

# Dynamips - Un émulateur de routeur Cisco sur PC

Christophe Fillot

Université de Technologie de Compiègne, Service Informatique  
rue Personne de Roberval, 60200 Compiègne  
Christophe.Fillot@utc.fr

Jean-Marc Berenguier

Université de Technologie de Compiègne, Service Informatique  
rue Personne de Roberval, 60200 Compiègne  
Jean-Marc.Berenguier@utc.fr

## Résumé

*La création de maquettes « réseau » Cisco de taille conséquente se heurte à deux principales problématiques : le coût et la disponibilité des équipements mis en œuvre ou le comportement imparfait des simulateurs de routeurs tel Boson [1] quand ceux-ci sont utilisés, les spécificités des différentes versions d'IOS (fonctionnalités, comportements, bogues) n'étant pas prises en compte.*

*Nous avons développé et présentons **Dynamips** [2], une solution d'émulation hardware de routeurs Cisco, capable de faire fonctionner sur PC des images IOS non modifiées. La possibilité de démarrer et d'interconnecter plusieurs instances de **Dynamips** sur un ou plusieurs serveurs permet de réaliser des maquettes opérationnelles de réseaux complets.*

*Nous présentons également les projets connexes de configuration et de pilotage de **Dynamips**, **Dynagen** [3], interface en mode texte, et **GNS-3** [4], interface graphique.*

## Mots clefs

Routeurs Cisco, IOS, émulation

## 1 Introduction

**Dynamips** est un émulateur de routeurs Cisco capable de faire fonctionner des images Cisco IOS non modifiées comme si elles s'exécutaient sur de véritables équipements. Le rôle de **Dynamips** n'est pas de remplacer de véritables routeurs, mais de permettre la réalisation de maquettes complexes avec de vraies versions d'IOS. Contrairement à certains autres produits, il ne s'agit pas d'une émulation de la ligne de commande IOS et de son fonctionnement, mais d'une émulation complète du « hardware ». **Dynamips** peut être utilisé à des fins de formation, d'expérimentation, aide au diagnostic, validation de configurations, ...

## 2 Présentation générale

**Dynamips** est écrit en langage C, sous licence GPL. Les plateformes hôtes supportées sont de type PC sous Linux, Mac Os X et Windows (via *Cywin* [5]). Un portage sur d'autres plateformes Unix est également possible.

Pour obtenir des performances correctes, un recompileur "JIT" (Just-In Time) recompile le code de l'image IOS à la volée en code natif Intel x86 (ou x86 64-bits). Bien que beaucoup plus lent, un mode d'exécution pas-à-pas existe également pour les plateformes hôtes non supportées (non x86). Une plateforme x86 est donc à préférer.

Les gammes de routeurs émulés sont:

- Cisco 7200 (NPE-100 jusqu'à NPE-400, NPE-G2)
- Cisco 3600 (3620, 3640, 3660)
- Cisco 3700 (3725, 3745)
- Cisco 2600 (2610 à 2651XM, 2691)
- Cisco 1700 (1710 à 1760)

L'émulation de *switches* (de type Catalyst) n'est pas supportée, mais un module de *switching* Ethernet (référence Cisco NM-16ESW) peut être utilisé sur les plateformes 3600, 3700 et 2600.

Les interfaces émulées sont de type:

- Ethernet, FastEthernet, GigabitEthernet,
- Série,
- ATM (Cisco 7200 uniquement),
- POS (Cisco 7200 uniquement).

Différentes instances de l'émulateur peuvent être interconnectées à travers ces interfaces. Une connexion à un réseau réel est réalisable en Ethernet via une interface physique du serveur hôte.

**Dynamips** fonctionnant uniquement en ligne de commande, des projets connexes non portés par l'UTC ont vu le jour :

**Dynagen** est un produit complémentaire écrit en Python s'interfaçant avec **Dynamips** grâce au mode hyperviseur que nous aborderons plus loin. **Dynagen** facilite la création et la gestion de maquettes grâce à un fichier de configuration simple décrivant la topologie du réseau à simuler et une interface texte interactive.

**GNS-3** reprend ces mêmes fonctionnalités sous forme d'interface graphique. Écrit en Python, il s'appuie sur des modules de **Dynagen**.

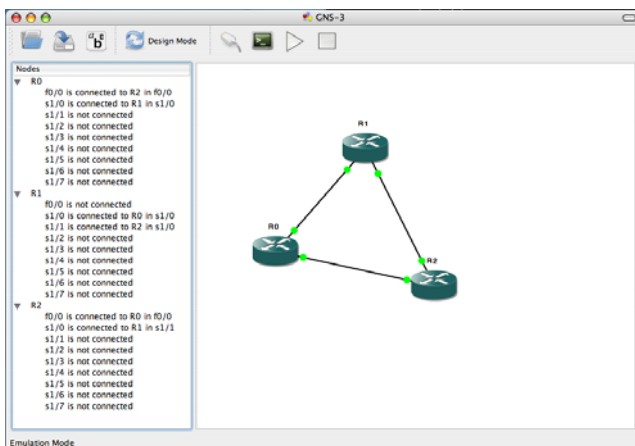


Figure 1 - Une copie d'écran de GNS-3

## 3 Architecture

### 3.1 Exécution du code IOS

Pour permettre l'exécution d'une image IOS, *Dynamips* doit émuler le processeur ainsi que tous les périphériques de la plateforme cible.

Les équipements Cisco étant basés sur des processeurs MIPS (64-bits) ou PowerPC (32-bits), *Dynamips* émule donc ces processeurs.

Pour des raisons de performances, le code MIPS ou PowerPC de l'image IOS est recompilé à la volée en code natif Intel x86 (ou x86 64-bits). Un compilateur JIT a donc été écrit à cet effet.

Par exemple, le code PPC suivant :

```
lis    r9,804
addi   r9,r9,4592
rlwinm r0,r0,2,0,29
```

sera compilé à la volée pour donner le code x86 :

```
mov    $0x32411f0,%ecx
mov    %ecx,0x28(%edi)
mov    0x4(%edi),%eax
rol    $0x2,%eax
and    $0xffffffffc,%eax
mov    %eax,0x4(%edi)
```

Pour les plateformes hôtes non-x86, un mode interpréteur, beaucoup plus lent, est disponible.

### 3.2 Emulation des périphériques

Outre le CPU, *Dynamips* doit également gérer l'ensemble des périphériques qui composent le routeur : mémoire RAM, NVRAM (stockage de la configuration), mémoire Flash, bus PCI, interfaces réseau...

*Dynamips* propose une couche d'abstraction pour la gestion de l'envoi et de la réception des paquets sur les interfaces réseau, appelée « NIO » (Network Input/Output). Cette

couche permet de découpler l'interface réseau émulée (visible sous IOS) de l'interface réseau sur le serveur hôte. Elle gère en particulier la communication inter-instances en s'appuyant sur des sockets UDP ou Unix.

## 4 Utilisation

### 4.1 Hyperviseur

*Dynamips* fonctionne selon deux modes : en ligne de commande Unix classique ou en mode « Hyperviseur ».

En mode « hyperviseur », une instance *Dynamips* attend les ordres de gestion des routeurs virtuels locaux (création, suppression, démarrage, arrêt, ...) et des interconnexions réseau, à travers un canal de commande TCP.

De même, l'accès aux ports console des routeurs émulés s'effectue au travers de canaux TCP dont les caractéristiques (port) sont spécifiés via l'hyperviseur.

### 4.2 Interconnexions réseau

Les interconnexions réseau s'effectuent grâce à des interfaces IOS Ethernet, Série ou ATM auxquelles sont affectées des méthodes de transport de la couche d'abstraction NIO en fonction de leur utilisation.

L'interconnexion d'instances s'appuie sur des sockets UDP ou Unix. Les sockets UDP permettent de relier des routeurs virtuels fonctionnant sur des serveurs différents, les sockets Unix offrant moins d'« overhead » pour des interconnexions locales.

La méthode NIO PCAP permet de faire correspondre une interface émulée IOS à une interface Ethernet physique du serveur hôte.

Enfin, la connexion à des machines virtuelles telles que QEMU [6] ou UML [7] est possible grâce au support de VDE (« Virtual Distributed Ethernet ») [8].

La création de maquettes intégrant à la fois du matériel physique, des routeurs virtuels (sous *Dynamips*) et de machines virtuelles (QEMU, UML) est donc réalisable.

*Dynamips* fournit également une boîte à outils comprenant des *bridges* et des *switches* virtuels Ethernet, ATM et Frame-Relay simplifiés.

### 4.3 Perturbation de trafic

La simulation de problèmes réseau est réalisable par l'adjonction de modules spécifiques. Un module permettant la coupure franche de liens ou la perte de paquets est disponible. Un module de « fuzzing » (modification aléatoire d'un certain nombre de bits) pour altérer des paquets est en cours de développement. Tout autre type d'altération du trafic peut être envisagé par la création de nouveaux modules.

### 4.4 Capture de trafic

*Dynamips* permet la capture de trafic entre instances au format « pcap » (compatible avec les outils tcpdump [9], ethereal/wireshark [10]). La capture est possible sur des liaisons émulées Ethernet et Série. Sur les liaisons série, les protocoles HDLC, PPP et Frame-Relay sont supportés pour la capture.

Indépendamment, l'aide au diagnostic par caractérisation du trafic peut également être réalisée en exploitant les fonctionnalités d'un collecteur « NetFlow » [11] alimenté par des exports « NetFlow » d'IOS. Nous proposons l'utilisation du collecteur « IPFlow » [12] que nous avons également développé.

## 5 De la topologie réseau à la simulation

Nous nous proposons de détailler ici les différentes étapes permettant de construire une maquette opérationnelle à partir de la topologie de la figure 2 ci-dessous :

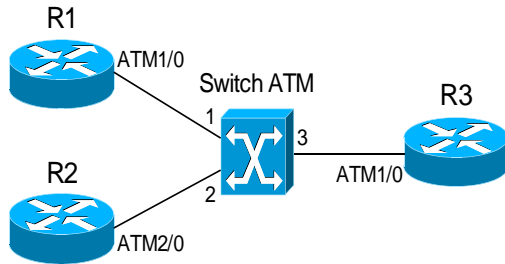


Figure 2 - Topologie

Le switch ATM sera simulé par *Dynamips* en utilisant un switch virtuel simplifié de la boîte à outils. Les routeurs émulés sont de type Cisco 7200.

Deux circuits ATM ont été construits entre R1 et R3, et R2 et R3. Pour cela, les PVC ATM suivants ont été mis en place entre :

- R1 (10/32) et R3 (20/100)
- R2 (10/33) et R3 (20/101)

Dans cette configuration, les flux entre R1 et R2 passent donc par R3.

Enfin, le protocole de routage OSPF a été activé entre les trois routeurs.

Le fichier de configuration de *Dynagen* correspondant à cette topologie est le suivant :

```
autostart = False

[localhost]

[[7200]]
# Type de routeur et image
image=/Users/chris/IOS_pool/extract/C7200-AD.BIN
idlepc = 0x606ba100
ram = 192

[[Router r1]]
# Définition du routeur R1 et
# interconnexion de son interface atm1/0
# avec l'interface 1 du switch ATM virtuel
model = 7200
a1/0 = Switch 1

[[Router r2]]
model = 7200
a2/0 = Switch 2
```

```
[[Router r3]]
model = 7200
a1/0 = Switch 3

[[ATMSW Switch]]
# Définition du switch virtuel ATM et des PVCs
1:10:32 = 3:20:100
2:10:33 = 3:20:101
```

Les extraits pertinents des configurations des trois routeurs sont donnés ci-dessous :

```
!
interface ATM1/0
no ip address
no atm ilmi-keepalive
!
interface ATM1/0.32 point-to-point
ip address 10.10.0.1 255.255.255.252
no snmp trap link-status
pvc 10/32
!
!
router ospf 100
log-adjacency-changes
network 10.0.0.0 0.255.255.255 area 0
!
```

### Configuration de R1

```
!
interface ATM2/0
no ip address
no atm ilmi-keepalive
!
interface ATM2/0.33 point-to-point
ip address 10.11.0.1 255.255.255.252
no snmp trap link-status
pvc 10/33
!
!
router ospf 100
log-adjacency-changes
network 10.0.0.0 0.255.255.255 area 0
!
```

### Configuration de R2

```

!
interface ATM1/0
 no ip address
 no atm ilmi-keepalive
!
interface ATM1/0.100 point-to-point
 ip address 10.10.0.2 255.255.255.252
 no snmp trap link-status
 pvc 20/100
!
!
interface ATM1/0.101 point-to-point
 ip address 10.11.0.2 255.255.255.252
 no snmp trap link-status
 pvc 20/101
!
!
router ospf 100
 log-adjacency-changes
 network 10.0.0.0 0.255.255.255 area 0
!

```

### Configuration de R3

La copie d'écran de la figure 3 ci-dessous présente différentes fenêtres de gestion de la maquette.

La première fenêtre (en haut à gauche) est celle de l'hyperviseur *Dynamips*. On peut y voir les sorties correspondant au démarrage des différentes instances.

L'hyperviseur a été démarré en tapant la commande suivante dans cette fenêtre :

```
# dynamips -H 7200
```

*Dynamips* attend ensuite les commandes envoyées par *Dynagen* sur le port TCP 7200 spécifié par l'argument `-H`.

La seconde fenêtre (en haut à droite) est celle de *Dynagen*.

La saisie des commandes « telnet r1 » et « telnet r2 » a provoqué l'ouverture des fenêtres du bas de la copie d'écran. Elles correspondent aux ports console des deux routeurs R1 et R2.

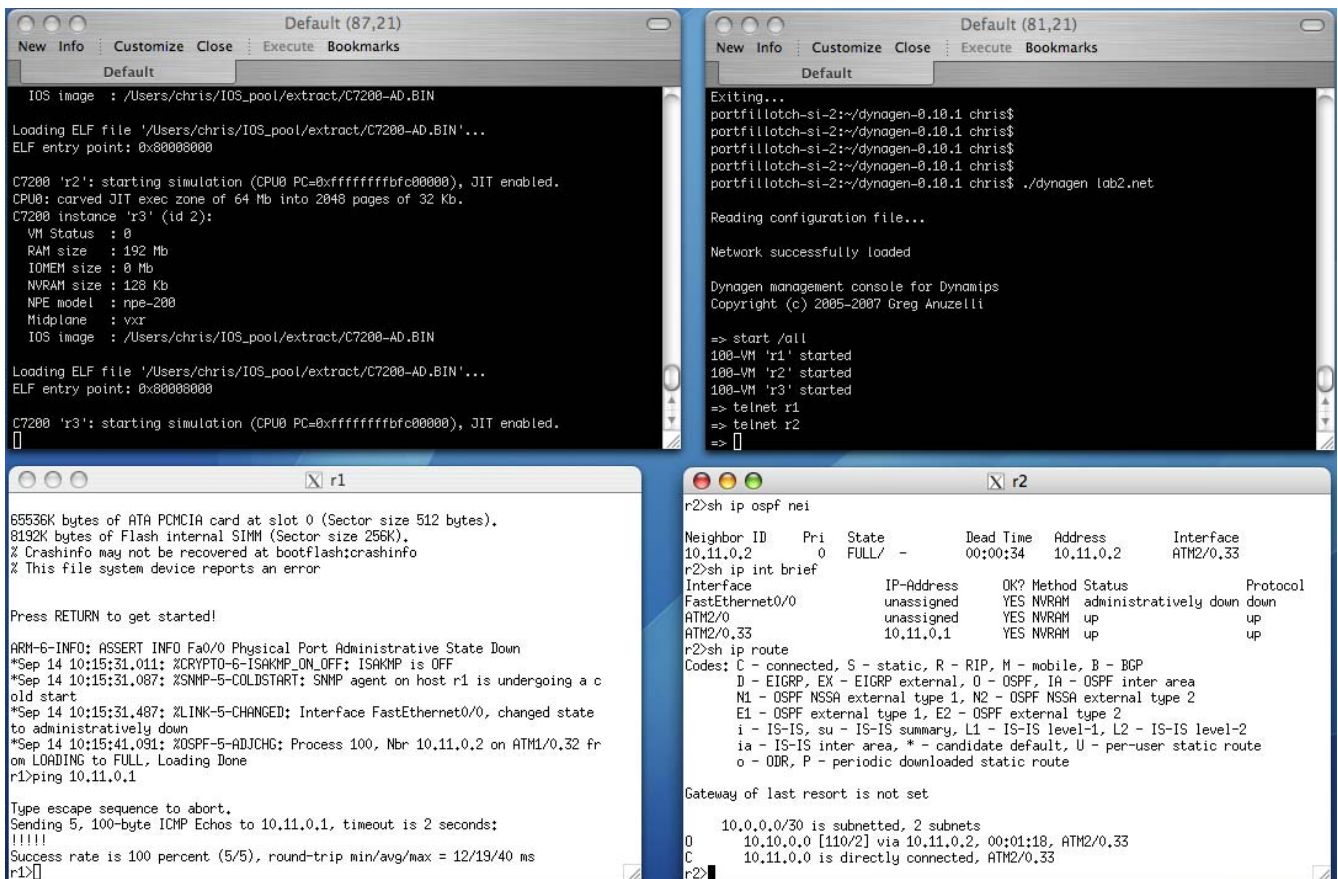


Figure 3 - Fenêtres de gestion de la maquette

## 6 Perspectives

*Dynamips* possède une importante communauté d'utilisateurs [13], source de nombreuses propositions d'évolutions.

*Dynamips* est amené à évoluer avec le support de nouvelles plateformes Cisco et de nouveaux modules réseau.

Un effort particulier sera porté sur la partie *switching*.

Nous accueillerons volontiers tout « contributeur » passionné par le domaine de l'émulation de processeurs et de « hardware », et par les plateformes Cisco.

## Bibliographie et Liens

- [1] Boson Software, Boson, <http://www.boson.com/>
- [2] Christophe Fillot, Jean-Marc Berenguier, Dynamips, [http://www.ipflow.utc.fr/index.php/Cisco\\_7200\\_Simulator](http://www.ipflow.utc.fr/index.php/Cisco_7200_Simulator)
- [3] Greg Anuzelli, Dynagen, <http://dynagen.org>
- [4] Projet de fin d'études EPITECH GNS-3, <http://www.gns3.net>
- [5] Cygwin, <http://www.cygwin.com/>
- [6] Fabrice Bellard, QEMU, <http://fabrice.bellard.free.fr/qemu/>
- [7] User-Mode Linux, UML, <http://user-mode-linux.sourceforge.net/>
- [8] Virtual Distributed Ethernet, VDE, <http://vde.sourceforge.net/>
- [9] Tcpdump, <http://www.tcpdump.org/>
- [10] Wireshark (Ethereal), <http://www.wireshark.org/>
- [11] Cisco NetFlow Version 9, RFC3954, <http://www.ietf.org/rfc/rfc3954.txt>
- [12] Christophe Fillot, Jean-Marc Berenguier, Collecteur Netflow « IPFlow », <http://www.ipflow.utc.fr>
- [13] Forum de la communauté d'utilisateurs de *Dynamips*, <http://7200emu.hacki.at>

