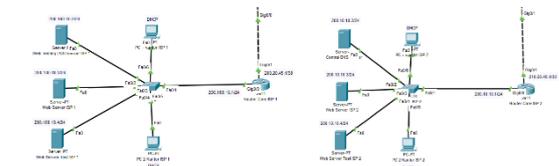
a. Instalasi dan konfigurasi umum

Pertama-tama yang harus dilakukan adalah melakukan pemasangan perangkat jaringan dan media transmisi yang diperlukan pada aplikasi Cisco Packet Tracer. Pemasangan jaringan dan media transmisi yang dilakukan pada Cisco Packet Tracer berdasarkan topologi dibawah ini.



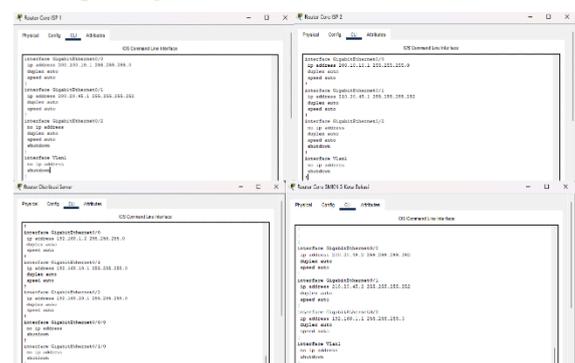
Gambar 7 Topologi Cisco Packet Tracer

Untuk topologi yang ada pada kedua ISP dapat dilihat pada gambar di bawah ini. Antara ISP 1 dan ISP 2 terdapat perbedaan pada server yang digunakan. Pada ISP 2 memiliki Central DNS Server untuk menampung keseluruhan dns yang ada pada jaringan.



Gambar 8 Topologi ISP 1 dan ISP 2

Setelah topologi dibuat, dilakukan konfigurasi pada masing-masing perangkat. Dibawah ini adalah konfigurasi yang terdaftar pada perangkat router yang saling terhubung. Pengecekan konfigurasi yang sedang berjalan dapat dilihat menggunakan command show running-config pada mode privileged EXEC command CLI.



Gambar 9 Interface Router

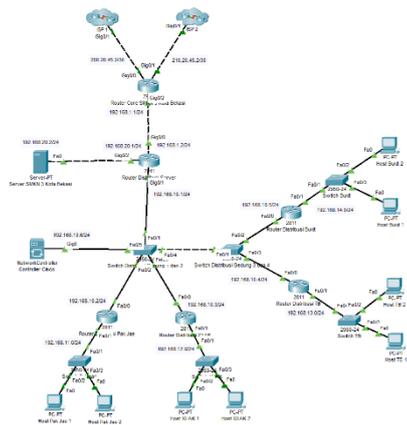
Konfigurasi yang dilakukan meliputi pengalamanan IP address, pengaturan dhcp client pada ISP dan host pada masing-masing router distribusi, dan proses routing menggunakan metode routing RIP menggunakan alamat ip setiap interface pada masing-masing router. Routing RIP yang digunakan adalah routing RIP versi 2 atau RIPv2.



Gambar 10 Routing Router

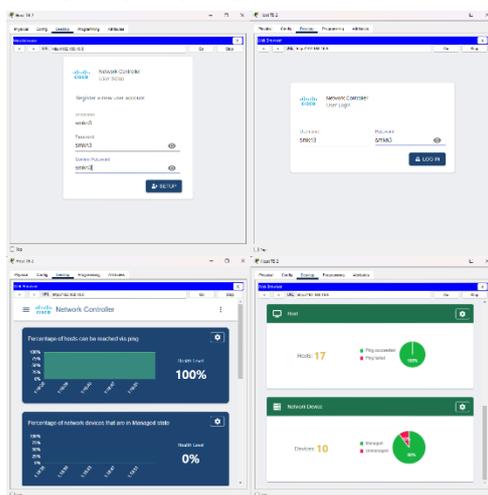
b. Konfigurasi controller

Simulasi controller ini dilakukan untuk memastikan konsep controller milik SDN dapat diimplementasikan pada topologi yang telah dibangun dan dikonfigurasi. Simulasi controller ini menggunakan fitur-fitur yang dimiliki oleh network controller milik Cisco. Ada penambahan perangkat untuk menggunakan fitur controller.



Gambar 11 Penambahan Network Controller

Untuk mengkonfigurasi controller, diperlukan pengalaman ip address, serta metode routing agar setiap perangkat dalam jaringan dapat berkomunikasi dengan perangkat network controller tersebut. Setelah dilakukan pengalaman serta perutean, controller dapat diakses melalui semua host yang ada pada jaringan.



Gambar 12 Network Controller

Secara umum, Network Controller pada Cisco Packet Tracer ini memiliki fungsi untuk mengumpulkan informasi mengenai jaringan. Informasinya meliputi host yang tersambung, QoS, topologi yang terbentuk, jalur lalu lintas data antar IP address, dan fungsi utamanya yaitu mengotomisasi security monitoring jaringan melalui API sesuai dengan scope dan policy yang tersedia.

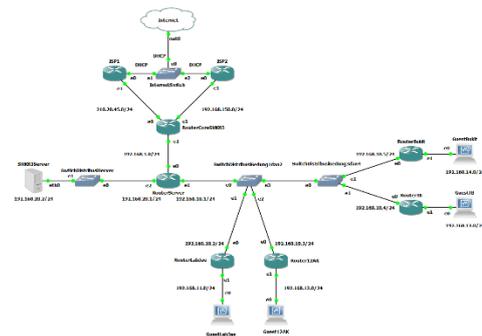
Simulasi yang telah dilakukan menunjukkan bahwa secara garis besar, konsep controller pada SDN dapat digabungkan dan diimplementasikan pada topologi yang telah dibuat berdasarkan pada pengalaman dan perutean jaringan, sehingga peneliti mendapatkan gambaran mengenai langkah-langkah yang akan dilakukan pada tahap implementasi.

2.4. Tahapan Implementasi

Secara konsep, tahapan implementasi dibuat berdasarkan pada tahap simulasi yang telah dilakukan. Namun, pengimplementasian dilakukan menggunakan perangkat router Mikrotik bukan perangkat router Cisco. Pengimplementasian dilakukan secara virtual menggunakan software GNS3 dan controller SDN ONOS atau Open Network Operating System. Ada beberapa langkah yang dilakukan, antara lain:

a. Konfigurasi Topologi Jaringan

Konfigurasi topologi jaringan diawali dengan membangun desain topologi sesuai dengan kebutuhan dan arahan pada tahap simulasi. Pada gambar dibawah ini, terdapat topologi yang telah dibuat pada software implementasi virtual, GNS3.



Gambar 13 Topologi pada GNS3

Setelah topologi dibuat, dilakukan konfigurasi jaringan seperti sebelumnya melalui CLI untuk mengkonfigurasi IP address, routing RIP, dan DHCP untuk menghubungkan seluruh host pada jaringan. Sebelum perangkat-perangkat jaringan dikonfigurasi, harus dilakukan penambahan perangkat jaringan dahulu pada aplikasi GNS3. Penambahan perangkat jaringan dilakukan menggunakan aplikasi tambahan Oracle VM VirtualBox sebagai aplikasi pihak ketiga yang menjadi tempat menjalankan mesin virtual perangkat jaringan, seperti Mikrotik sebagai router, Windows XP sebagai host, serta NAT sebagai internet untuk menghubungkan GNS3 kepada internet.

Setelah mesin-mesin virtual dibuat dan ditambahkan sebagai perangkat jaringan pada GNS3 serta topologi telah dibangun, konfigurasi IP address pada masing-masing router yang ada pada jaringan. Pada router ISP 1 dan ISP 2, lakukan konfigurasi dhcp-client untuk mendapatkan IP address secara DHCP dari internet.

```

[admin@ISP1] > ip add pr
Flags: X - disabled, I - invalid, D - dynamic
# ADDRESS NETWORK INTERFACE
# 210.20.45.1/24 210.20.45.0 ether2
# 192.168.122.70/24 192.168.122.0 ether1
[admin@ISP1] > ip route pr
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, P - rip, B - bgp, O - ospf, M - mme,
B - blackhole, U - unreachable, P - prohibit
# DST-ADDRESS PREF-SRC GATEWAY DISTANCE
# S 0.0.0.0 0 192.168.122.1 1
# A 0.0.0.0 0 0 0 0
1 A C 192.168.122.0/24 192.168.122.70 ether1 1
2 A C 210.20.45.0/24 210.20.45.1 ether2 0
[admin@ISP1] > ip dhcp-client pr
Flags: X - disabled, I - invalid, D - dynamic
# INTERFACE USE ADD-DEFAULT-ROUTE STATUS ADDRESS
# ether1 yes yes bound 192.168.122.70/24
[admin@ISP1] > ip firewall nat pr
Flags: X - disabled, I - invalid, D - dynamic
# action srcnat action masquerade out-interface=ether1
[admin@ISP1] >
    
```

Gambar 14 Konfigurasi IP Address – DHCP Client

Untuk pengalamatan source IP secara static, lakukan konfigurasi IP address, IP route gateway, IP DNS pada masing-masing router untuk setiap interface yang terpakai.

```

[admin@RouterCore001] > ip add pr
Flags: X - disabled, I - invalid, D - dynamic
# ADDRESS NETWORK INTERFACE
# 210.20.45.2/24 210.20.45.0 ether1
# 192.168.150.2/24 192.168.150.0 ether2
[admin@RouterCore001] > ip route pr
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, P - rip, B - bgp, O - ospf, M - mme,
B - blackhole, U - unreachable, P - prohibit
# DST-ADDRESS PREF-SRC GATEWAY DISTANCE
# S 0.0.0.0 0 192.168.150.1 1
# A 0.0.0.0 0 0 0 0
1 S 0.0.0.0 0 192.168.150.2 1
2 A C 192.168.150.0/24 192.168.150.2 ether2 0
3 A C 210.20.45.0/24 210.20.45.2 ether1 0
[admin@RouterCore001] > ip dns pr
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, P - rip, B - bgp, O - ospf, M - mme,
B - blackhole, U - unreachable, P - prohibit
# DST-ADDRESS PREF-SRC GATEWAY DISTANCE
# S 0.0.0.0 0 192.168.150.1 1
# A 0.0.0.0 0 0 0 0
1 S 0.0.0.0 0 192.168.150.2 1
2 A C 192.168.150.0/24 192.168.150.2 ether2 0
3 A C 210.20.45.0/24 210.20.45.2 ether1 0
[admin@RouterCore001] > ip dns sr
server: 192.168.122.1
dynamic-servers:
  non-doh-server:
    verify-doh-cert: no
  allow-remote-requests: no
  max-udp-packet-size: 4096
  query-server-timeout: 2s
  query-total-timeout: 10s
  max-concurrent-queries: 100
  max-concurrent-tcp-sessions: 20
  cache-size: 2048KB
  cache-max-ttl: 1w
  cache-used: 10KB
[admin@RouterCore001] >
    
```

Gambar 15 Konfigurasi IP Address – Static

Setelah masing-masing router telah dikonfigurasi IP source addressnya secara static, maka setelahnya konfigurasi IP destination addressnya. Terdapat 2 cara yang dilakukan untuk mengkonfigurasi IP destination address, tergantung dari jenis perangkat yang dihubungkan. Jika perangkat yang dihubungkan adalah perangkat jaringan seperti router, maka lakukanlah konfigurasi secara static untuk mencegah IP address yang berubah-ubah. Namun jika perangkat yang dihubungkan adalah perangkat host seperti komputer, maka dapat dilakukan konfigurasi secara DHCP menggunakan command ip dhcp-server setup pada terminal Mikrotik untuk memberikan koneksi otomatis DHCP pada host.

```

[admin@router1] > ip add -static
Flags: X - disabled, I - invalid, D - dynamic
# ADDRESS NETWORK INTERFACE
# 192.168.10.2/24 192.168.10.0 interface-ether1 actual-interface=ether1
# 192.168.10.1/24 192.168.10.0 interface-ether2 actual-interface=ether2
[admin@router1] > ip dhcp-server setup
Flags: D - dynamic, X - disabled, I - invalid
# name "dhcp"
# interface=ether1
# use-peer-dns=yes
# use-peer-dns-timeout=30s
[admin@router1] > ip firewall nat
Flags: X - disabled, I - invalid, D - dynamic
# action srcnat action masquerade out-interface=ether1
[admin@router1] >
    
```

Gambar 16 Konfigurasi DHCP Server

Jika saat melakukan pengetesan ping google.com sudah berjalan, maka router yang dikonfigurasi telah terhubung ke internet.

```

[admin@router1] > ping google.com
  SIZE TTL TIME STATUS
SEQ HOST
0 142.251.10.138 56 102 35ms
1 142.251.10.138 56 102 37ms
2 142.251.10.138 56 102 39ms
3 142.251.10.138 56 102 37ms
4 142.251.10.138 56 102 43ms
sent 5 received 5 packet-loss=0% min-rtt=35ms avg-rtt=38ms max-rtt=43ms
[admin@router1] >
    
```

Gambar 17 Pengetesan Ping

Setelah itu lakukan perutean menggunakan routing RIP agar setiap interface dapat saling berkomunikasi.

```

[admin@router1] > routing rip interface add network=ether1
[admin@router1] > routing rip network add network=192.168.10.0/24
[admin@router1] > routing rip network add network=192.168.13.0/24
[admin@router1] >
    
```

Gambar 18 Routing RIP

Setelah semua host sudah dapat berkomunikasi melalui ping, penelitian akan dilanjutkan untuk menambahkan serta mengkonfigurasi controller ONOS untuk melakukan manajemen terhadap perangkat jaringan di dalam wilayah SMKN 3 Kota Bekasi.

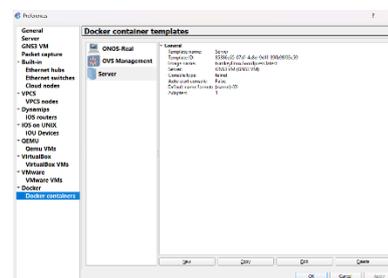
```

PC1> ping 192.168.13.150
84 bytes from 192.168.13.150 icmp_seq=1 ttl=62 time=5.761 ms
84 bytes from 192.168.13.150 icmp_seq=2 ttl=62 time=20.179 ms
84 bytes from 192.168.13.150 icmp_seq=3 ttl=62 time=7.938 ms
84 bytes from 192.168.13.150 icmp_seq=4 ttl=62 time=4.207 ms
84 bytes from 192.168.13.150 icmp_seq=5 ttl=62 time=5.524 ms
PC1>
PC2> ping 192.168.11.150
84 bytes from 192.168.11.150 icmp_seq=1 ttl=62 time=9.473 ms
84 bytes from 192.168.11.150 icmp_seq=2 ttl=62 time=3.807 ms
84 bytes from 192.168.11.150 icmp_seq=3 ttl=62 time=3.504 ms
84 bytes from 192.168.11.150 icmp_seq=4 ttl=62 time=3.448 ms
84 bytes from 192.168.11.150 icmp_seq=5 ttl=62 time=2.839 ms
PC2>
    
```

Gambar 19 Pengetesan Ping Routing

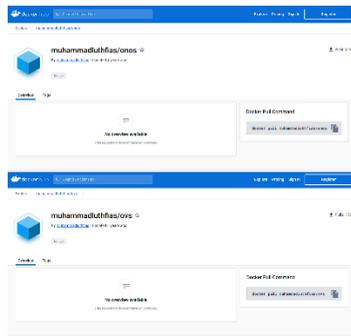
b. Konfigurasi Controller ONOS

Dalam pengkonfigurasi controller ONOS pada platform GNS3, hal pertama yang harus dilakukan adalah menambahkan perangkat ONOS pada GNS3. Penambahan appliance ONOS pada GNS3 dilakukan dengan metode docker.

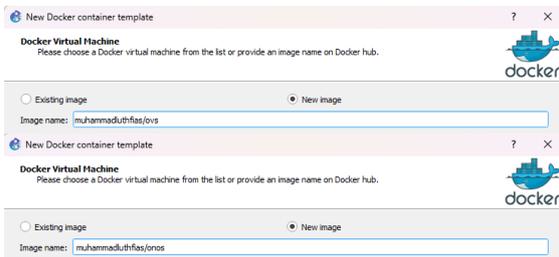


Gambar 20 Pemilihan Docker ONOS dan OVS

Perangkat ONOS dan OpenVSwitch (OVS) ditambahkan menggunakan docker container yang diambil dari container milik user-user di internet. Docker container yang peneliti gunakan adalah container yang ada diambil dengan command muhammadluthfias/onos untuk mengambil image ONOS dan muhammadluthfias/ovs untuk mengambil image OVS.

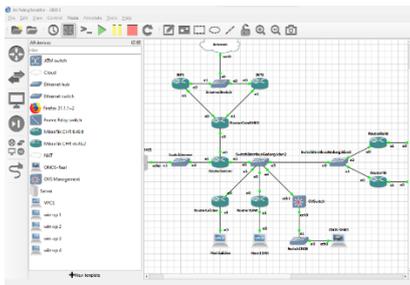


Gambar 21 Docker Container

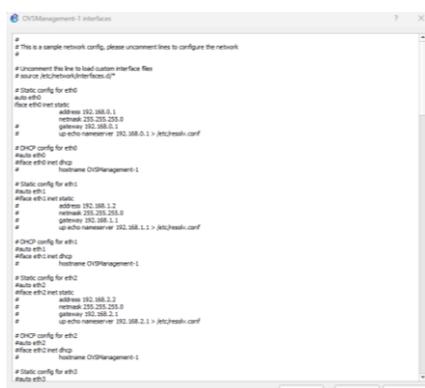


Gambar 22 Insert Pull Command

Setelah itu, lakukan penambahan ONOS dan OpenVSwitch pada topologi yang telah dibangun. Setelah ditambahkan, lakukan pengalamanan IP address pada ONOS dan OpenVSwitch. IP address yang digunakan berada dalam range ip 192.168.0.0/24.



Gambar 23 Insert ONOS dan OVS



Gambar 24 Konfigurasi IP pada OVS



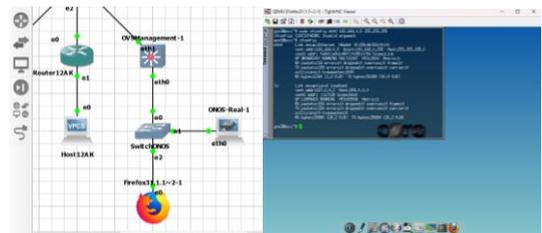
Gambar 25 Konfigurasi IP pada ONOS

Setelah IP address telah dikonfigurasi pada OVS dan ONOS. Lakukan konfigurasi pada console OVS. Konfigurasi yang dilakukan ini adalah untuk memastikan salah satu jalur ethernet pada OVS dikhususnya menjadi milik ONOS, dengan command.

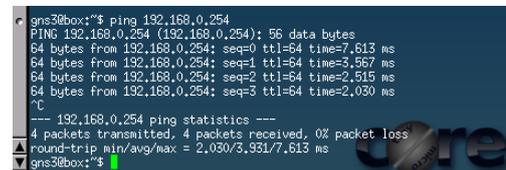
```
# ovs-vsctl del-port br0 eth0
# ovs-vsctl set-controller br0 tcp:192.168.0.254:6633
# ovs-vsctl show
```

Gambar 26 Command pada OVS

Setelah itu tambahkan, host client untuk mengakses web ONOS. Peneliti menggunakan mesin bawaan GNS3 untuk Firefox. Lalu samakan IP pada mesin Firefox ke dalam range IP ONOS dengan menggunakan command sudo ifconfig eth0 192.168.0.5 255.255.255.0.



Gambar 27 Penambahan dan Pengalamanan Firefox



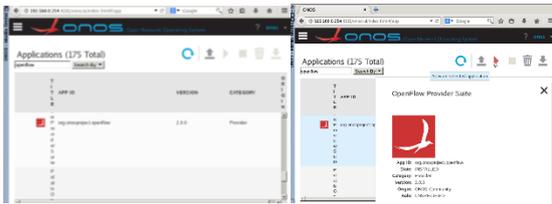
Gambar 28 Pengetesan Ping Firefox

Setelah pengetesan ping berhasil, masuk ke dalam browser Firefox. Lalu pada search bar, ketikkan 192.168.0.254:8181/onos/ui/index.html untuk masuk ke dalam tampilan web dari ONOS. 192.165.0.254 merupakan IP static milik ONOS. Setelah itu akan muncul tampilan login ONOS. Setelah itu masukkan username dan password default milik ONOS, yaitu "onos" untuk username, dan "rocks" untuk password.



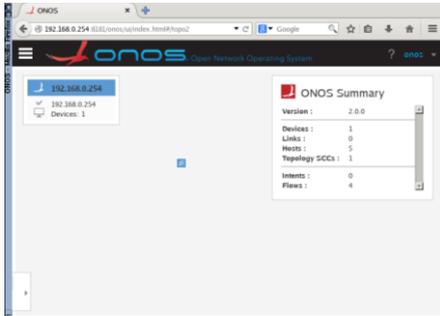
Gambar 29 Login dan Dashboard ONOS

Pada gambar diatas, tampilan dashboard ONOS belum terkonfigurasi. Untuk mengkonfigurasi tampilan, aplikasi OpenFlow harus diaktifkan terlebih dahulu.

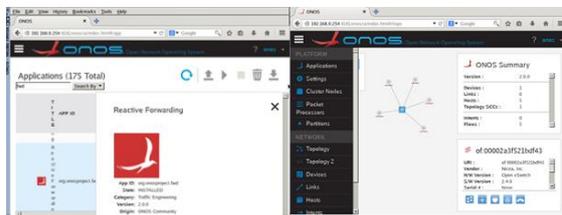


Gambar 30 Aktivasi OpenFlow

Setelah OpenFlow diaktivasi, maka topologi akan terlihat. Namun topologi yang terlihat hanyalah node dari OpenVSwitch saja. Untuk mengatasi hal itu, harus mengaktifkan aplikasi ONOS lainnya, yaitu Forwarding.

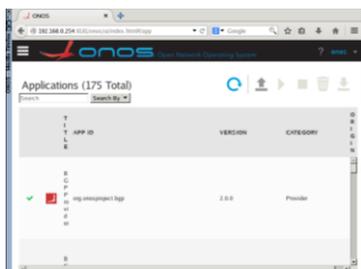


Gambar 31 Node OpenVSwitch



Gambar 32 Aktivasi Forwarding

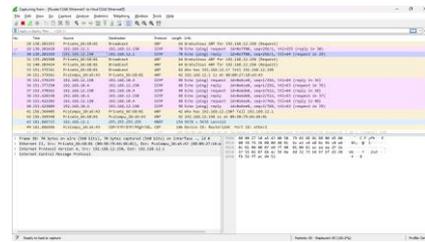
Terlihat koneksi dari *node-node* yang terhubung ke dalam jaringan *ONOS*. *Node-node* tersebut dapat dikonfigurasi menggunakan *API* yang sudah disediakan oleh komunitas *ONOS*. Saat ini, sudah ada total 175 aplikasi yang dapat diterapkan ke dalam jaringan yang terintegrasi dengan *ONOS*.



Gambar 33 API pada ONOS

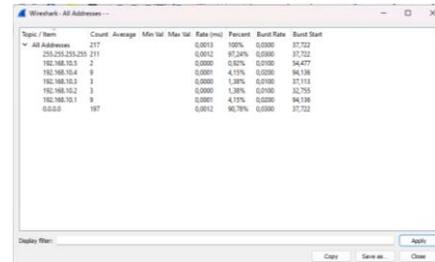
2.5. Tahap Monitoring

Monitoring jaringan dilakukan menggunakan aplikasi Wireshark yang telah terintegrasi ke dalam GNS3. Wireshark akan menangkap aliran traffic data yang sedang dimonitor. Pada aliran milik Host12AK, terlihat adanya pergerakan komunikasi antar host.



Gambar 34 Monitoring Wireshark

Pada jalur antara RouterServer dan SwitchDistribusiGedung1dan2 menggunakan Wireshark, terlihat IP address yang sedang terkoneksi pada jalur tersebut.



Gambar 35 IP Address – Wireshark

3. KESIMPULAN

Implementasi konsep Software-Defined Networking dengan metode Network Development Life Cycle (NDLC) menggunakan controller Open Network Operating System (ONOS) pada SMK Negeri 3 Kota Bekasi dapat ditemukan beberapa kesimpulan, diantaranya.

- a. Telah berhasil dilakukan pembangunan serta pengimplementasian jaringan secara virtual pada platform GNS3 berbasis Software-Defined Networking dengan controller ONOS serta melakukan beberapa konfigurasi tambahan ke dalam jaringan menggunakan ONOS API sesuai dengan analisis, simulasi, pengalamatan, serta perutean yang telah didapatkan.
- b. Telah berhasil melakukan monitoring interface serta monitoring paket data untuk mengetahui jaringan yang sedang terkoneksi dan berkomunikasi dengan jaringan lain.
- c. Telah berhasil melakukan penyederhanaan jaringan sebagai kebutuhan jaringan SMK Negeri 3 Kota Bekasi

Disarankan agar dapat mempertimbangkan untuk mengembangkan jaringan SDN ONOS ini, dengan beberapa pengembangan diantaranya.

- a. Mengembangkan kembali implementasi ini, agar dapat diimplementasikan secara real menggunakan perangkat-perangkat jaringan yang telah dijelaskan. Juga mengembangkan konsep menggunakan antenna ubiquiti untuk menghubungkan koneksi antar gedung.
- b. Mengujicoba kembali controller ONOS menggunakan Mikrotik dengan menghubungkannya pada OpenFlow.

c. Gunakan OpenVSwitch untuk memudahkan implementasi ONOS, karena sudah terintegrasi dengan OpenFlow, dibandingkan dengan Mikrotik yang harus dikonfigurasi lebih detail agar dapat terhubung pada ONOS.

PUSTAKA

- Amalia, R., Kalsum, T. U., & Riska. (2021). Analisis dan Implementasi Software Defined Networking (SDN) untuk Automasi Perangkat Jaringan. *Infotek : Jurnal Informatika dan Teknologi*, 312-322. <https://doi.org/10.29408/jit.v4i2.3734>
- Dewannanta, D. (2013). Mengenal Software Simulator Jaringan Komputer GNS3. <https://ilmukomputer.org/wp-content/uploads/2013/01/gns3.pdf>
- Harja, D. P., Rakhmatsyah, A., & Nugroho, M. A. (2019). Implementasi Metode Deep Packet Inspection untuk Meningkatkan Keamanan Jaringan pada Software Defined Networks. *Ind. Journal on Computing*, 133-146. <https://doi.org/10.21108/INDOJC.2019.4.1.286>
- Hellyana, C. M. (2018). MODUL PERKULIAHAN JARINGAN KOMPUTER UNIVERSITAS BINA SARANA INFORMATIKA. <https://repository.bsi.ac.id/repo/files/237077/download/Modul-JARKOM-2018.pdf>
- Hidayat, R. P., Primananda, R., & Widasari, E. R. (2018). Analisis Performa Centralized Firewall pada Multi Domain Controller di Arsitektur Software-Defined Networking (SDN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2698-2705. <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/1674>
- Irianti, W. (2018). RANCANGAN SISTEM JARINGAN LAN (LOKAL AREA NETWORK) DI SATUAN KERJA STAF OPERASI MABESAU. *Jurnal Online Mahasiswa Sistem Informasi dan Manajemen Informatika*, 155-165. <https://jom.universitassuryadarma.ac.id/index.php/simika/article/view/20/21>
- Mahmud, P. T., Araf, M. T., Destianti, L. L., Cahyani, I., Cantika, I., Huda, F. N., . . . Maharani, A. (2019). SNIFFING JARINGAN MENGGUNAKAN WIRESHARK. *JURNAL JARINGAN KOMPUTER*. <https://mfr.osf.io/export?format=pdf&url=https%3A/files.osf.io/v1/resources/h5wu7/providers/osfstorage/5e7f3fabd0e354001e44ec0e%3Fformat%3Dpdf%26action%3Ddownload%26direct%26version%3D1>
- Maulana, A. R., Walidainy, H., Irhamsyah, M., Fathurrahman, & Akhyar. (2021). Analisis Quality of Service (QoS) Jaringan Internet Pada Website e-Learning Universitas Syiah Kuala Berbasis Wireshark. *KITEKTRO: Jurnal Komputer, Teknologi Informasi, dan Elektro*, 27-30. <https://doi.org/10.24815/kitektro.v6i2.22284>
- Melissa. (2018). Analisis Kinerja Arsitektur Software Defined Network Berbasis Opendaylight Controller. <http://elibrary.unikom.ac.id/id/eprint/587>
- Micro, A. (2012). *Dasar-dasar Jaringan Komputer*.
- Napitu, D. H. (2017). Karakteristik Openflow Controller dengan ONOS. *Jurnal Teknik Informatika Kaputama (JTIK)*, 10-14. <https://jurnal.kaputama.ac.id/index.php/JTIK/article/view/36/31>
- Narayana, S. (2019). Control-Data Plane Separation. 552, *Fall 2019: Computer Networks*. Rutgers University. <https://people.cs.rutgers.edu/~sn624/552-F19/lectures/>
- Pamungkas, D. P., Setiawan, A. B., & Ramadhani, R. A. (2018). *Jaringan Komputer Dasar*. Jombang: CV. Kasih Inovasi Teknologi.
- Pratama, I. A., & Wikantya, I. A. (2019). Implementasi dan Analisis Simulasi QOS dan Performance Device dengan Menggunakan ONOS dan Iperf3. *Jurnal Informatika Universitas Pamulang*, 57-64. <https://dx.doi.org/10.32493/informatika.v4i2.2730>
- Prayoga, D., Ijtihadie, R. M., & Husni, M. (2017). Implementasi POX pada Perangkat Lunak Software-Defined Networking Controller untuk Data Center Berbasis Container. *JURNAL TEKNIK ITS*, A352-A355. <http://dx.doi.org/10.12962/j23373539.v6i2.23448>
- Rohaya, S. (2008). INTERNET: PENGERTIAN, SEJARAH, FASILITAS DAN KONEKSINYA.
- Saputra, H., & Dalimunthe, R. A. (2021). IMPLEMENTASI FIREWALL PADA DATA LINK LAYER MENGGUNAKAN ARSITEKTUR SOFTWARE DEFINED NETWORK. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, 245 - 250. <https://doi.org/10.33330/jurteks.v7i3.1199>

- Saputra, Y. (2021). Analisis Performansi Software Defined Networking (SDN) Controller Floodlight, RYU, dan ODL Pada Topologi Jaringan Universitas Islam Riau. <http://repository.uir.ac.id/id/eprint/8901>
- Simarmata, R. F., Tulloh, R., & Haryani, Y. S. (2018). SIMULASI JARINGAN SOFTWARE DEFINED NETWORK MENGGUNAKAN PROTOKOL ROUTING OSPF DAN RYU CONTROLLER. *e-Proceeding of Applied Science*, 2887-2896. <https://openlibrarypublications.telkomuniversity.ac.id/index.php/appliedscience/article/view/7542/7430>
- Sudiby, N. A. (2021). Materi 4 DISTRIBUSI DATA.
- Yasin, A. (2020). SIMULASI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) DI SOFTWARE-DEFINED NETWORK DI SIMULASI JARINGAN MININET DAN GNS3. *JTI*, 32-41. <https://doi.org/10.30869/jtii.v5i1.927>
- Yasmin, A., & Sari, L. O. (2019). Simulasi Software Defined Network (SDN) untuk Manajemen Jaringan Data PT. Chevron Pasifik Indonesia. *Jom FTEKNIK*. <https://jom.unri.ac.id/index.php/JOMFTEKNIK/article/view/23320/22579>
- Zsa-zsa, A. (2019). REDESIGN INFRASTRUKTUR DAN MANAJEMEN JARINGAN DENGAN METODE NETWORK DEVELOPMENT LIFE CYCLE (NDLC) DI KANTOR POS REGIONAL III PALEMBANG. <http://repository.binadarma.ac.id/id/eprint/182>