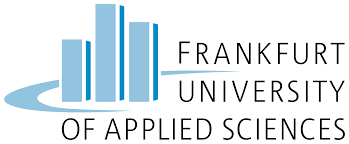
​*Fachhochschule Frankfurt – University of Applied Sciences   
Fachbereich 2: Informatik und Ingenieurwissenschaften   
Studiengang: Informatik (B.S)*

*Praxisprojekt Wintersemester 2020*

**Betreuer: Prof. Dr. Maurizio Petrozziello**



**Projektthema:**

***Software-defined Networking mit Openflow***

**Name** **Matrikelnummer**

Tung Thanh Trinh 1320718

James Belmonte 1340604

Naghmeh Ghavidel Rostami 1249307

Mücahit Sagiroglu 1228852

Hiermit erklären wir an Eides statt, dass ich den vorliegenden Projektbericht bis auf die offizielle Betreuung selbst und ohne fremde Hilfe angefertigt habe und die benutzten Quellen und Hilfsmittel vollständig angegeben sind.

Frankfurt, 27.01.2022

Tung Thanh Trinh

James Belmonte

Naghmeh Ghavidel Rostami

Mücahit Sagiroglu

1. Einleitung
2. Projekt
   1. Projektziel
   2. Projektumfeld
   3. Vorgehen
3. Projektplannung
   1. Aufbau des Projektplanes
   2. Festlegen von Meilensteinen
4. Durchführung des Projektes
5. Gesamtergebnis
6. Resümee und Ausbilck

**1 Einleitung**

**2 Projekt**

**2.1 Projektziel**

**2.2 Projektumfeld**

**2.3 Vorgehen**

**3 Projektplannung**

**3.1 Aufbau Projektplan**

**3.2 Festlegen von Meilensteine**

**4 Durchführung des Projektes im Virtual Maschine mit Mininet**

Die Durchführung des Projektes habe ich in den unteren Bereichen unterteilt:

* Verwendete Werkzeuge
* Aufbau Netzwerkplan
* Verschlüsselung durch VPN Tunnel

**4.1 Verwendete Werkzeuge**

**4.1.1 Mininet**

**4.1.1.1 Einführung**

Mininet ist ein Netzwerk Emulator mit der man Netzwerke simulieren kann. Bei Mininet handelt es sich um kostenlose Open-Source-Software, die die virtuelle Maschine und Controller die Recherche in SDN und OpenFlow ermöglichen.

Mininets ermöglichen eine sehr groß angelegte Topologie, wodurch ein Netzwerk von Hosts, Switch-, Controller- und virtuelle Links erstellt wird. Das Ausführen von Tests mit den Komponenten ist unkompliziert und kann über Python Schnittstelle erledigt werden.

**4.1.1.2 Funktionalität**

* Mininet stellt ein einfaches Netzwerk Testbed dar, welches aber auch gleichzeitig auch günstig ist. Da der Switch Openflow in Mininet alle Eigenschaften hat, wie ein echter switch OpenFlow, ist die Anwendung von Netzwerkemulator mit Mininet praktisch sinnvoll.
* Ermöglicht das Debuggen und Ausführen von Tests großer Netzwerke mithilfe von CLI.
* Unterstützung zum Einrichten beliebiger benutzerdefinierter Diagramme
* Die Anwendungen im Mininet können im echten Netzwerk realisiert werden, ohne dass man sein Code ändern muss.
* Mininet bieten eine benutzerfreundliche und erweiterbare Python-API

**4.1.1.3 Nachteil**

Aktuelle Nachteile von Mininet: nur unter Linux lauffähig. Dadurch, dass Mininet nur auf einem Rechner ausgeführt werden kann, ist es leistungsmäßig eingeschränkt. Daher hängt die Leistung von den Ressourcen dieses Rechners ab.

* + - 1. **Komponenten**
* **Links**: Die Links in einem Mininet sind ein Paar virtueller Ethernets, die wie ein Draht funktionieren, der zwei virtuelle Schnittstellen verbindet. Pakete werden von einer Schnittstelle zur anderen gesendet, diese Schnittstellen stellen für alle genau dasselbe wie Ethernet-Ports dar System- und Anwendungssoftware.
* **Hosts:** Die Netzwerk-Namespaces enthalten den Netzwerkstatus (network state). Sie bieten Prozessen (oder Gruppen von Prozessen) die Kontrolle über Schnittstellen, Ports und Routing-Tabellen. Jeder Host hat seine eigenen Ethernet-Schnittstellen (initiiert und gesetzt durch den Befehl ip link add/set) und eine Verbindung zum Mininet.
* **Switches**: OpenFlow-Softswitches bieten die gleiche Semantik für das Senden

Paket wie ein Hardware-Switch. Ein Switch arbeitet auf der Datenverbindungsschicht (Layer 2) und manchmal auf der Netzwerkschicht (Layer 3) des OSI (Open Systems Interconnection) -Referenzmodells und unterstützt daher jedes beliebige Paketprotokoll. LANs, die zur Verbindung von Segmenten Switches verwenden, werden als geswitchte LANs oder, im Falle von Ethernet-Netzwerken, als geswitchte Ethernet-LANs bezeichnet. In Netzwerken ist der Switch das Gerät, das Pakete zwischen LAN-Segmenten filtert und weiterleitet

* **Controller:** Controller können sich überall im realen Netzwerk oder in der Umgebung befinden Simulation.
* **Router:** Router ist mit mindestens zwei Netzwerken verbunden, in der Regel zwei LANs oder WANs (Wide Area Networks) oder ein LAN und sein ISP (Internet Service Provider) Netzwerk. Der Router befindet sich in der Regel an Gateways, den Stellen, an denen zwei oder mehr Netzwerke miteinander verbunden sind. Anhand von Headern und Weiterleitungstabellen bestimmt der Router den besten Weg zur Weiterleitung der Pakete. Darüber hinaus verwendet der Router Protokolle wie ICMP (Internet Control Message Protocol), um miteinander zu kommunizieren, und konfiguriert die beste Route zwischen zwei beliebigen Hosts. Mit einem Wort: Der Router leitet Datenpakete weiter.

**4.1.1.5 Installation**

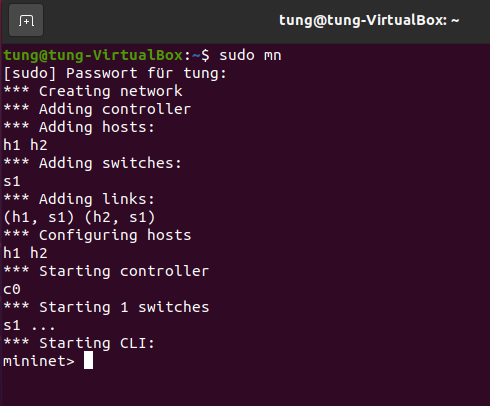
**$sudo apt-get install git**

**$git clone git://github.com/mininet/mininet**

**$sudo mininet/util/install.sh –a**

**4.1.1.6 Aufbau**

Durch die Eingabe von dem Befehl sudo mn erfolgt ein Default Mininet. Auf der virtuellen Maschine laufen 4 Entitäten (2 Hosts, 1 Switch, 1 Controller).

****

**4.1.2 Floodlight Controller**

Derzeit gibt es auf dem Markt einige Controller, die in SDN verwendet werden, wie zum Beispiel: OpenDaylight,Ryu, POX, NOX, HP VENTIL... Wir haben uns für Floodlight Controller entschieden, weil er die Anforderungen des Projekts erfüllt. Floodlight Controller ist sehr benutzerfreundlich und bietet sogar eine Benutzeroberfläche.