Chapitre X: Software Defined Network

Par Ilyas Bambrik

Problématique:

- Avec l'évolution des applications et la perspective de futures services, le modèle TCP/IP est devenu incapable de répondre au besoins de celles-ci.
- Dans un réseaux TCP/IP, le changement de politiques nécessite la reconfiguration de chaque équipement (Switch, Routeur) manuellement. Dans un réseau grand échelle ceci est :
  - Difficile à accomplir;
  - Couteux en temps et en argent;
  - Peut introduire des erreurs de configuration;
- Cette inflexibilité du réseau TCP/IP a poussé les chercheurs à trouver un nouvel paradigme répondant à ce problème. Ce paradigme, s'appel ***Software Defined Network (SDN)***.

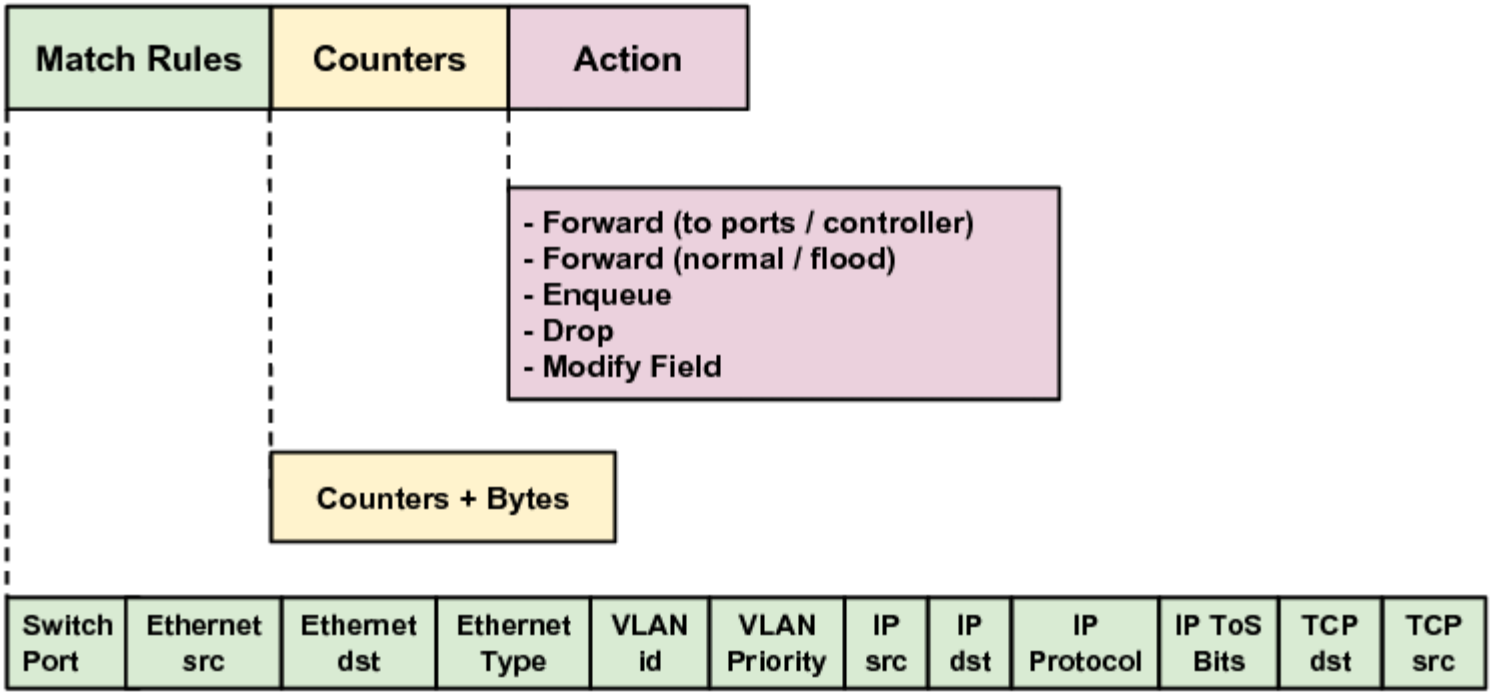
Principe SDN

- L'idée de cette technologie est de permettre la programmation du réseau à partir d'une instance centrale qui gouverne tout le réseau. Cette instance est appelée le **Contrôleur**.
- Par ailleur, l'intelligence des Switchs / Routeurs est exportée de ces derniers au contrôleur. Par intelligence, on désigne les algorithmes (RIP, OSPF, etc) permettant de créer un Source Address Table (SAT) du switch et la table de routage au niveau du routeur.
- Les tables de routage et SATs sont remplacées par **Flow Table (Table des Flux)**.
- ***L'intelligence des équipements est migrée au Contrôleur.***

Table de Flux (Flow Table)

- Chaque entrée de la Table de Flux est répartie en trois sections:
  - **Matching Rules (aussi appelées Header Fields):** joue le rôle des adresses IP pour les routeurs/ MAC pour les switchs. Par contre, cette section **comporte plusieurs valeurs de champs** (Numéro de port switch, Numéro de port de la couche transport, Protocole de transport, etc) contrairement à seulement l'adresse IP/MAC. Cette section permet de vérifier si un paquet correspond à la description d'un flux.
  - **Counters:** comporte des statistiques (nombre de paquets, bits, temps écoulé) pour le flux correspondant (celui dans l'entete correspond au Matching Rules).
  - **Action:** désigne un ou plusieurs actions à effectuer sur à chaque paquet qui correspond aux Matching Rules. L'action peut être de supprimer le paquet, de changer l'adresse IP destination / source, transmettre le paquet sur un port donné du switch, etc.
  - **Idle Timeout:** Temps après le quel l'entrée du flux sera supprimée de la table du flux **si aucun paquet n'est reçu.**
  - **Hard Timeout:** Temps après le quel l'entrée du flux sera supprimée;
- Chaque fois qu'un paquet est reçu par le Switch, les valeurs des entetes Liaison / IP / Transport sont comparées avec chaque Matching Rules afin d'identifier à quel flux ce paquet appartient. **Une fois qu'une entrée correspondante est trouvée dans la Table de Flux, les actions placées dans cette entrée sont appliquées**
- **Si aucune entrée ne correspond au paquet, le paquet est transmis au Contrôleur. Celui traite le paquet et renvoie une nouvelle entrée de la table du flux au Switch pour installer celle-ci dans ça Table des Flux.**

Structure de la Table de Flux

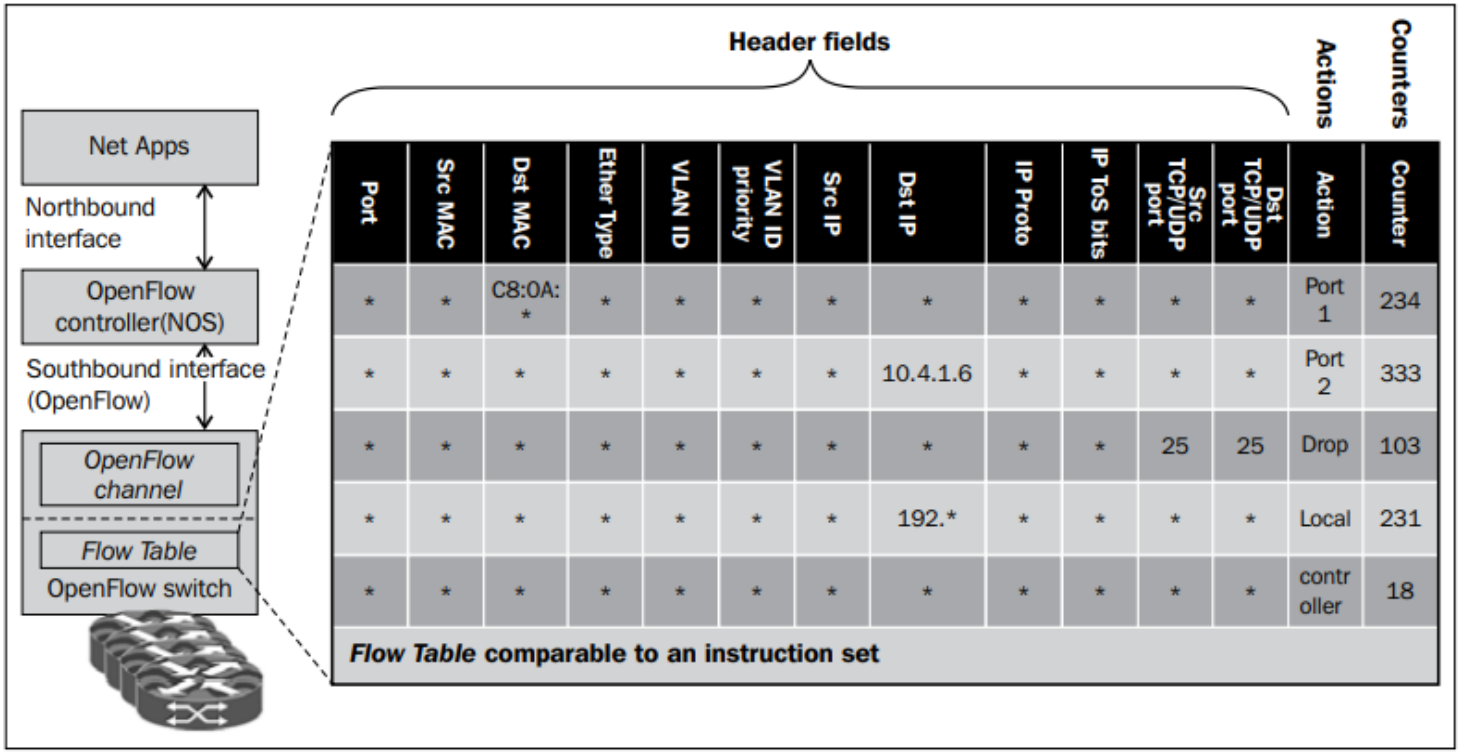


Actions supportées par les Equipements

- Les types d'action typiques sont:
- **FORWARD:** Retransmettre le paquet sur un port donné du Switch/Routeur. Plusieurs types de retransmissions existent:
  - **ALL:** Retransmettre sur tout les ports du Switch;
  - **CONTROLLER:** Encapsuler et retransmettre le paquet au Contrôleur;
  - **LOCAL:** Faire passer le paquet à la couche IP (fonctionnement TCP/IP normal);
- **DROP:** Supprimer le paquet;
- **ENQUEUE:** Mettre le paquet dans une file d'attente d'une interface donnée ;
- **MODIFY FEILD:** Permet de changer la valeur d'un champ des entete du paquet (destination source par exemple);
- **Si la liste des Actions correspondante à un flux ne contient aucune action FORWARD, le paquet est supprimé par défaut.**

Exemple de la Table de Flux

- Par exemple, la deuxième entrée de la table suivante indique que les paquets reçus avec une adresse destination 10.4.1.6 sont retransmis sur le port 2 du Switch. Les cellules avec une valeur * signifient que n'importe valeur sera acceptée (dite aussi ANY);
- **Remarque:** Une entrée appelée Table-Miss doit etre définie dans la Table des Flux afin définir le traitement des paquets qui ne correspondent à aucune entrée dans la table. Dans cet exemple, cette entrée correspond à la dernière entrée (toutes les valeurs == *, Action ==Controller)
- Un paquet peut satisfaire les règles de plusieurs entrées. Dans ce cas, les actions de l'entrée avec les valeurs de règles les plus spécifiques (minimum de valeurs ANY) sont appliquées;



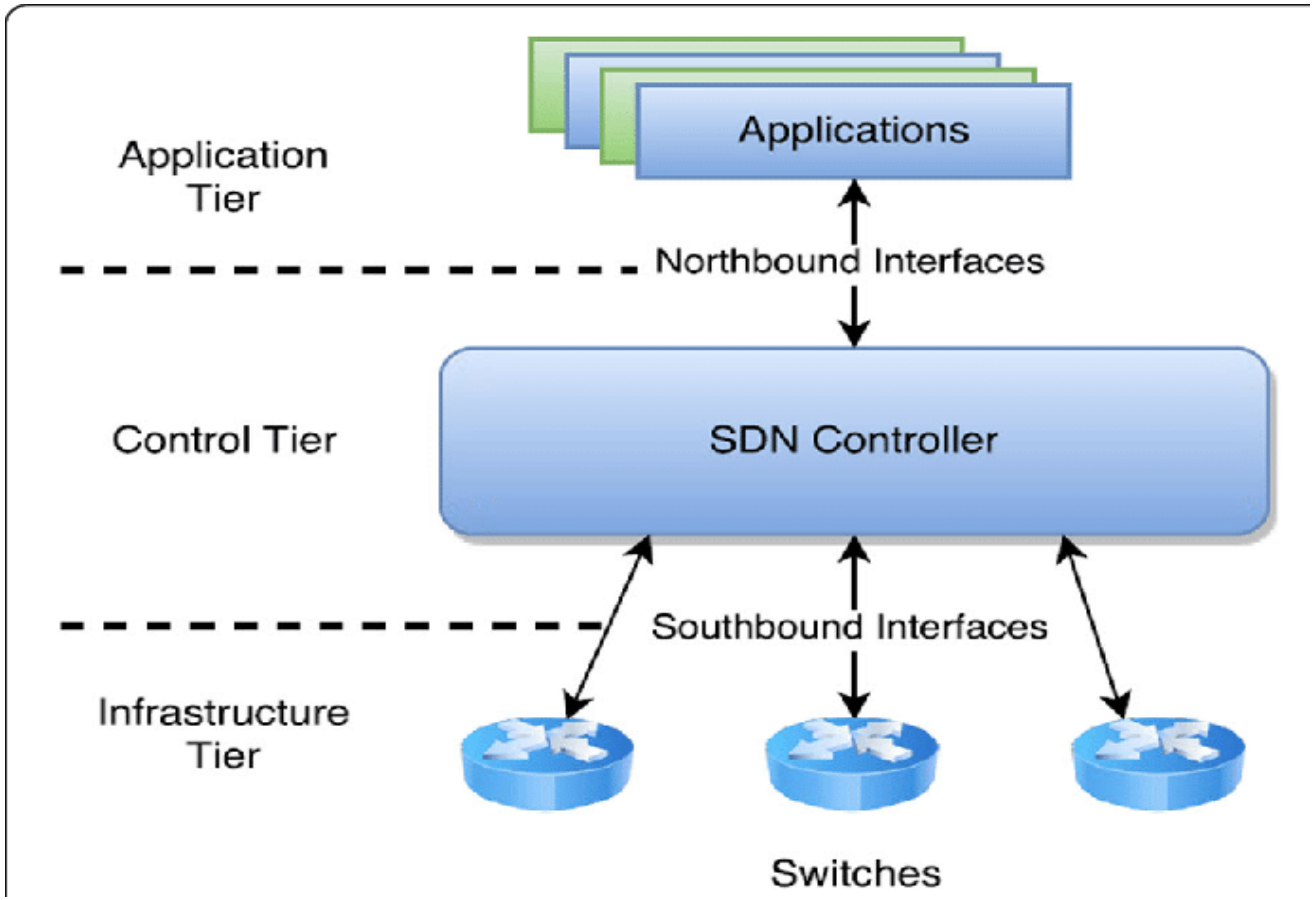
Header fields												Actions	Counters
Port	Src MAC	Dst MAC	Ether Type	VLAN ID	VLAN ID priority	Src IP	Dst IP	IP Proto	IP ToS bits	Src TCP/UDP port	Dst TCP/UDP port	Action	Counter
*	*	C8:0A:*	*	*	*	*	*	*	*	*	*	Port 1	234
*	*	*	*	*	*	*	10.4.1.6	*	*	*	*	Port 2	333
*	*	*	*	*	*	*	*	*	*	25	25	Drop	103
*	*	*	*	*	*	*	192.*	*	*	*	*	Local	231
*	*	*	*	*	*	*	*	*	*	*	*	controller	18

***Flow Table* comparable to an instruction set**

Contrôleur SDN

- Plusieurs Contrôleurs SDN existent actuellement (POX, NOS, Ryu, Open Floodlight). Un Contrôleurs est un logicielle qui contrôle les équipements réseau comme un Système d'exploitation;
- Selon le Contrôleur, celui-ci est programmable avec un langage de programmation moderne (Java, en Python ou C++). La majorité des Contrôleurs actuels et ceux les plus utilisés sont programmables en Python;
- En programmant le Contrôleur, l'utilisateur peut redéfinir la politique du réseau sans reconfigurer les équipements Switchs / Routeurs;
- Les applications peuvent interagir avec le contrôleur pour accéder au réseau à travers l'API proposée par le contrôleur. Cette API est appelée aussi Northbound Interface (Interface vers le Nord);

Architecture SDN



Protocole OpenFlow

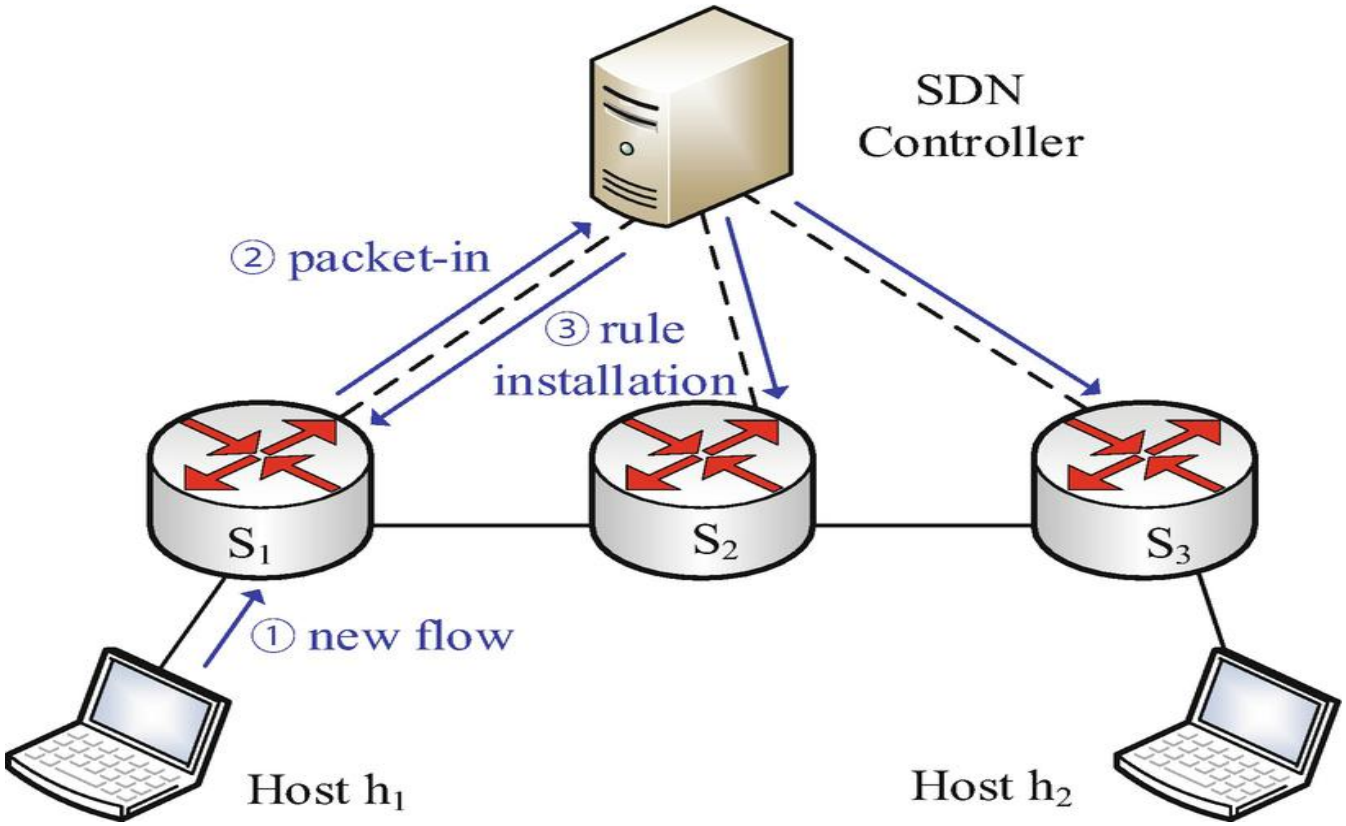
- La communication entre les Routeurs/Switchs SDN et le Contrôleur est faite grâce au protocole OpenFlow qui définit les types des messages échangés entre ces deux derniers;
- Les Routeurs/Switchs doivent êtres configurés avec un certificat d'authentification et l'adresse IP du Contrôleur ;
- Au lancement du Routeurs/Switchs SDN, celui-ci connecte au Contrôleur (par défaut le port TCP 6633 est utilisé par ce protocole). Les Routeurs/Switchs établissent une connexion sécurisée au Contrôleur avec le protocole TLS (Transport Layer Security);
- Les paquets reçus du Contrôleur auront une action correspondante **Local** dans la Table de Flux pour que les paquets reçus du Contrôleur soient traités par le protocole OpenFlow;
- **Remarque:** TLS n'est qu'un protocole qui fonctionne avec TCP et qui commence par l'échange des Certificats numériques (RSA/AES) des deux parties (Switch et Contrôleur) avant que les données ne soient envoyées;

Message OpenFlow

- Après l'établissement de la connexion, le Contrôleur est capable interroger un Switch ou de modifier sa configuration / Table de Flux avec les messages suivants:
  - **Features:** Le contrôleur peut transmettre un message **Feature Request** pour obtenir les capacités/fonctionnalités supportées du Switch. Ce dernier répond par un **Feature Reply;**
  - **Configuration:** Permet de changer ou d'obtenir la configuration d'un Switch;
  - **Modify-State:** Ce type de message permet au Contrôleur de manipuler la Table de Flux. Plusieurs sous-types existent dans ce type de message:
    - **ADD:** Permet d'ajouter une entrée à la Table de Flux. Si le flag **Check_Overlap** est allumé, le Switch vérifie si une entrée correspondante existe et si c'est le cas, l'entrée n'est pas ajoutée.
    - **Modify:** Permet de modifier une entrée de la Table de Flux. Si aucune entrée correspond au message Modify, une entrée est créé dans la Table de Flux.
    - **Delete:** Permet de supprimer l'entrée de flux correspondante.
  - **Read State:** Permet de demander les statistiques du Switch;
  - **Send-Packet:** Demande au Switch de transmettre le paquet sur un port donné;

Création d'une entrée de flux par le Contrôleur

1. H1 transmet un paquet pour le Switch S1 ne possède aucune entrée (1);
2. S1 transmet un Packet-In au Contrôleur SDN (2);
3. Le Contrôleur SDN transmet un message **Modify-State** pour installer le nouvel flux dans S1 (3);



Messages Symétriques

- Les messages symétriques peuvent être initiés par le Contrôleur ou le Switch:

  - **HELLO:** Ce type de message est transmis lors de l'initiation de la connexion;
  - **ECHO:** Ce type de message est utilisé comme les message Echo request / reply ICMP pour vérifier si la destination est enligne. Le Switch teste la connexion avec le Contrôleur par ce mécanisme.

- En cas où la connexion est rompue entre le Switch / Contrôleur (le Contrôleur ne répond pas au Echo Request du Switch), le Switch commence à utiliser une Table de Flux alternative appelée **Emergency Table et supprime les entrées installées par le Contrôleur avant la coupure (ce fonctionnement est appelée Mode d'Urgence)**. Le Switch continue à tenter de connecter au Contrôleur et s'il succède, le Switch bascule au mode de fonctionnement normal.

Messages Asynchrones

- Ce type de message est initié lorsqu'un événement réseau est détecté. Les messages suivants tombent dans cette catégorie:
  - **Packet-In:** Lorsqu'un paquet est reçu par un Switch qui ne correspond à aucune entrée de la Table de Flux ou bien l'action correspondante est égale à **CONTROLLER**, un **Packet-In** est généré et transmis vers le Contrôleur. Avant de transmettre le Packet-In au Contrôleur, le paquet ***initial (celui pour le quel aucune entrée de la Table de Flux ne corresponde)*** est placé dans le buffer et un identifiant est affecté à celui-ci. Le **Packet-In** transmis au Contrôleur contiendra l'identifiant buffer et une copie de l'entete du paquet. Lorsque le Contrôleur renvoie une réponse **Send-Packet**, celui-ci comprendra l'identifiant buffer.
  - **Flow-Removal:** Si une entrée de la Table de Flux est supprimée à cause du Idle / Hard Timeout, un message **Flow-Removal** est transmis au Contrôleur;

Conclusion

- Avec ces messages basiques d'OpenFlow le Contrôleur peut gérer la QoS, la consommation d'énergie ainsi que la sécurité.
- Le désavantage de l'architecture SDN est le fait que toute la topologie est géré par une seule instance (le Contrôleur). Plusieurs Contrôleur peuvent coexister dans le même réseau afin de faire face à ce problème.