

# ***Manuel d'installation et d'utilisation de Network-In ! version 2.0***



## Table des matières

<b>PRÉSENTATION ET INSTALLATION.....</b>	<b>5</b>
A propos de cette documentation.....	6
Pourquoi Network-In! ?.....	6
Existence et correction des bugs.....	7
Plate-formes supportées.....	7
Linux.....	7
Windows.....	7
Autre.....	7
Installation sous Linux.....	8
Installation type Debian (fichier .deb).....	8
Installation type Ubuntu (fichier .deb).....	8
Installation Linux universelle (fichier .tgz).....	8
Installation sous Windows.....	9
Configuration.....	9
Fichier de configuration principal.....	9
Paramètres de configuration.....	9
Configuration personnalisée.....	10
<b>MISE EN ŒUVRE DU SIMULATEUR.....</b>	<b>12</b>
Démarrage du simulateur.....	13
Aspect général.....	13
Démarrage de la simulation.....	13
Démarrage d'un matériel.....	14
Démarrage de l'ensemble de la simulation.....	15
Arrêt de la simulation.....	15
Gestion des projets.....	15
Sauvegarde du projet courant.....	15
Enregistrer le projet.....	15
Ouvrir un projet existant.....	15
Concevoir un nouveau projet.....	16
Types de matériels disponibles.....	16
Ajouter un matériel.....	16
Connecter les matériels entre eux.....	17
Paramétrier le matériel.....	17

# *Simulateur Network-In ! version 2.0*

Interface graphique d'un matériel.....	17
Informations.....	18
Informations sur un matériel.....	18
Informations sur une connexion.....	18
Commentaires associés à un matériel.....	18
Aspect de l'interface du simulateur.....	19
Interface pour débutant.....	19
Interface complète.....	20
Problèmes rencontrés.....	20
Noyau manquant sur un ordinateur ou routeur.....	20
Disque système manquant sur un ordinateur ou routeur.....	21
Matériel VM VirtualBox indisponible.....	21
<b>MINI-DIDACTICIEL POUR COMMENCER.....</b>	<b>22</b>
Objectif.....	23
Dessin du réseau.....	23
Configuration IP.....	26
Configuration d'un service Web.....	30
<b>RÉFÉRENCES TECHNIQUES.....</b>	<b>33</b>
Présentation technique.....	34
Généralités.....	34
Matériels de type ordinateurs et routeurs.....	34
User mode Linux.....	34
Interface graphique.....	34
Ajout d'un noyau spécifique.....	35
Ajout d'un disque système spécifique.....	35
Switchs et hubs virtuels.....	36
Câbles de connexion.....	36
Affichage des machines virtuelles.....	36
Technique de masquage de l'IP de communication avec Xorg.....	37
Configuration pare-feu pour gérer le flux d'affichage.....	37
Gestion des VM VirtualBox avec le simulateur.....	38
Portage sous Windows.....	38
Caractéristiques des composants.....	39
Ordinateur PC.....	39

## *Simulateur Network-In ! version 2.0*

Ordinateur portable.....	40
Ordinateur Linux mode texte.....	41
Ordinateur serveur.....	42
Routeur.....	43
Routeur NAT réseau réel.....	44
Machine virtuelle VirtualBox.....	45
Pont avec l'hôte.....	46
Commutateur non administrable.....	47
Concentrateur.....	47

## **Présentation et installation**

Si vous désirez vous plonger rapidement dans le maniement de Network-In!, faites donc un tour du côté du didacticiel fourni plus loin dans cette documentation

## **A propos de cette documentation**

Il est possible que cette documentation ait été mise à jour afin de corriger des erreurs ou ajouter certaines informations.

Consultez donc l'adresse : <https://network-in.vverdon.fr>

Ce document ainsi que le logiciel Network-In ! sont diffusés sous licence GNU GPL version 3, définie par la Free Software Foundation ([www.fsf.org](http://www.fsf.org)).

Ils sont libres d'utilisation et de modification, dans les limites de leur licence.

Network-In! et sa documentation sont réalisés par V. Verdon Corp. !

## **Pourquoi Network-In ! ?**

En 2010, j'étais professeur en sciences et technologies industrielles (STI). Le gouvernement a décidé de réformer cette filière qui allait désormais s'appeler sciences et techniques de l'industrie et du développement durable (STIDD).

La création de ce nouveau Bac a vu émerger l'enseignement des réseaux, tant dans le tronc commun (couches OSI, IP, MAC, architecture d'un réseau, protocoles HTTP et FTP,...) que dans la spécialité SIN (architecture client/serveur, analyse de trames,...).

Il existe de nombreux logiciels dont l'objectif est de simuler les réseaux. Ils sont souvent commerciaux ou non libres, ou encore trop hermétiques pour une utilisation à ce niveau.

C'est ainsi qu'est venue l'idée de développer un nouveau logiciel : « Network-In ! ».

« Network-In ! » s'inspire des logiciel « Packet Tracer » de Cisco et « Marionnet » développé à l'Université Paris-13.

A « Packet Tracer », j'ai emprunté l'ergonomie et la souplesse d'utilisation. Ce logiciel est excellent, mais avec deux défauts majeurs :

1. D'une part, il n'est pas libre et son utilisation en classe nécessite (officiellement) d'être formateur Cisco.
2. D'autre part, il est très orienté (en toute logique) vers le matériel Cisco, ce qui n'est pas forcément souhaitable si la finalité est généraliste.

A « Marionnet », j'ai emprunté la technologie des composants virtuels. En fait, le fonctionnement de « Network-In ! » s'inspire fortement de ce logiciel, en gommant ce que je considère comme des petits défauts :

- S'il fonctionne très bien, ce logiciel est austère, il n'y a aucune interface de configuration simple pour un débutant (il faut savoir configurer des machines sous Linux en mode texte).
- Le placement des matériels et câbles n'est pas très ergonomique non plus.
- La logique de mise en place du réseau n'est pas respectée.
- Les sauvegardes et réouverture d'un projet sont très longues, du fait

## *Simulateur Network-In ! version 2.0*

qu'il faut enregistrer à chaque fois les disques virtuels des machine UML (voir explications à suivre).

Marionnet reste un très bon logiciel et je remercie mille fois ses développeurs pour m'avoir fourni les bases nécessaires à la réalisation de Network-In!.

### **Existence et correction des bugs**

Network-In! est un logiciel libre, prévu pour vous rendre service, dont les défauts (bugs) sont (ou seront) corrigés le mieux possible.

Il est par contre fourni sans aucune garantie d'aucune sorte. L'auteur ne peut être tenu responsable de problèmes qui pourraient survenir !

Je l'utilise depuis un moment sans aucun problème...

Il est possible de me contacter pour faire remonter les problèmes rencontrés. Je vous répondrai dans la mesure du possible

<https://network-in.vverdon.fr>

### **Plate-formes supportées**

#### ***Linux***

Ce logiciel est destiné à fonctionner sous Linux, en version 64bits.

Sous Linux, les distributions testées sont :

- Debian (plate-forme de développement) ;
- Ubuntu.

Il n'y a aucune raison que cela ne fonctionne pas avec d'autres distributions, mais je n'ai pas testé. Merci à ceux qui le feraient de me faire remonter les informations et les problèmes éventuels.

#### ***Windows***

Network-In! est disponible sous Windows en utilisant la technologie WSL.

Certains composants du simulateur ne peuvent pas fonctionner :

- Machines virtuelles VirtualBox externes liées à une maquette ;
- Ponts avec la machine hôte.

#### ***Autre***

Il est possible d'exécuter Network-In! dans une machine virtuelle Linux (VirtualBox ou VmWare), en perdant les mêmes fonctionnalités que sous Windows.

## **Installation sous Linux**

### ***Installation type Debian (fichier .deb)***

Network-in! est disponible sous la forme d'un paquet .deb prévu pour une distribution Debian, Ubuntu, etc.

Ouvrir un terminal.

Devenir admin (root) : su - (saisir le mot de passe de root)

Installer en utilisant la commande **apt**, qui se charge d'installer automatiquement les paquets nécessaires au simulateur:

**apt install <dossier du paquet deb>/network-in-<version>.deb**

Exemple :

```
apt install /home/vincent/Téléchargements/network-in-2.0.deb
```

### ***Installation type Ubuntu (fichier .deb)***

Ouvrir un terminal.

Installer en utilisant la commande **sudo apt**, qui se charge d'installer automatiquement les paquets nécessaires au simulateur:

**sudo apt install <dossier du paquet deb>/network-in-<version>.deb**  
(saisir le mot de passe de l'utilisateur)

Exemple :

```
sudo apt install /home/vincent/Téléchargements/network-in-2.0.deb
```

### ***Installation Linux universelle (fichier .tgz)***

Network-In! a besoin des paquets logiciels suivants :

bash , uml-utilities , vde2 , sudo , tcl(>=8.6) , tk(>=8.6) , lssof , wmctrl , x11-utils , iptables , xfwm4 , xserver-xephyr , xsel , hsetroot, x11-xserver-utils, xterm

Les noms des paquets peuvent différer suivant les distributions.

L'installateur refusera de poursuivre si ces logiciels ne sont pas présents.

Ouvrir un terminal.

Devenir admin (root) : su - (saisir le mot de passe de root)

utiliser la commande tar pour désarchiver l'archive .tgz :

**tar -xzf network-in\_2.0\_amd64.tgz**

Vous obtenez alors deux fichiers : install.sh et network-in.tar

Exécutez le script d'installation en lançant la commande :

**./install.sh**

## **Installation sous Windows**

Note : dû à la limitation du composant WSL, l'installation doit être faite pour chaque compte qui désire utiliser le simulateur.

Exécutez l'installateur en tant que **simple utilisateur** (pas en mode administrateur) : **Network-In-install-2.0.exe**

L'installateur commence par vérifier si le composant WSL est installé sur le système. Si WSL n'est pas installé l'installation de Network-In! s'interrompt et vous devez installer WSL en suivant la méthode suivante :

1. Ouvrez un terminal cmd.exe en tant qu'administrateur
2. Exécutez la commande wsl.exe --install --no-distribution
3. Redémarrez l'ordinateur.

Relancez l'installateur de Network-In!.

## **Configuration**

Network-In! fonctionne directement après installation.

Néanmoins, vous aurez peut-être besoin de modifier certains paramètres pour personnaliser la configuration du logiciel.

### **Fichier de configuration principal**

Le fichier de configuration global est */etc/network-in.cfg*

Le rôle des différents paramètres de configuration est indiqué dans le paragraphe suivant.

Dans le fichier de configuration, il est possible d'utiliser les variables d'environnement du système, notamment pour configurer l'espace utilisateur : \$env(USER), \$env(HOME), etc.

Il n'est pas nécessaire en général de modifier la configuration proposée.

### **Paramètres de configuration**

Paramètre	usage	valeurs typiques
coul(bg_schema)	Couleur du fond de la zone de schéma	Valeur en hexadécimal : #bbbbcc
set coul(bg_simul)	Couleur de fond de la zone de simulation	Valeur en hexadécimal : #555577

## Simulateur Network-In ! version 2.0

<b>Paramètre</b>	<b>usage</b>	<b>valeurs typiques</b>
coul(texte)	Couleur du fond de la zone de dessin du réseau.	Valeur en : Grey7
coul(note)	Couleur de fond de l'étiquette d'affichage d'un commentaire.	Valeur en hexadécimal : #f9e79f
lang	Langue par défaut de l'interface du logiciel.	Choix par défaut du système d'exploitation : auto (sous Linux seulement) Langue française : fr Langue anglaise : en
niveau(defaut)	Niveau de détail de l'interface au démarrage d'un projet.	Interface limitée pour débutant : 1 Interface complète : 2
niveau(max)	Niveau de détail maximum possible pour l'interface.	2 est la valeur maximum possible, correspondant à l'interface complète.
rep_home	Localisation du répertoire personnel de l'utilisateur.	\$env(HOME)
rep_proj	Localisation du répertoire du projet courant.	\$rep_home/network-in-proj/
rep_conf	Localisation du répertoire de configuration personnalisée du simulateur pour l'utilisateur.	\$rep_home/.network-in/
mac_prefix	Les deux premiers chiffres hexa de l'adresse Mac. Attention à choisir des valeurs ne correspondant pas à des adresses constructeur.	0a
ip_hote masque_hote	Paramètres de configuration IP pour la communication de l'interface graphique des machines virtuelles Usermode Linux avec la machine hôte.	172.31.255.1 255.255.255.0
screen	Numéro d'écran pour affichage des machines	:5
x_wm	WM manager utilisé pour la fenêtre d'affichage des machines	xfwm4 --display \$::screen
kernel	version du noyau par défaut (doit être installé, voir plus loin)	6.1.82
img_disk_main	nom de l'image disque par défaut	bookworm_1.01.img
img_disk_linux	nom de l'image disque machine Linux text	bookworm_linux_1.01.img

### **Configuration personnalisée**

Un utilisateur peut adapter la configuration du logiciel en plaçant des éléments de configuration dans le répertoire `.network-in/network-in.cfg` (par

## *Simulateur Network-In ! version 2.0*

défaut, modifiable par la configuration globale) situé dans son répertoire personnel.

## **Mise en œuvre du simulateur**

Si vous désirez vous plonger rapidement dans le maniement de Network-In!, faites donc un tour du côté du didacticiel fourni plus loin dans cette documentation

## Démarrage du simulateur

Le simulateur démarre en cliquant sur l'icône ou en invoquant la commande : **network-in** ou **networkin**

Lors du premier démarrage, le logiciel crée automatiquement :

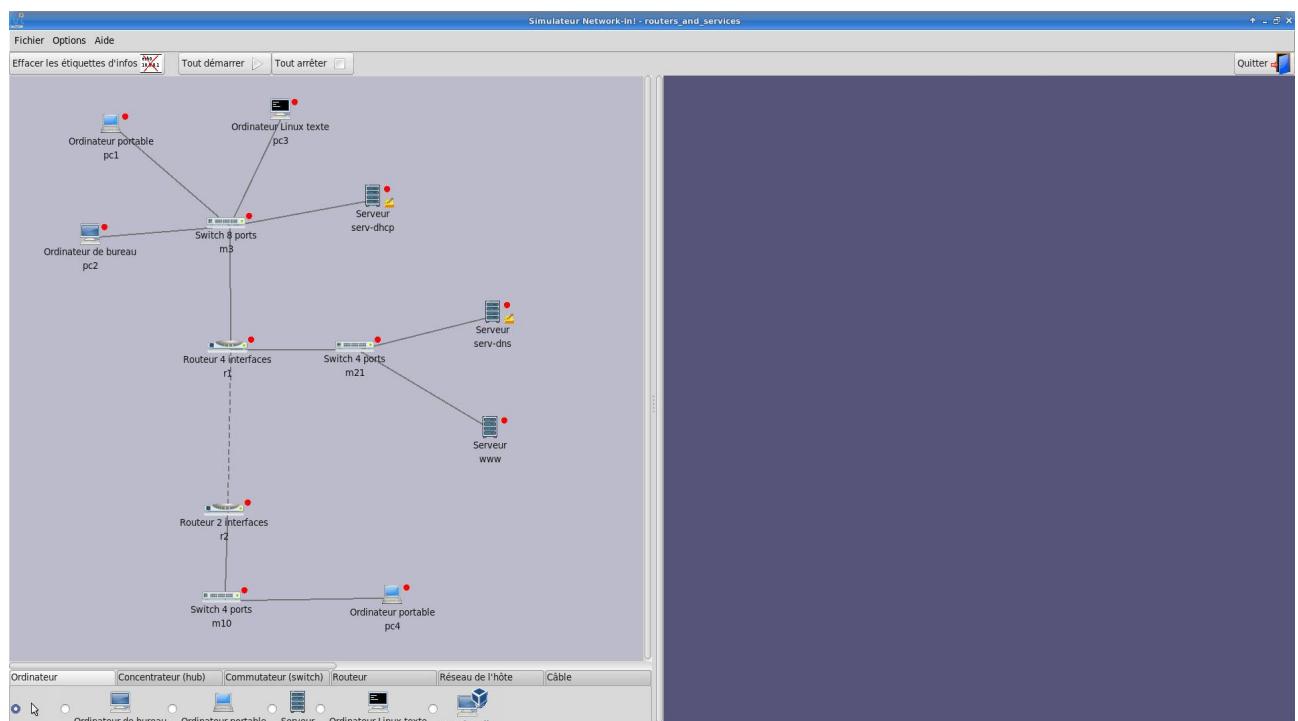
- le répertoire du projet actif *network-in-proj*/
- le répertoire de configuration personnelle de l'utilisateur *.network-in/*

## Aspect général

Le simulateur présente deux parties sous la forme de deux panneaux redimensionnables :

- Une zone de schéma du réseau côté gauche
- Une zone de simulation côté droit

Il est facile d'agrandir l'une des zones pour obtenir plus de place pour le schéma ou la simulation.



La fenêtre Network-In! : à gauche le schéma, à droite la zone de simulation (simulation arrêtée)

## Démarrage de la simulation

En phase de simulation, la fenêtre de simulation de Network-In! montre une représentation des différentes machines en cours de simulation.

Sur l'exemple ci-après on voit :

## Simulateur Network-In ! version 2.0

1. Les écrans des ordinateurs en cours de simulation. 7 ordinateurs sont démarrés ici,
  - 3 PC "client" avec un bureau
  - 1 PC "client" en mode texte (Linux)
  - 1 serveur DHCP
  - 1 serveur DNS
  - 1 serveur Web
2. 2 routeurs sont démarrés et présentent une interface graphique pour leur configuration

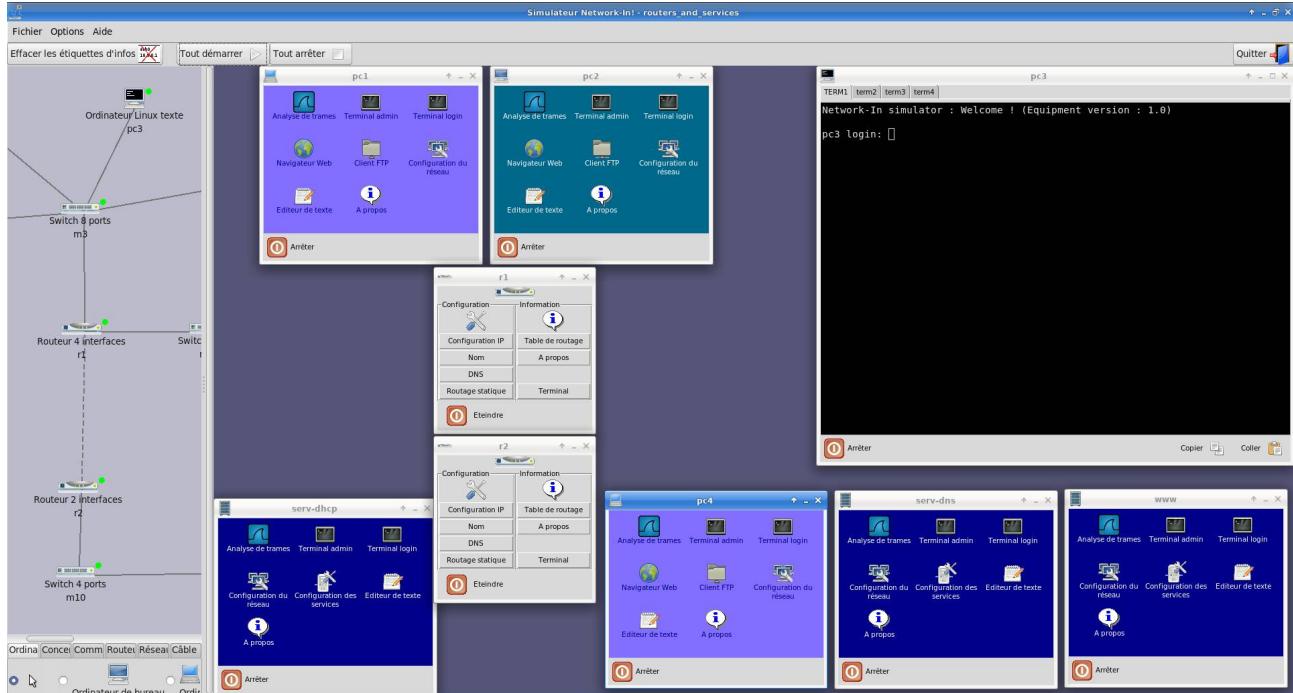


Figure 1: Simulation en cours. La fenêtre de simulation a été agrandie

### Démarrage d'un matériel

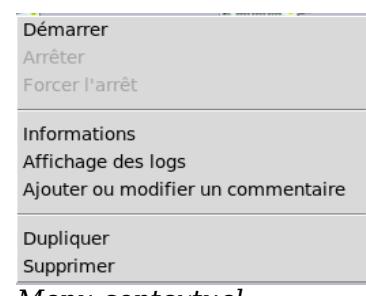
Chaque matériel peut être démarré manuellement.

Un clic droit sur le matériel à démarrer mène au menu contextuel qui comporte la commande nécessaire.

Après un temps plus ou moins long suivant le type de matériel, le matériel est démarré.

Un matériel a trois états possibles. Il peut être :

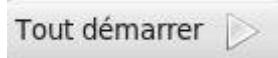
- Arrêté, visuellement, cela est représenté par un voyant rouge
- En cours de démarrage, le voyant est orange
- Démarré, le voyant est vert



Menu contextuel :  
démarrer

## *Simulateur Network-In ! version 2.0*

### **Démarrage de l'ensemble de la simulation**

Le bouton  permet de démarrer l'ensemble des machines qui peuvent l'être.

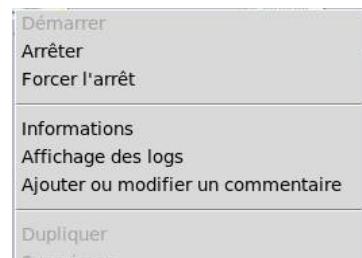
Si VirtualBox n'est pas installé ou que la VM VirtualBox n'est pas accessible, alors le composant ne démarre pas mais aucune erreur n'est déclenchée.

### **Arrêt de la simulation**

De la même manière il est facile d'arrêter l'ensemble des matériels ou bien certains.

Il peut arriver qu'un matériel ne réponde plus, tout comme dans la réalité !

Dans ce cas le menu contextuel permet de forcer son arrêt. Attention tout comme avec un véritable matériel, le système peut se trouver endommagé.



*Menu contextuel :  
arrêter*

## **Gestion des projets**

### **Sauvegarde du projet courant**

Il n'y a rien à faire, le projet est automatiquement sauvegardé au fur et à mesure que vous modifiez votre simulation.

Au prochain démarrage, vous retrouverez le projet tel que vous l'avez laissé.

### **Enregistrer le projet**

Si vous voulez passer à une autre simulation ou si vous voulez enregistrer votre projet pour le transférer sur un autre ordinateur, il faut l'archiver.

Cette action s'effectue via l'habituel menu **Fichier > Enregistrer le projet actuel**.

Suivant la complexité du projet et surtout les modifications apportées aux machines virtuelles utilisées, la sauvegarde peut être assez longue.

### **Ouvrir un projet existant**

Cette action s'effectue via le menu **Fichier > Ouvrir un projet existant**.

Si vous ne sauvegardez pas le projet actuellement affiché, il sera perdu. Une boîte de message le rappelle.

L'ouverture d'un projet archivé est presque immédiate.

## *Simulateur Network-In ! version 2.0*

Le premier démarrage d'une des machines virtuelles peut être un peu plus long qu'à l'habitude (décompression de l'archive de la machine).

### ***Concevoir un nouveau projet***

Pour démarrer un nouveau projet, il suffit d'invoquer le menu **Fichier > Nouveau projet**.

L'espace de conception de réseau est désormais vide.

### ***Types de matériels disponibles***

Différents matériels sont disponibles en version 2.0 :

- Des ordinateurs de bureau ;
- Des ordinateurs portables ;
- Des ordinateurs serveur ;
- Des ordinateurs sans interface graphique de type Linux ;
- Des hubs ;
- Des switchs non administrables ;
- Des routeurs, avec routage statique ;
- Des ponts qui permettent de créer et connecter des interfaces réseaux (TAP) de la machine hôte en tout point du réseau du simulateur ;
- Des machines virtuelles VirtualBox ;
- Un routeur NAT, offrant une connexion du réseau du simulateur avec l'internet.

Les PC clients et ordinateurs portables recouvrent pour le moment le même type de matériel, hormis la représentation sur le schéma et le fait qu'on ne peut modifier la carte réseau (adresse MAC) d'un ordinateur portable

Un ordinateur serveur propose des services de base :

- DHCP ;
- DNS ;
- HTTP ;
- FTP ;
- SSH.

### ***Ajouter un matériel***

Il suffit de sélectionner, dans les onglets situés en bas de la fenêtre, le type de matériel voulu puis de cliquer sur le modèle précis. Ensuite, en cliquant sur l'espace de travail, on dépose le matériel sélectionné.

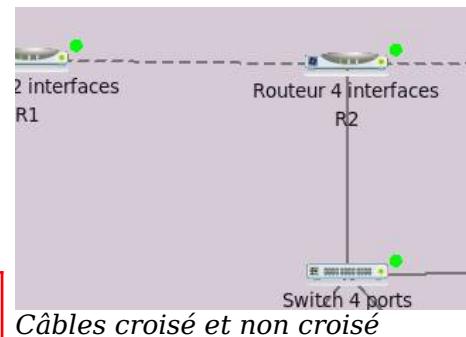


*Onglets de choix de matériel*

## **Connecter les matériels entre eux**

Les matériels peuvent être reliés actuellement :

- par des câbles non croisés, représentés par des **lignes continues** ;
- par des câbles croisés, représentés par des **lignes pointillées**.



Les types de matériels DCE (Data Communication Equipment) et DTE (Data Terminal Equipment) sont pris en compte, si bien qu'il est nécessaire d'utiliser le bon câble sinon la connexion ne fonctionne pas.

## **Paramétrier le matériel**

A l'issue du "câblage" réseau, l'architecture est en place. Mais aucun matériel n'est encore paramétré.

En effet, dans un cas réel, si vous installez un nouveau PC, vous devrez le raccorder au réseau à l'aide de câbles puis ensuite le démarrer pour le paramétrer.

Le simulateur Network-In ! respecte la démarche qui consiste à démarrer un ordinateur pour pouvoir le configurer, si bien qu'il n'est pas possible, à l'avance, de décider de son IP ou de son nom par exemple.

## **Interface graphique d'un matériel**

La plupart des matériels affichent une fenêtre après démarrage.

Certains composants ne sont pas configurables et ne proposent pas d'interface graphique :

- hubs
- switchs non administrables

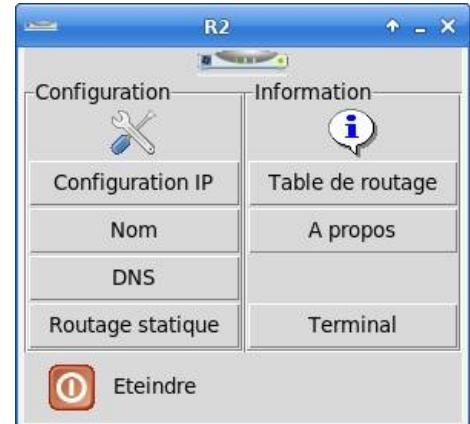
L'interface graphique représente le matériel en question et permet sa configuration.

Pour un ordinateur, la fenêtre est une représentation de son bureau, comportant des icônes cliquables (doubles-clics).

## *Simulateur Network-In ! version 2.0*



*Interface graphique d'un ordinateur sous la forme d'un bureau*



*Interface graphique d'un routeur*

Pour un routeur, une fenêtre donne accès aux divers éléments de configuration du matériel.

## **Informations**

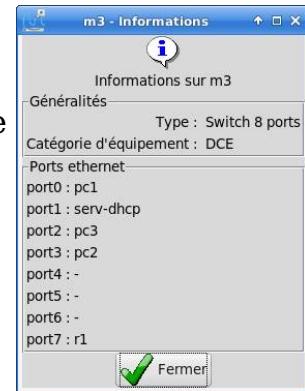
### **Informations sur un matériel**

Un clic droit sur le matériel considéré permet de sélectionner l'entrée Informations du menu contextuel. Une fenêtre s'ouvre alors, donnant un certain nombre d'informations variables suivant le type de matériel.

### **Informations sur une connexion**

Un clic sur une connexion fait apparaître des informations sur la configuration des matériels connectés au deux extrémités :

- Le noms de l'interface ou du port, eth pour l'interface ethernet d'une carte réseau d'ordinateur ou de routeur, port pour un switch ou un hub.
- L'adresse IP affectée à l'interface quand le matériel en possède une et qu'il est démarré ;
- Le masque associé à l'IP.



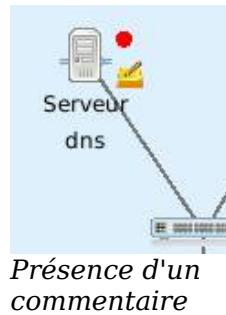
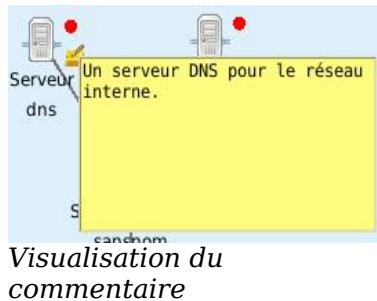
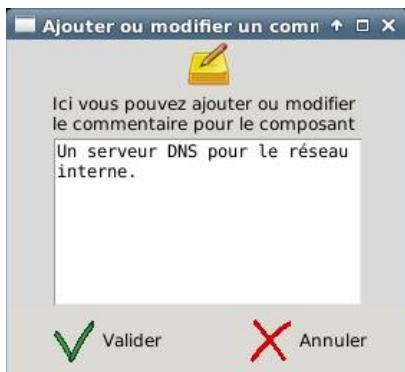
*Figure 2: Fenêtre d'information (ici un switch)*

### **Commentaires associés à un matériel**

On peut associer à un matériel un icône représentant un commentaire.

## *Simulateur Network-In ! version 2.0*

En cliquant sur cet icône, on pourra visualiser la note associée au matériel :



*Fenêtre de saisie*

Il est très facile de saisir, modifier ou supprimer un commentaire, en choisissant l'entrée **Ajouter ou modifier un commentaire** du menu contextuel du matériel concerné.

Pour supprimer un commentaire, effacer son texte depuis la fenêtre de saisie.

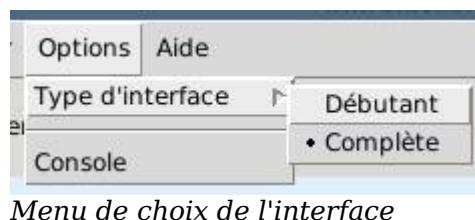
## **Aspect de l'interface du simulateur**

L'interface de Network-In! peut être configurée pour s'adapter au niveau de l'utilisateur.

Actuellement, deux niveaux existent :

- L'interface pour débutant ;
- L'interface complète.

On peut basculer d'une interface à l'autre depuis le menu Options>Type d'interface, sauf si la configuration du logiciel a été modifiée pour limiter le niveau d'interface accessible.



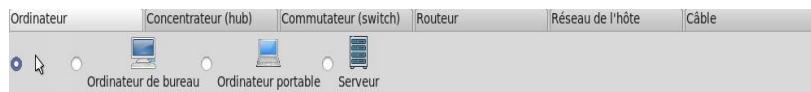
Quand vous basculez d'une interface à l'autre, le choix reste actif au redémarrage du logiciel, jusqu'au prochain changement de projet.

### **Interface pour débutant**

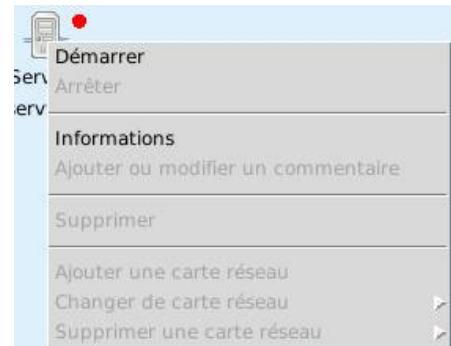
Elle correspond au niveau 1 dans le fichier de configuration du logiciel.

## *Simulateur Network-In ! version 2.0*

Les composants plus complexes, notamment ceux qui permettent la communication avec la machine hôte sont retirés.



*Interface débutant : certains onglets et composants ont disparus*



*Avec l'interface débutant, certaines actions sont impossibles pour certains composants*

Par contre, même si ce niveau d'interface est activé, il reste possible de démarrer une simulation qui comporte des composants non disponibles dans cette interface. Par contre, certaines actions sont rendues impossibles sur ces composants.



*Interface complète : tous les onglets et composants sont disponibles.*

## Problèmes rencontrés

### **Noyau manquant sur un ordinateur ou routeur**

Chaque matériel de type ordinateur ou routeur utilise un noyau Linux. Si la version d'origine n'est pas présente, alors le simulateur utilise la version définie par défaut dans sa configuration. Dans ce cas, certains modules du noyau peuvent ne pas être installés et cela peut poser problème.

## *Simulateur Network-In ! version 2.0*

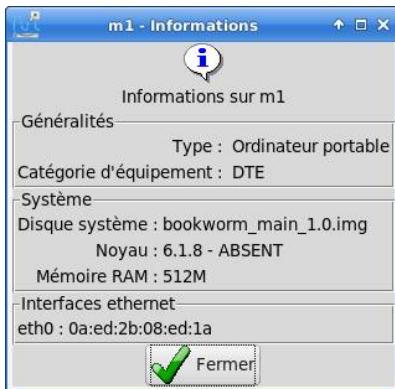


Figure 3: Noyau manquant

Se référer aux explications fournies en annexe<sup>35</sup> pour l'ajout d'un noyau.

### **Disque système manquant sur un ordinateur ou routeur**

Si le disque système de base n'est pas présent, alors le simulateur ne peut pas démarrer le matériel concerné.



Figure 4: Disque système absent,  
démarrage impossible

Se référer aux explications fournies en annexe<sup>35</sup> pour l'ajout ou la création d'un disque système.

### **Matériel VM VirtualBox indisponible**

Cela peut se produire si :

- VirtualBox n'est pas installé : les icônes des VM sont grisés sur le schéma et présentent un point d'interrogation.
- La VM VirtualBox n'est pas présente sur la machine hôte ou n'a pas encore été déclarée dans la configuration du matériel VM VirtualBox du schéma. L'icône n'est pas grisé mais présente un point d'interrogation.

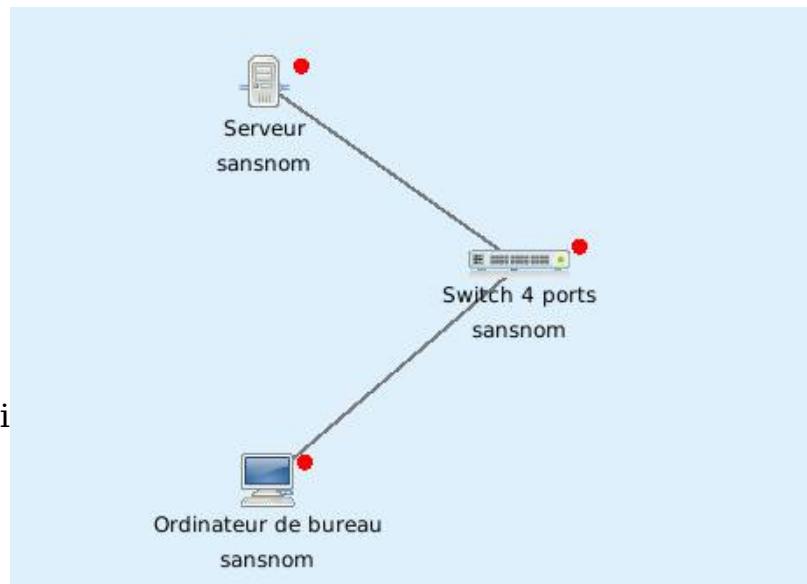


## **Mini-didacticiel pour commencer**

## Objectif

Ce didacticiel a été réalisé avec la version 1 du simulateur. L'aspect de certains composants et de l'interface a un peu changé, mais le fonctionnement reste le même.

Le but de ce didacticiel n'est pas de vous apprendre les principes des réseaux : ce qui suit est destiné à vous permettre de découvrir le fonctionnement de Network-In ! en concevant le réseau très simple ci-contre.



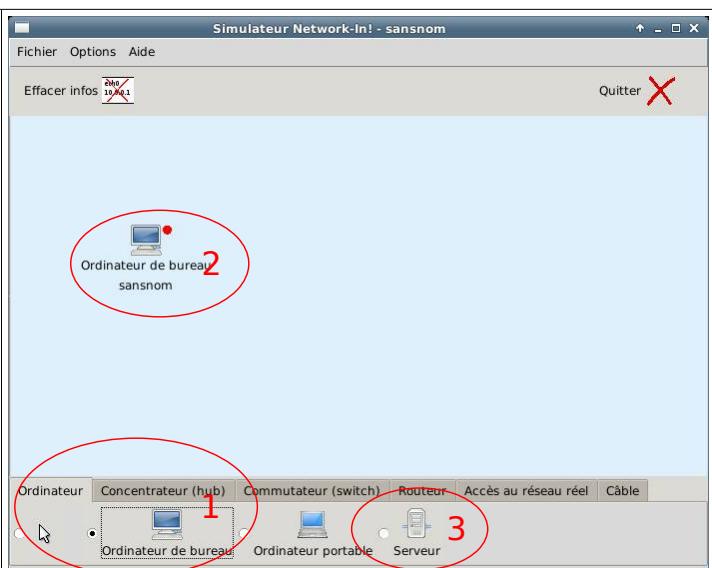
Ce réseau comprendra :

- Un ordinateur client appelé *pc1* ;
- Un ordinateur serveur Web appelé *servweb* ;
- Un switch pour les relier ensemble.

## Dessin du réseau

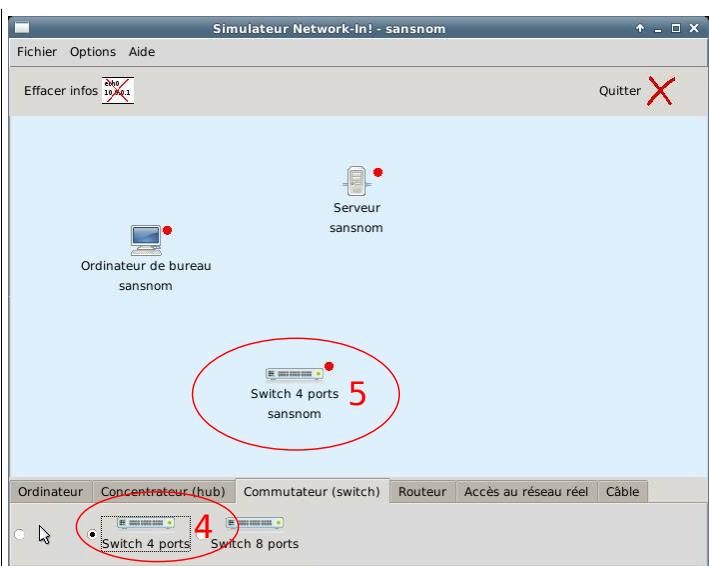
Commencez par démarrer Network-In ! depuis le menu d'applications.

1. Cliquez sur l'onglet *Ordinateur* puis choisissez *Ordinateur de bureau*.
2. Cliquez sur la zone de dessin de réseau. Un icône représentant un ordinateur de bureau apparaît.
3. De la même façon, placez un *ordinateur serveur*.



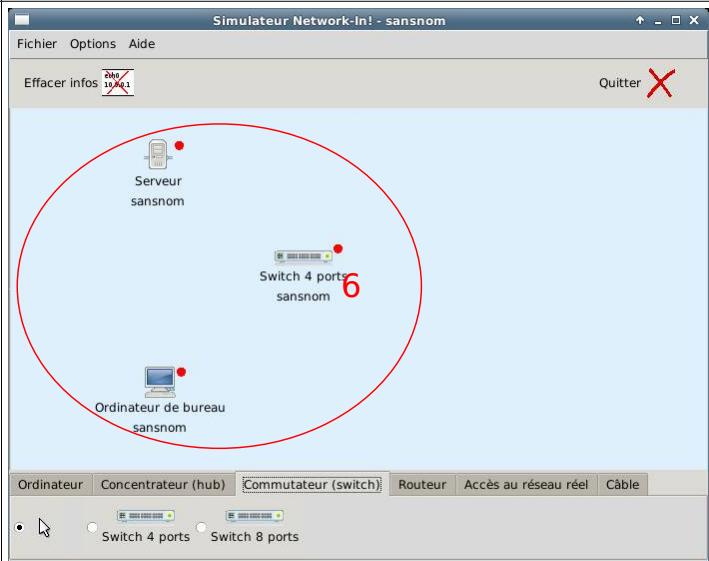
## Simulateur Network-In ! version 2.0

4. Cliquez sur l'onglet *Commutateur*.
5. Puis choisissez et placez un *switch à 4 ports*.



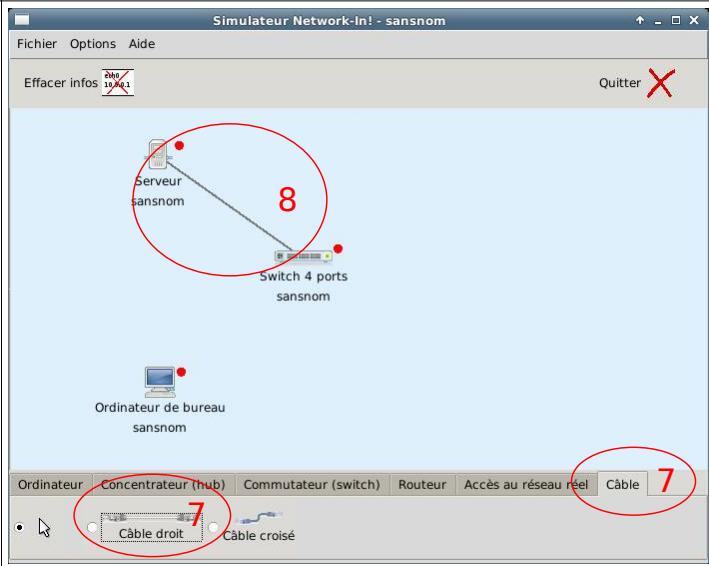
A tout moment, en cliquant sur un matériel et en glissant la souris, vous pouvez le déplacer où bon vous semble.

6. Disposez donc les trois appareils comme ceci.



Vous allez maintenant relier ensemble les trois matériels.

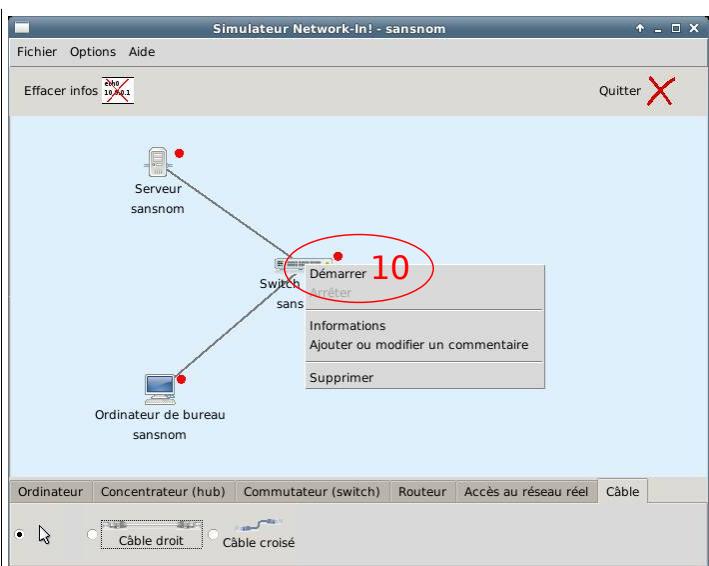
7. Cliquez sur l'onglet *Câble* puis choisissez un *câble droit*.
8. Cliquez sur le serveur puis sur le switch afin de les relier. On vous demande de choisir un port sur le switch. N'importe lequel convient.
9. Procédez de même pour relier l'ordinateur de bureau au switch.



## Simulateur Network-In ! version 2.0

Votre réseau est câblé. Vous allez pouvoir démarrer vos appareils !

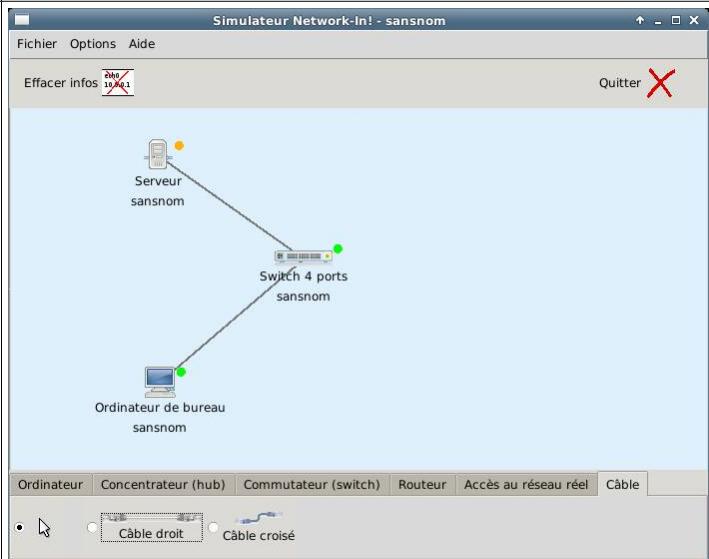
10. D'un clic droit sur le switch vous accédez à un menu contextuel.  
Choisissez *Démarrer* dans ce menu.
11. Faites de même pour les deux ordinateurs.



Le switch démarre presque immédiatement.

Les ordinateurs mettent un peu plus de temps à démarrer. Cela est tout à fait normal : il faut charger leur système d'exploitation.

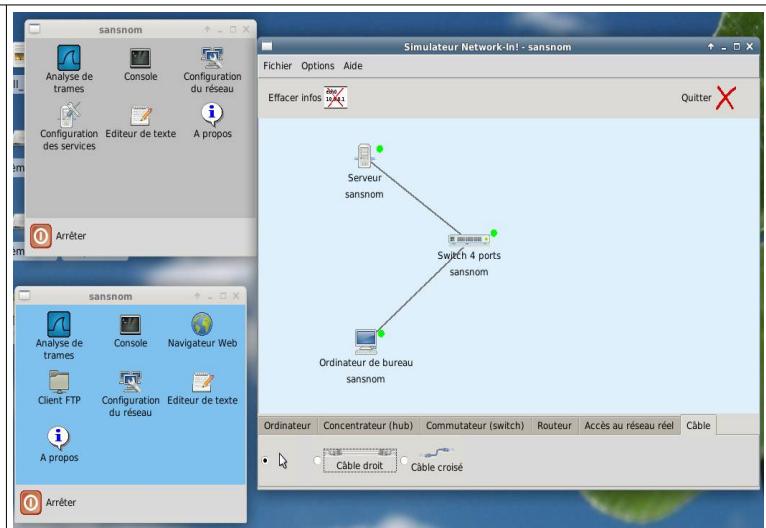
Vous remarquez qu'un matériel démarré est vert et qu'un matériel en cours de démarrage est orange.



Ca y est, votre première simulation est en cours d'exécution. Votre écran doit ressembler à peu près à cela :

## *Simulateur Network-In ! version 2.0*

- Les trois appareils sont démarrés.
- Le bureau de l'ordinateur serveur est affiché (fond d'écran gris).
- Le bureau de l'ordinateur de bureau est affiché (fond d'écran bleu ciel).



Si vous souhaitez quitter le logiciel, il est nécessaire d'arrêter les matériels qui sont en fonctionnement. Comme pour le démarrage, cela se fait à partir du menu contextuel de chaque matériel.

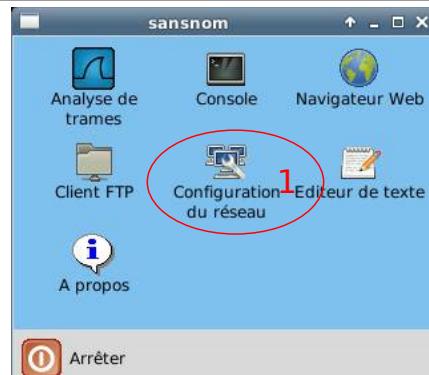
You n'avez pas besoin de sauvegarder avant de quitter le logiciel, la sauvegarde est automatique et vous retrouverez votre configuration à l'identique lors du prochain démarrage.

## **Configuration IP**

Les deux ordinateurs sont reliés entre eux par l'intermédiaire du switch, mais ils ne sont pas encore configurés pour communiquer ensemble.

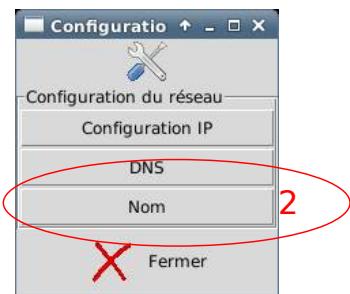
Il faut au minimum configurer leurs IP.

1. Cliquez sur l'icône *Configuration du réseau*, présent dans la fenêtre représentant le bureau de l'ordinateur.



## Simulateur Network-In ! version 2.0

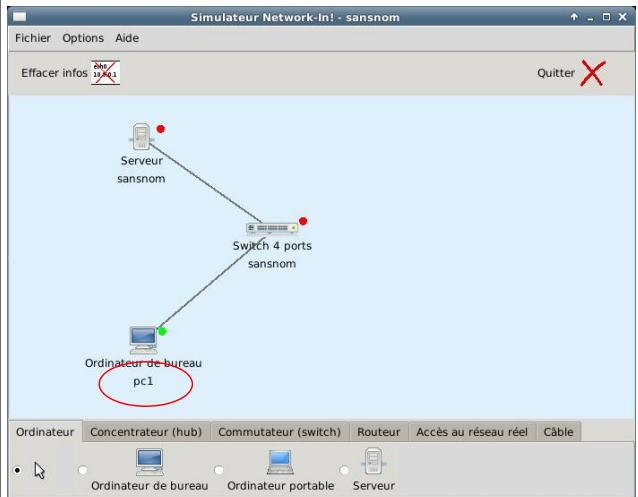
2. Dans un premier temps, cliquez sur le bouton *Nom*.



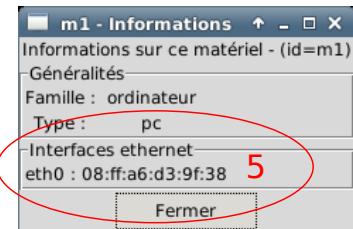
3. Donnez le nom *pc1* à votre ordinateur et cliquez sur valider.



Vous constatez que les informations se mettent à jour sur le schéma du réseau.



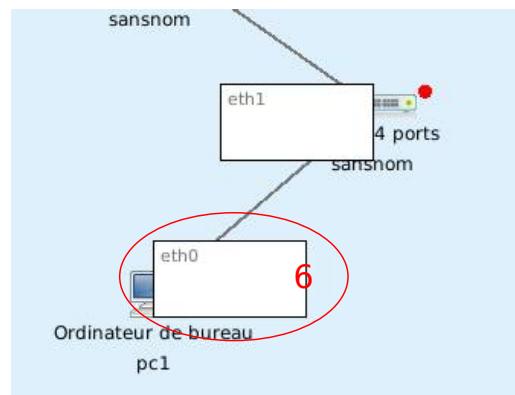
4. Faites apparaître le menu contextuel de *pc1* et choisissez *Informations*.  
 5. Une boîte apparaît et vous donne certains renseignements. Notamment, vous pouvez voir que l'IP n'est pas configurée car seule apparaît l'adresse MAC de l'interface.



## Simulateur Network-In ! version 2.0

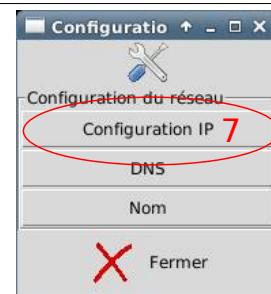
6. Vous pouvez obtenir ce même renseignement en cliquant sur le câble reliant l'ordinateur au switch. Des étiquettes vous informent sur la configuration de la connexion, pour chaque matériel relié par le câble.

En cliquant sur une des étiquettes, elles s'effacent automatiquement.



Revenez à l'interface de configuration de l'ordinateur *pc1*.

7. Cliquez maintenant sur *Configuration IP*.



8. Commencez par cocher la case *Interface active*.

9. Saisissez les informations suivantes :

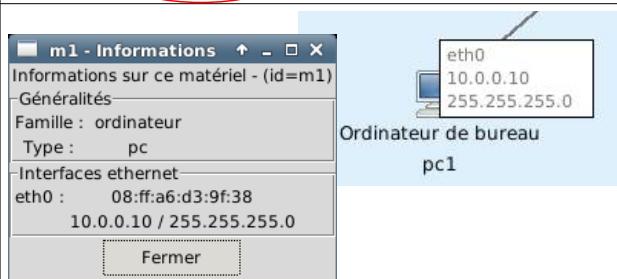
Adresse IP : 10.0.0.10

Masque : 255.255.255.0

Passerelle : 10.0.0.254

10. Puis terminez en activant le bouton *Valider*.

Si tout c'est bien passé, vous constatez que votre interface réseau est maintenant configurée et que les informations sont à jour sur le schéma du réseau.



## Simulateur Network-In ! version 2.0

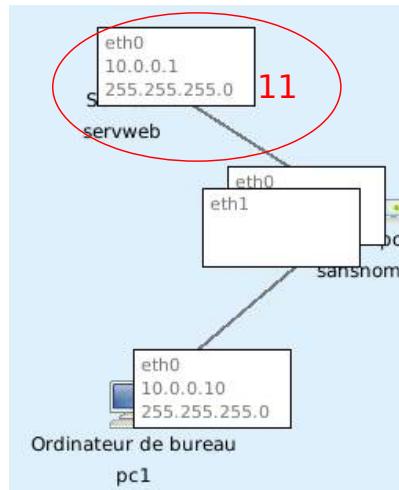
11. Procédez de la même façon pour configurer l'ordinateur serveur avec les paramètres suivants :

Nom : servweb

Adresse IP : 10.0.0.1

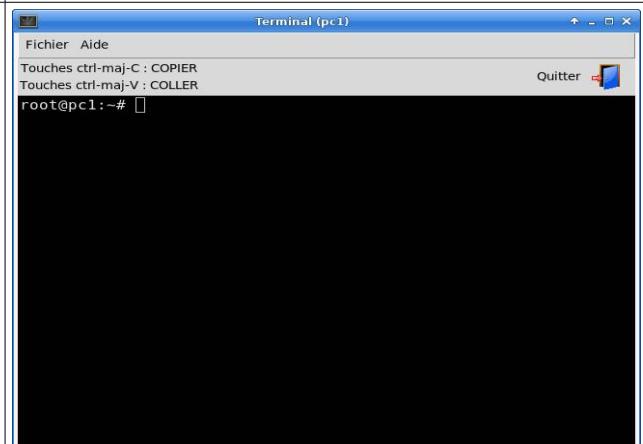
Masque : 255.255.255.0

Passerelle : 10.0.0.254



Vous allez maintenant vérifier que la connexion fonctionne, en envoyant des *ping* d'une machine à l'autre.

12. Commencez par faire apparaître un *terminal* sur *pc1*, en double-cliquant sur l'icône *Terminal admin* du bureau de *pc1*.



13. Vous pouvez vérifier que votre interface est convenablement configurée en tapant la commande **ifconfig** (commande Linux) ou **ipconfig** (commande Windows).

```

root@pc1:~# ifconfig
eth0      Link encap:Ethernet HWaddr 08:ff:a6:d3:9f:38
          inet addr:10.0.0.10 Bcast:10.0.0.255 Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
          Interrupt:5

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:452 (452.0 B) TX bytes:452 (452.0 B)

root@pc1:~#

```

## *Simulateur Network-In ! version 2.0*

14. Vous pouvez maintenant vérifier la connexion au serveur *servweb* en tapant la commande **ping** suivi de son adresse IP (10.0.0.10).

Si tout se passe correctement, *servweb* répond.

Pour arrêter l'envoi des ping, appuyez sur les touches **ctrl** et **c**.

15. Un bilan s'affiche alors.

16. Vous pouvez tenter l'expérience depuis *servweb* : contactez *pc1* depuis cet ordinateur.

N'arrêtez pas l'envoi des ping pour le moment.

Tentez une expérience :

17. Depuis le menu contextuel du switch, arrêtez celui-ci.

18. Le réseau ne fonctionne plus, vous le constatez car l'envoi des ping est désormais sans réponse.

Bien entendu, si vous redémarrez le switch, la connexion sera rétablie.

The terminal window shows the following output:

```
Interrupt:5
lo
Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
Rx bytes:452 (452.0 B) Tx bytes:452 (452.0 B)

root@pc1:~# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data. 14
64 bytes from 10.0.0.1: icmp_req=1 ttl=64 time=10.6 ms
64 bytes from 10.0.0.1: icmp_req=2 ttl=64 time=0.293 ms
64 bytes from 10.0.0.1: icmp_req=3 ttl=64 time=0.311 ms
64 bytes from 10.0.0.1: icmp_req=4 ttl=64 time=0.302 ms
64 bytes from 10.0.0.1: icmp_req=5 ttl=64 time=0.293 ms
64 bytes from 10.0.0.1: icmp_req=6 ttl=64 time=0.181 ms
64 bytes from 10.0.0.1: icmp_req=7 ttl=64 time=0.318 ms
^C
--- 10.0.0.1 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6063ms 15
rtt min/avg/max/mdev = 0.181/1.769/10.685/3.640 ms
root@pc1:~#
```

The terminal window shows the following output:

```
root@servweb:~# ping 10.0.0.10 16
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_seq=1 ttl=64 time=0.347 ms
64 bytes from 10.0.0.10: icmp_seq=2 ttl=64 time=0.279 ms
64 bytes from 10.0.0.10: icmp_seq=3 ttl=64 time=0.287 ms
|
```

The terminal window shows the following output:

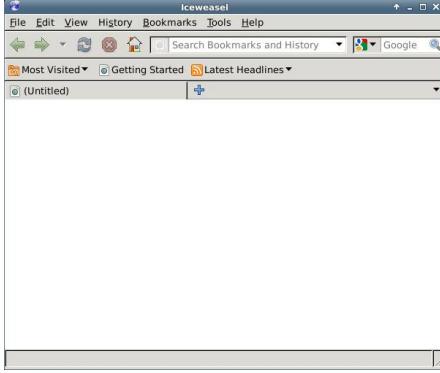
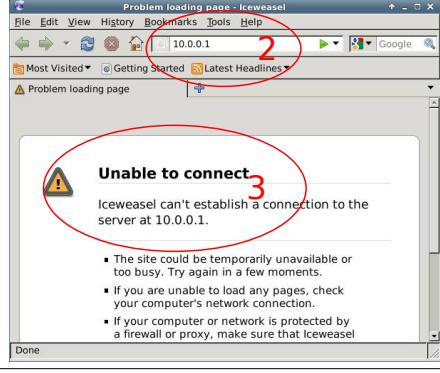
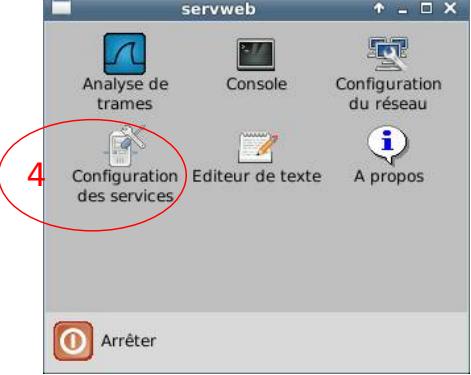
```
root@servweb:~# ping 10.0.0.10
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_seq=1 ttl=64 time=10.6 ms
64 bytes from 10.0.0.10: icmp_seq=2 ttl=64 time=0.275 ms
64 bytes from 10.0.0.10: icmp_seq=3 ttl=64 time=0.304 ms
64 bytes from 10.0.0.10: icmp_seq=4 ttl=64 time=0.311 ms
64 bytes from 10.0.0.10: icmp_seq=5 ttl=64 time=0.275 ms
From 10.0.0.1 icmp_seq=56 Destination Host Unreachable
From 10.0.0.1 icmp_seq=57 Destination Host Unreachable
From 10.0.0.1 icmp_seq=58 Destination Host Unreachable
From 10.0.0.1 icmp_seq=59 Destination Host Unreachable
From 10.0.0.1 icmp_seq=60 Destination Host Unreachable 18
From 10.0.0.1 icmp_seq=61 Destination Host Unreachable
From 10.0.0.1 icmp_seq=62 Destination Host Unreachable
From 10.0.0.1 icmp_seq=63 Destination Host Unreachable
From 10.0.0.1 icmp_seq=64 Destination Host Unreachable
|
```

## Configuration d'un service Web

Le mini-réseau actuel n'a pas grande utilité, puisque les ordinateurs communiquent, mais n'ont rien à échanger de particulier.

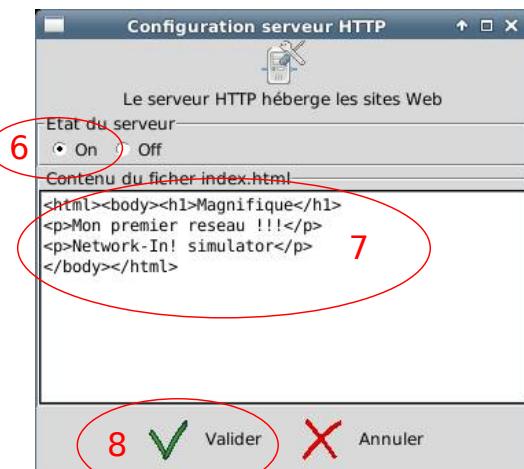
Vous allez maintenant configurer le serveur pour qu'il propose des pages Web à *pc1*.

## Simulateur Network-In ! version 2.0

<p>1. Depuis le bureau de <i>pc1</i>, démarrez le navigateur en cliquant sur l'icône <i>Navigateur Web</i> (patientez un peu le temps qu'il se charge).</p>	
<p>2. Dans la barre d'adresse, tapez l'adresse IP de <i>servweb</i> : 10.0.0.1</p> <p>3. Vous constatez que <i>servweb</i> ne propose pas de service Web pour le moment.</p>	
<p>4. Démarrer l'interface de configuration des services en double-cliquant sur l'icône <i>Configuration des services</i> du bureau de <i>servweb</i>.</p>	
<p>5. Cliquez maintenant sur le bouton <i>Serveur HTTP</i>, depuis la boîte qui vient d'apparaître.</p>	

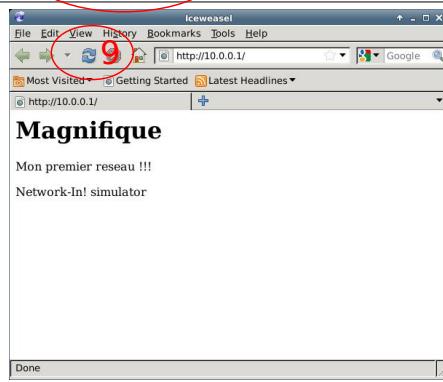
## Simulateur Network-In ! version 2.0

6. Demandez le démarrage du serveur en cliquant sur *On*.
7. Vous pouvez changer le contenu de la page d'accueil du site (fichier *index.html*).
8. A la fin appuyez sur le bouton *Valider*. Au bout de quelques secondes, le serveur est configuré.



9. Rechargez la page depuis le navigateur Web de *pc1*, vous obtenez désormais la page d'accueil servie par *servweb*.

Votre serveur est fonctionnel.



## **Références techniques**

## **Présentation technique**

### **Généralités**

Network-In! s'exécute nativement sous Linux.

Le logiciel est presque entièrement développé en Tcl/Tk. Notamment :

- L'interface centrale du logiciel (la zone de dessin du réseau) ;
- Toutes les interfaces des différents composants virtuels, ordinateurs, routeurs,...

Certaines parties du logiciel font appel en plus à des scripts Bash, notamment ce qui touche à la configuration des interfaces virtuelles "Tap" dont le simulateur a besoin dans son fonctionnement.

Les scripts Bash obtiennent les droits root nécessaires grâce à l'utilisation de l'utilitaire *sudo*, configuré pour Network-In! lors de son installation.

Des scripts Bash sont également installés sur les disques des machines virtuelles. Ils assurent la communication entre les machines virtuelles et le simulateur.

## **Matériels de type ordinateurs et routeurs**

Pour les composants de type ordinateurs et routeurs, de véritables instances de systèmes Linux sont démarrées.

### **User mode Linux**

La technique employée est celle du "User mode Linux" : le noyau est exécuté dans l'espace utilisateur, comme une simple application. En y adjoignant un disque virtuel comportant l'OS, on obtient de vraies machines Linux.

L'image du disque système est la même pour tous les composants (ordinateurs, routeurs) à l'exception de la machine "Linux texte".

Au premier démarrage d'une machine, un fichier COW (Copy On Write) est créé pour le nouveau matériel. Ce fichier enregistre les modifications opérées par rapport au disque système.

### ***Interface graphique***

Pour rendre la configuration des composants plus aisée, une interface graphique a été développée pour chaque matériel :

- un bureau pour les matériels de type ordinateur à interface graphique
- un écran avec 4 terminaux virtuels pour un ordinateur "Linux texte"
- une boîte donnant accès aux fonctionnalités pour les routeurs
- des boîtes de dialogue pour toutes les opérations de configuration

En version 2.0, les utilitaires suivants sont pré-installés dans les machines virtuelles ordinateurs :

## *Simulateur Network-In ! version 2.0*

- Le client FTP **Gftp**
- Le client en ligne de commande **ftp**
- L'utilitaire d'analyse **nmap**
- Le logiciel de capture de trames **tcpdump**
- Le logiciel de capture de trames **termshark**
- Le navigateur Web **Netsurf**
- L'émulateur de terminal **sterm**
- L'éditeur de texte **nano**

Certains utilitaires ont été conçus spécialement pour les machines de Network-In!:

- Un utilitaire de capture et d'affichage des captures utilisant Termshark
- Un terminal utilisant Sterm
- L'éditeur de texte dédié **Networkin-editor**

### **Ajout d'un noyau spécifique**

Dans certains cas, le noyau utilisé à la création du matériel n'est pas disponible. Le noyau installé par défaut est dans ce cas utilisé en remplacement automatiquement.

Si nécessaire, le noyau manquant peut être ajouté :

- Dans le répertoire `/usr/lib/network-in/kernels/` (droits administrateur nécessaires)
- Dans le répertoire `.network-in/kernels/` de son dossier personnel.

Le noyau doit être déposé dans un répertoire au nom de la version du noyau.

Attention, il s'agit d'un noyau spécialement compilé pour le user-mode-linux

Dans ce répertoire doit être présent le noyau lui-même nommé `linux.uml` et un disque virtuel `modules.img` contenant les modules compilés pour ce noyau.

Exemple :

```
ls -l .network-in/kernels/6.1.27/
total 163876
-rwxr-xr-x 1 vincent vincent 10516264 22 oct. 2023 linux.uml
-rw-r--r-- 1 vincent vincent 157286400 22 oct. 2023 modules.img
```

### **Ajout d'un disque système spécifique**

Si un matériel a été défini avec un disque système non disponible sur le simulateur sur lequel on souhaite démarrer ce matériel, le démarrage est impossible car le disque COW (Copy On Write) lié au matériel ne correspond à aucun disque système disponible installé.

Un disque système manquant peut être ajouté :

## *Simulateur Network-In ! version 2.0*

Dans le répertoire `/usr/lib/network-in/disks/` (droits administrateur nécessaires)

Dans le répertoire `.network-in/disks/` de son dossier personnel

Exemple :

```
ls -l .network-in/disks/  
total 965008  
-rw-r--r-- 1 vincent vincent 1610612736 6 janv. 2025 bookworm_1.01.img
```

Dans le répertoire `/usr/lib/network-in/disks/create_disk/` un ensemble de scripts permet de créer un nouveau disque système afin de disposer d'une configuration particulière

## **Switchs et hubs virtuels**

Les switchs et hubs utilisent un composant logiciel : les switch VDE version 2.

En version 2.0 de Network-In!, les switchs ne sont pas encore configurables, mais la technologie VDE permet de les manager, de créer des VLAN et de gérer le spanning tree. Cela sera implémenté dans une version future du simulateur.

Lien vers VDE2 : [http://wiki.virtualsquare.org/wiki/index.php/Main\\_Page](http://wiki.virtualsquare.org/wiki/index.php/Main_Page)

## **Câbles de connexion**

Les connexions entre matériels sont également fournies par VDE2.

## **Affichage des machines virtuelles**

Les machines virtuelles s'affichent dans le serveur X virtuel Xephyr. La fenêtre d'affichage de ce serveur est embarquée et affichée dans la zone de simulation de l'application.

Les machines exécutent un système différent du système hôte, mais affichent leurs fenêtres en utilisant le serveur Xephyr démarré sur l'hôte par le simulateur Network-In!. Cela nécessite d'établir une connexion TCP entre machines et hôte.

A cette fin chaque machine possède une interface ethernet nommée eth99 et une configuration IP automatiquement établie par le simulateur.

Côté hôte, une interface nommée `networkin_com` permet de communiquer avec l'ensemble des machines, par défaut en utilisant le réseau 172.31.255.1/24

## *Simulateur Network-In ! version 2.0*

### **Technique de masquage de l'IP de communication avec Xorg**

L'objectif est de masquer l'interface ethernet de communication eth99.

La solution qui est utilisée dans Network-In! est de travailler au niveau des commandes utilisateur.

La commande native est remplacée par un autre exécutable qui lui-même appelle la commande d'origine mais "nettoie" les affichages des informations sur les interfaces à masquer. Ainsi, l'utilisateur ne voit pas certaines interfaces pourtant nécessaires au bon fonctionnement du logiciel. Les commandes d'origine sont toujours disponibles, présentes dans */sbin*.

Le path par défaut des utilisateurs cherche les exécutables tout d'abord dans */sbin/network-in/* avant de chercher dans */sbin/*

Cette opération est appliquée aux commandes suivantes :

- ifconfig
- ip
- arp
- route
- netstat

### **Configuration pare-feu pour gérer le flux d'affichage**

Le pare-feu de l'hôte est automatiquement configuré au démarrage du simulateur pour à la fois permettre les communications des machines virtuelle vers le serveur X Xephyr, et interdire aux autres flux éventuels d'utiliser cette communication.

#### Exemple de paramétrage automatique du pare-feu :

```
root@pc-vv:~# iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target  prot opt in     out    source          destination
      0    0 ACCEPT   tcp  --  networkin_com any   anywhere       anywhere        tcp
dpt:x11-5
      0    0 ACCEPT   tcp  --  networkin_com any   anywhere       anywhere        tcp
dpt:x11
      0    0 DROP     all  --  networkin_com any   anywhere       anywhere
```

```
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
```

```
pkts bytes target  prot opt in     out    source          destination
      0    0 DROP     all  --  networkin_com any   anywhere       anywhere
```

```
Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
```

```
pkts bytes target  prot opt in     out    source          destination
```

## **Gestion des VM VirtualBox avec le simulateur**

Les machines virtuelles qui sont intégrées à une simulation sont pilotées et configurées en utilisant l'utilitaire **vboxmanage** qui est fourni lors de l'installation de VirtualBox.

La connexion réseau entre la carte ethernet de la VM VirtualBox et le simulateur est assurée par un composant fourni par la suite VDE2 au travers de la bibliothèque *libvdeplug.so*

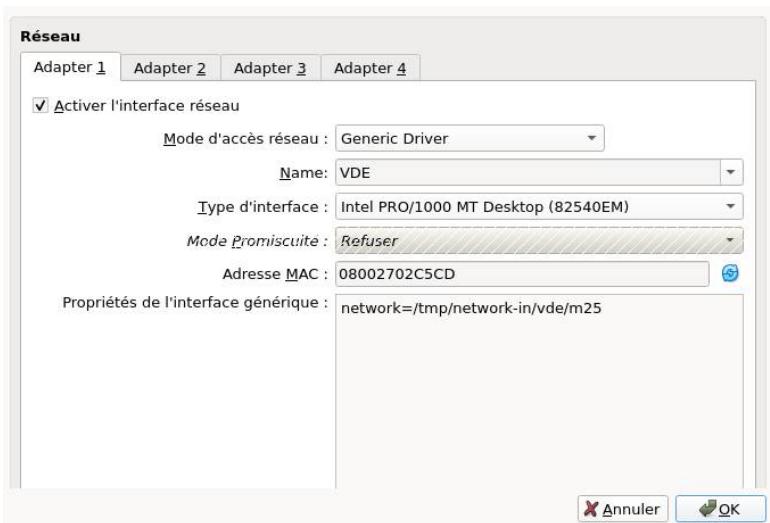


Figure 5: Configuration de l'interface réseau de la VM vue depuis VirtualBox

## **Portage sous Windows**

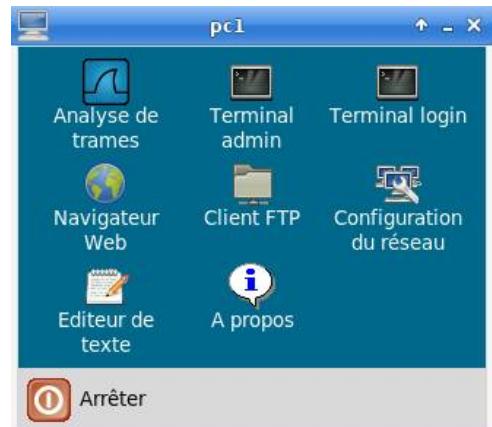
Network-In! est disponible sous Windows en utilisant la technologie WSL (Windows System for Linux) qui permet d'exécuter des machines virtuelles Linux de manière transparente.

Il s'agit d'un artifice qui limite les possibilités du simulateur :

- Actuellement les maquettes de simulations qui utilisent des machines virtuelles VirtualBox externes ne peuvent les faire fonctionner.
- De même il n'est pas possible d'offrir des interfaces en mode pont avec la machine hôte.

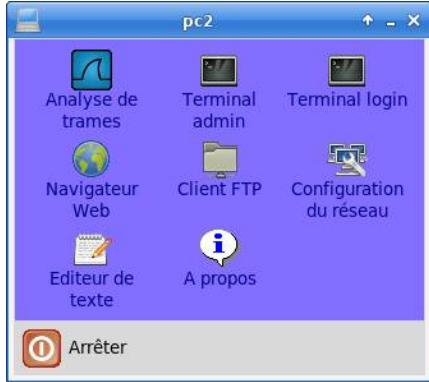
## **Caractéristiques des composants**

### **Ordinateur PC**

Nom interne du composant	desktop
Description	Ordinateur client standard
Type de matériel	DTE
Technologie employée	Noyau Usermode-Linux Exécution système Debian
Persistance de la configuration	Disque en Copy-On-Write
Mémoire vive allouée	512Mo
Interface utilisateur	De type bureau avec icônes de lancement des applications 
Connectivité réseau	1 interface ethernet : eth0 Ajout possible d'autres interfaces
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Configuration du réseau</li> <li>• Navigateur Web (Netsurf)</li> <li>• Client FTP (Gftp)</li> <li>• Terminal de commandes (basé sur Simple Term)</li> <li>• Capture de trames (basé sur Termshark)</li> <li>• Editeur de texte (Network-In!)</li> </ul>

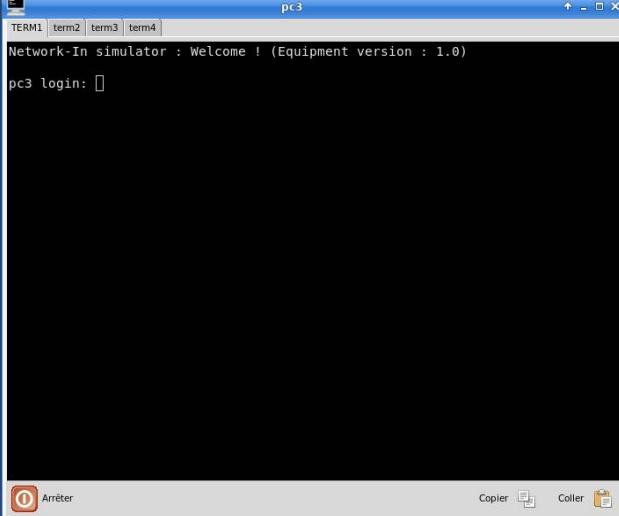
## Simulateur Network-In ! version 2.0

### Ordinateur portable

Nom interne du composant	laptop
Description	Ordinateur standard (portable). Pas de possibilité d'ajout de carte réseau ni autre reconfiguration.
Type de matériel	DTE
Technologie employée	Noyau Usermode-Linux Exécution système Debian
Persistance de la configuration	Disque en Copy-On-Write
Mémoire vive allouée	512Mo
Interface utilisateur	De type bureau avec icônes de lancement des applications  
Connectivité réseau	1 interface ethernet : eth0 Pas de modification possible
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Configuration du réseau</li> <li>• Navigateur Web (Netsurf)</li> <li>• Client FTP (Gftp)</li> <li>• Terminal de commandes (basé sur Simple Term)</li> <li>• Capture de trames (basé sur Termshark)</li> <li>• Editeur de texte (Network-In!)</li> </ul>

## Simulateur Network-In ! version 2.0

### Ordinateur Linux mode texte

Nom interne du composant	linux
Description	Ordinateur Linux en mode CLI destiné à l'apprentissage du système Linux : toute la configuration doit être faite en ligne de commande
Type de matériel	DTE
Technologie employée	Noyau Usermode-Linux Exécution système Debian
Persistance de la configuration	Disque en Copy-On-Write
Mémoire vive allouée	256Mo
Interface utilisateur	Mode CLI offrant plusieurs terminaux virtuels 
Connectivité réseau	1 interface ethernet : eth0 Ajout possible d'autres interfaces
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Commandes Linux usuelles</li> <li>• tcpdump</li> <li>• ssh (client)</li> </ul>
Compte administrateur	root Mot de passe par défaut : root

## Simulateur Network-In ! version 2.0

### Ordinateur serveur

Nom interne du composant	server
Description	Ordinateur serveur offrant des services de base
Type de matériel	DTE
Technologie employée	Noyau Usermode-Linux Exécution système Debian
Persistance de la configuration	Disque en Copy-On-Write
Mémoire vive allouée	512Mo
Interface utilisateur	<p>De type bureau avec icônes de lancement des applications</p> 
Connectivité réseau	<p>1 interface ethernet : eth0</p> <p>Ajout possible d'autres interfaces</p>
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Configuration du réseau</li> <li>• Configuration des services</li> <li>• Client FTP (Gftp)</li> <li>• Terminal de commandes (basé sur Simple Term)</li> <li>• Capture de trames (basé sur Termshark)</li> <li>• Editeur de texte (Network-In!)</li> </ul>
Services disponibles	<ul style="list-style-type: none"> <li>• Service DHCP (isc-dhcp-server)</li> <li>• Service DNS (Bind9)</li> <li>• Service FTP (Proftpd)</li> <li>• Service HTTP (Apache 2)</li> <li>• Service SSH (OpenSSH)</li> </ul>

## Simulateur Network-In ! version 2.0

### Routeur

Nom interne du composant	router2 (2 interfaces ethernet) router4 (4 interfaces ethernet)
Description	Routeur proposant 2 ou 4 interfaces ethernet. Routage statique exclusivement.
Type de matériel	DTE
Technologie employée	Noyau Usermode-Linux Exécution système Debian
Persistance de la configuration	Disque en Copy-On-Write
Mémoire vive allouée	256Mo
Interface utilisateur	Différents boutons permettant la configuration du routeur  
Connectivité réseau	2 ou 4 interfaces ethernet : eth0, eth1, eth2, eth3  Pas de modification possible
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Configuration IP</li> <li>• Configuration du nom</li> <li>• Configuration DNS</li> <li>• Routage statique</li> <li>• Affichage table de routage</li> <li>• Terminal de commandes (basé sur Simple Term)</li> </ul>

## Simulateur Network-In ! version 2.0

### **Routeur NAT réseau réel**

Nom interne du composant	nat
Description	Routeur proposant 1 interface ethernet côté simulateur et connecté au réseau de l'hôte par un routage NAT/Masquerading Remarque : le routeur bloque le <i>ping</i> dans les 2 sens
Type de matériel	DTE
Technologie employée	SlirpVDE
Persistance de la configuration	dans fichier structure.xml
Mémoire vive allouée	~0
Interface utilisateur	Différents boutons permettant la configuration du routeur  
Connectivité réseau	1 interface ethernet côté simulateur : eth0 Pas de modification possible
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Configuration IP</li> <li>• Configuration du nom</li> </ul>

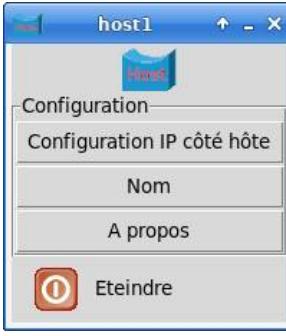
## Simulateur Network-In ! version 2.0

### **Machine virtuelle VirtualBox**

Nom interne du composant	virtualbox
Description	Composant qui permet de connecter une VM VirtualBox au simulateur
Type de matériel	DTE
Technologie employée	VirtualBox Connecteur libvdeplug.so (VDE2)
Persistance de la configuration	dans fichier structure.xml
Mémoire vive allouée	Mémoire VM VirtualBox
Interface utilisateur	Différents boutons permettant la configuration de l'intégration de la VM VirtualBox 
Connectivité réseau	1 interface ethernet côté simulateur : eth0 Interface connectée à l'une des interfaces de la VM par l'utilisation d'un composant VDE2 <sup>38</sup>
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>• Configuration du nom</li> <li>• Choix de la VM à connecter</li> <li>• Choix de l'interface ethernet à relier à la simulation</li> </ul>

## Simulateur Network-In ! version 2.0

### Pont avec l'hôte

Nom interne du composant	bridge
Description	Composant qui permet de connecter une VM VirtualBox au simulateur
Type de matériel	DTE
Technologie employée	Interface TAP
Persistance de la configuration	dans fichier structure.xml
Mémoire vive allouée	~0
Interface utilisateur	<p>Différents boutons permettant la configuration du pont</p> 
Connectivité réseau	<ul style="list-style-type: none"> <li>1 port ethernet côté simulateur : port 0</li> <li>1 interface TAP pour l'hôte : eth_mXX</li> </ul>
Applications / utilitaires disponibles	<ul style="list-style-type: none"> <li>Configuration du nom</li> <li>Configuration IP sur l'hôte</li> </ul>

## *Simulateur Network-In ! version 2.0*

### **Commutateur non administrable**

Nom interne du composant	switch
Description	Composant switch non administrable
Type de matériel	DCE
Technologie employée	VDE2 switch
Persistance de la configuration	dans fichier structure.xml
Mémoire vive allouée	~0
Interface utilisateur	Aucune
Connectivité réseau	<ul style="list-style-type: none"><li>• 2 ou 4 ports ethernet : port0 à port3 ou port7</li></ul>
Applications / utilitaires disponibles	<ul style="list-style-type: none"><li>• spanning tree activé</li></ul>

### **Concentrateur**

Nom interne du composant	hub
Description	Composant hub
Type de matériel	DCE
Technologie employée	VDE2 switch
Persistance de la configuration	dans fichier structure.xml
Mémoire vive allouée	~0
Interface utilisateur	Aucune
Connectivité réseau	<ul style="list-style-type: none"><li>• 2 ou 4 ports ethernet : port0 à port3 ou port7</li></ul>
Applications / utilitaires disponibles	<ul style="list-style-type: none"><li>• Aucune</li></ul>