# VLAN

# LAN “tradizionale” [1]

- L’organizzazione delle reti IP avviene secondo criteri di amministrazione logica
  - **Aggregazione di indirizzi IP** per ridurre regole di routing
  - Assegnazione rispetto ai “ruoli” degli host associati
    - Es: rete dei computer utenti amministrativi, rete degli sviluppatori, rete dei server, …

- Finora abbiamo configurato reti in cui **l’organizzazione logica corrispondeva a quella fisica**, ad esempio **tutti e soli gli host di una rete collegati allo stesso dispositivo di livello 2 (e.g., switch, bridge)**

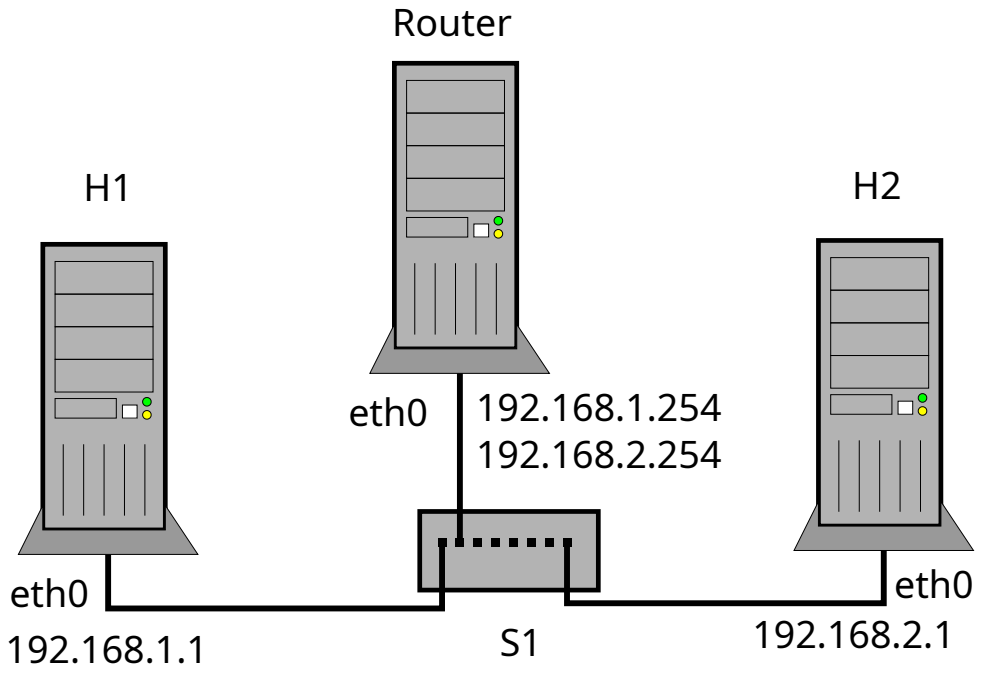
# LAN “tradizionale” [2]

- **Non è sempre possibile** avere reti locali in cui l’organizzazione fisica e logica coincidono, e in molti casi sarebbe estremamente complesso/costoso gestire simultaneamente entrambi gli aspetti
  - Esempio principale: nel contesto del **cablaggio strutturato**, chi si occupa dell’installazione **fisica** dei dispositivi **non si preoccupa dei ruoli di chi utilizzerà la postazioni (organizzazione logica)**
    - “Installiamo uno switch al piano dell’edificio, che verrà utilizzato da tutti gli host di quel piano, a prescindere dagli utilizzatori”
- **Possibile soluzione: configuro più reti sullo stesso switch**
- **Difetto: reti IP differenti possono comunicare a livello H2N**
- **Rischio: possibili attacchi a livello H2N fra reti IP differenti (possibilmente associate ad utenti caratterizzati da privilegi differenti)**

# LAN “tradizionale” [3]

- **Possibile soluzione: configuro più IP reti sullo stesso switch**
- **Difetto: reti IP differenti possono comunicare a livello H2N**
- **Rischio: possibili attacchi a livello H2N fra reti IP differenti (possibilmente associate ad utenti caratterizzati da privilegi differenti)**

**LAN1:** 192.168.1.0/24
**LAN2:** 192.168.2.0/24




*Configureremo la rete fra poco*

# Virtual LAN (VLAN)

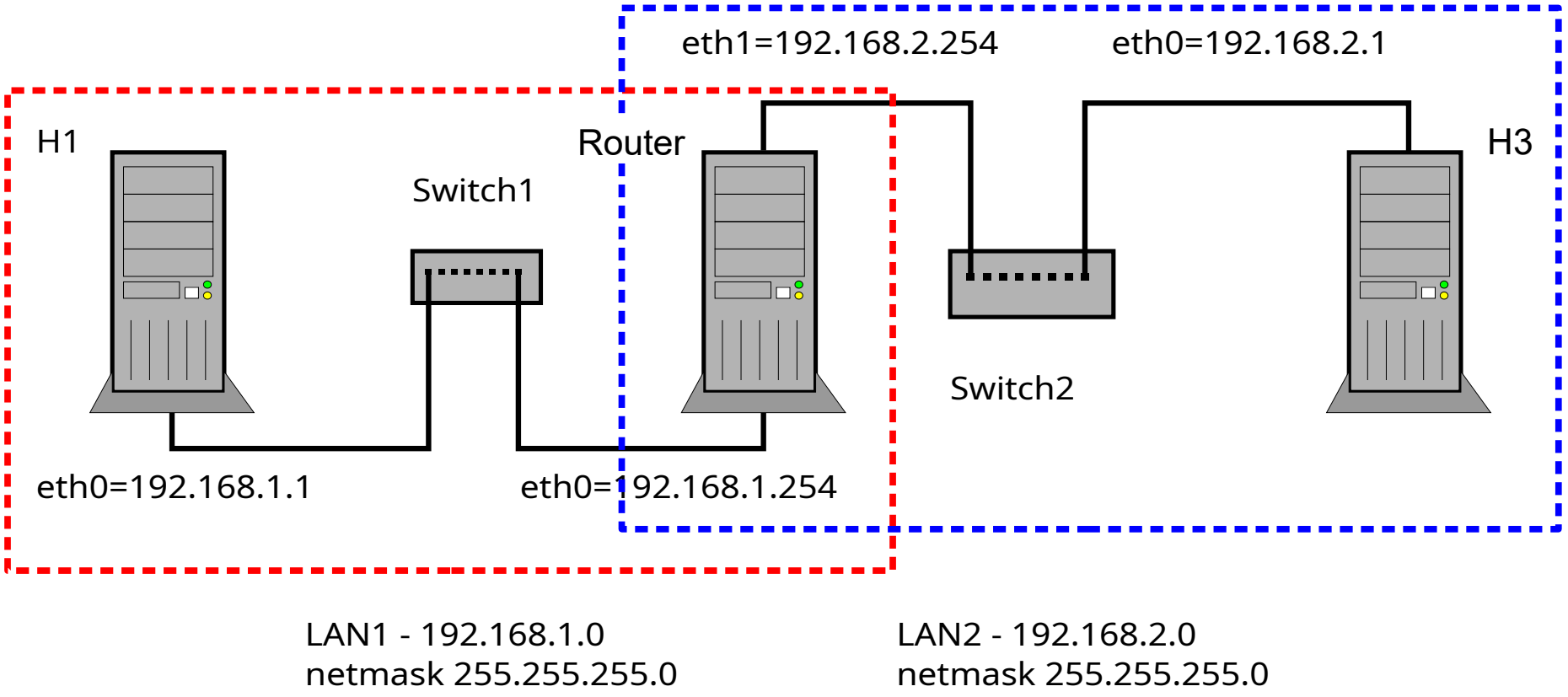
## Da non confondere con le WLAN…

- Le VLAN impiegano delle ***estensioni*** dei protocolli Ethernet per consentire di dare "più intelligenza" agli switch e ai bridge, in modo tale che possano **separare il dominio di broadcast Ethernet di reti IP differenti, anche se collegate alla stesso dispositivo**

- Esistono diversi modi per consentire
  - In questa esercitazione utilizziamo meccanismi **VLAN port-based**, in cui configuriamo staticamente le porte degli switch
    - Esistono anche VLAN port-based dinamiche
- Esistono anche altre metodologie che non vediamo, ad esempio **switch di livello 3** in grado di leggere le informazioni riguardanti gli indirizzi IP nell'header IP, e definire i domini di broadcast di conseguenza

# Routing e dominio di collisione Ethernet [1]

In lezioni precedenti abbiamo analizzato scenari in cui **reti differenti** sono state configurate in **domini di broadcast** differenti (separazione tramite uso di switch differenti)

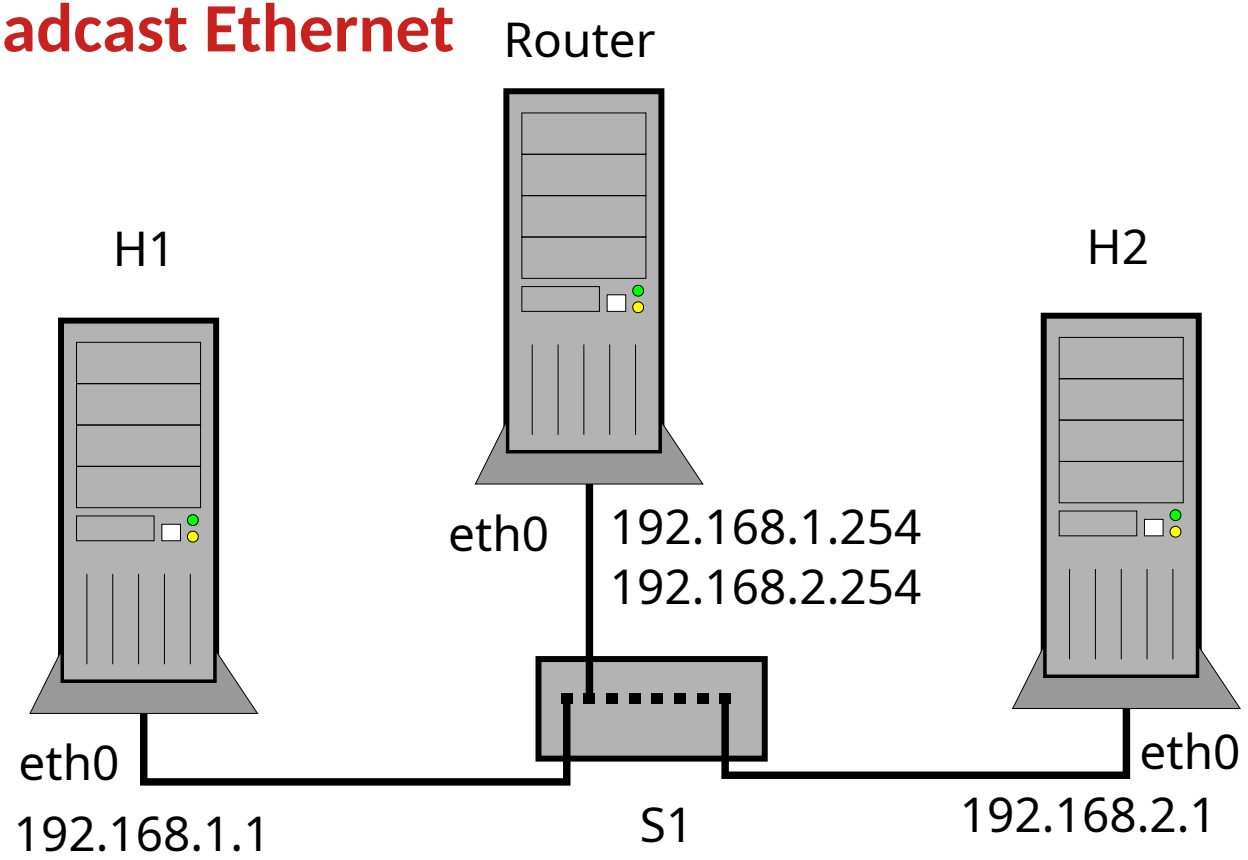


# Routing e dominio di collisione Ethernet [2]

Possiamo però realizzare la stessa rete precedente utilizzando **un unico switch per entrambe le reti IP**

- dal punto di vista "funzionale" le reti sono analoghe a prima
- **!!! dal punto di vista della sicurezza, ora le due reti condividono lo stesso dominio di broadcast Ethernet**

LAN1: 192.168.1.0/24
LAN2: 192.168.2.0/24




*(Vedi slide configurazione indirizzi IP multipli)*

# Virtual Local Area Network

- Mediante le VLAN gli host possono essere raggruppati "logicamente"
  - dipartimento, applicazioni che eseguono, funzioni, livello di riservatezza, …
- Facilità di gestione / Costi
  - La topologia fisica (H2N) rimane indipendente dalla gestione logica (IP), senza sacrificare sicurezza o incrementare i costi
- **Isolamento**
  - **Comunicazione H2N impedita tra VLAN diverse**
  - **Il traffico di broadcast è limitato agli host della VLAN**

# Implementazione

- Tecnicamente creare una VLAN equivale a creare un dominio di broadcast separato
  - Gli host che si trovano all'interno della VLAN possono comunicare direttamente
  - Gli host che si trovano in VLAN differenti possono cominicare mediante l'intermediazione di un dispositivo di rete (router)

- Per risolvere un indirizzo IP, il protocollo ARP individua l'indirizzo di destinazione MAC mediante broadcast
  - La richiesta broadcast arriverà solo agli host facenti parte delle stessa VLAN (ovvero, collegati a tutte e sole le porte dello switch associate a quella VLAN)

# Bridge VLAN-aware (VLAN port-based)

- Access list definiscono quali porte possono ricevere/inviare frame da/verso le diverse VLAN
  - Un frame in arrivo al bridge viene etichettato con l'identificatore numerico della VLAN
  - Viene inoltrato solo sulle porte che possono accedere alla relativa VLAN

- L'assegnazione della VLAN può avvenire in base a diverse proprietà del pacchetto
  - livello 1 - porta di ingresso
  - livello 2 - mac address del frame
  - ma anche a livello 3, 4

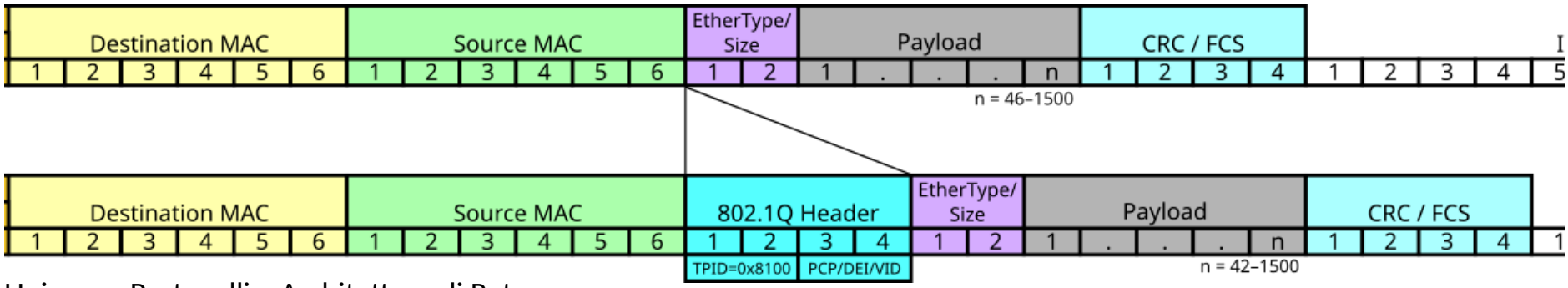
# Collegamento allo switch o bridge

- Access link
  - Usato da dispositivi o segmenti di rete **VLAN-unaware**. Il tagging e l'untagging sono eseguiti in modo trasparente dal bridge
- Trunk link
  - Tutti i dispositivi connessi a questo link sono **VLAN-aware** e sono in grado di interpretare il tag VLAN presente nei frame [i.e. IEEE 802.1q]
- Hybrid link
  - Un link sulla quale possono essere connessi entrambi i tipi di dispositivi.

# IEEE 802.1q

- Lo switch non analizza l'header IP, bensì l'**header Ethernet viene esteso** con un campo aggiuntivo, detto **tag** o **VLAN ID (VID)** di **12bit**
  - 12 bit -> 4094 possibili VLAN (i valori 0 e 4095 riservati)
- **Nota:** l'estensione è prevede di importare il campo **Type** al valore **0x8100** per distinguerlo da un normale frame ethernet
- Il campo **Type "originale"** viene "spostato" di 32 bit
  - **16 per il nuovo campo type=0x8100**
  - 3 bit per Priority Code Point (PCP)
  - 1 per Drop Eligible Indicator (DEI)
  - **12 per VID**

*Feature aggiuntive per QoS, non ci interessano*



# VLAN trunk su Linux

- Linux supporta la creazione di interfacce virtuali che gestiscono il tagging e l'untagging dei frame per una particolare VLAN

- Si crea un'interfaccia “virtuale” figlia di una fisica
  - All'interfaccia virtuale arriveranno solo i pacchetti per la VLAN relativa ricevuti dall'interfaccia fisica
  - I pacchetti inviati tramite l'interfaccia virtuale avranno il tag VLAN aggiunto automaticamente e saranno inviati sull'interfaccia fisica

# VLAN trunk su Linux: Configurazione temporanea

Creazione dell'interfaccia VLAN

```
ip link add link <physif> <virtif> type vlan id <N>
```

Visualizzazione del VID

```
ip -d link show dev <virtif>
```

Rimozione

```
ip link del <virtif>
```

L'interfaccia *<virtif>* si configura con i tool standard.

**NB:** l'interfaccia creata con i precedenti comandi sarà *temporanea*.

# VLAN trunk su Linux: Configurazione permanente ifupdown

Per rendere le modifiche permanenti (con il sistema ifupdown), è necessario inserire nel file /etc/network/interfaces un'interfaccia del tipo *<physif>*.<N>

I parametri sono gli stessi necessari per la configurazione di una classica interfaccia Ethernet. Ad esempio:

```
auto <physif>.<N>
iface <physif>.<N> inet static
  address <ip_address>
  netmask <netmask>
  gateway <ip_addr_gateway>
```

# *vde_switch*

VDE = Virtual Distributed Ethernet http://vde.sourceforge.net/

Progetto Virtual Square (http://wiki.v2.cs.unibo.it)

vde_switch mette a disposizione funzionalità utili per la virtualizzazione di una rete LAN avanzata, configurabili tramite un terminale

- supporto VLAN, bridge, STP, altro...
- distribuito (switch su diverse macchine host )
- modulare
- compatibile con uml e uml_switch

# Console di vde_switch

Comandi principali di vde_switch che useremo:

- **`port`** : gestione delle porte
- **`vlan`** : gestione delle VLAN
- **`hash`** : gestione dell'hash table dello switch

***help [comando] è vostro amico!***

***NB**:* altri comandi potrebbero essere disponibili in seguito al caricamento di plugin (e.g. vedi traffic sniffing con vde_switch con pdump)

# Configurazione delle VLAN su vde_switch

- Creazione di due VLAN sugli switch

  **`vlan/create vlan_number`**

- Impostare la VLAN per ogni porta a cui è collegato un host

  **`port/setvlan port_number vlan_number`**

- Aggiungere la porta collegata all'altro switch alla VLAN

  **`vlan/addport vlan_number port_number`**

**NB**: differenza fra **`port/setvlan`** e **`vlan/addport`**?
Impostano rispettivamente VLAN untagged e tagged

# Esempio prompt di vde_switch: help

```
vde$ help
0000 DATA END WITH '.'
COMMAND PATH        SYNTAX           HELP
------------        --------------   ------------
ds                  ============     DATA SOCKET MENU
ds/showinfo                          show ds info
help                [arg]            Help (limited to arg when specified)
logout                               logout from this mgmt terminal
shutdown                             shutdown of the switch
showinfo                             show switch version and info
load                path             load a configuration script
[…]
```

# Esempio prompt di vde_switch: VLAN

```
vde$ vlan/print
0000 DATA END WITH '.'
VLAN 0000
 -- Port 0001 tagged=0 active=1 status=Forwarding
 -- Port 0002 tagged=0 active=1 status=Forwarding
VLAN 0042
 -- Port 0002 tagged=1 active=1 status=Forwarding
 -- Port 0003 tagged=0 active=1 status=Forwarding
.
1000 Success
```

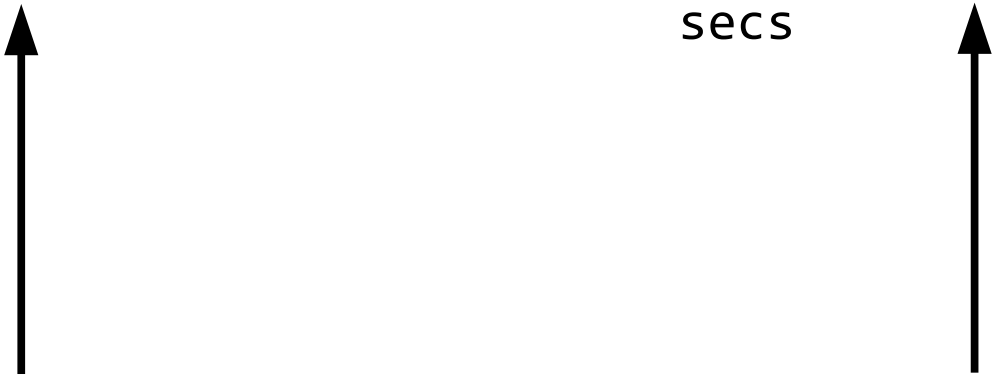
Tagged = 1 : *Trunked Link*, accetta solo pacchetti taggati IEEE 802.1q

Tagged = 0 : *Access Link*, accetta anche pacchetti non taggati, applica il tag a quelli non taggati

# Esempio prompt di vde_switch: hash table

```
vde$ hash/print
0000 DATA END WITH '.'
Hash: 0024 Addr: 6a:9b:43:98:97:27 VLAN 0000 to port: 002 age 1
                                                    secs
Hash: 0114 Addr: 12:42:12:f0:65:c3 VLAN 0000 to port: 001 age 1
                                                    secs

.
1000 Success
```



Il colleganto fra MAC e porta è necessario per conoscere a che porte sono collegati i nodi uml quando ci si collega con stack standard uml (...eth0=daemon,...)