

Manuel d'installation et d'utilisation de Network-In ! version 2.1.x



Table des matières

PRÉSENTATION ET INSTALLATION.....	5
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
MISE EN ŒUVRE DU SIMULATEUR.....	12
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
Paramétrer le matériel.....	17

Simulateur Network-In ! version 2.1.x

Interface graphique d'un matériel.....	17
Informations.....	18
Informations sur un matériel.....	18
Informations sur une connexion.....	18
Commentaires associés à un matériel.....	18
Outil de capture de trames intégré.....	19
Mise en œuvre d'une capture.....	19
Aspects techniques des captures.....	21
Niveaux de l'interface du simulateur.....	21
Interface pour débutant.....	22
Interface complète.....	22
Problèmes rencontrés.....	22
Noyau manquant sur un ordinateur ou routeur.....	22
Disque système manquant sur un ordinateur ou routeur.....	23
Matériel VM VirtualBox indisponible.....	23
RÉFÉRENCES TECHNIQUES.....	24
Présentation technique.....	25
Généralités.....	25
Matériels de type ordinateurs et routeurs.....	25
User mode Linux.....	25
Interface graphique.....	25
Ajout d'un noyau spécifique.....	26
Ajout d'un disque système spécifique.....	26
Switchs et hubs virtuels.....	27
Câbles de connexion.....	27
Affichage des machines virtuelles.....	27
Technique de masquage de l'IP de communication avec Xorg.....	28
Configuration pare-feu pour gérer le flux d'affichage.....	28
Gestion des VM VirtualBox avec le simulateur.....	29
Portage sous Windows.....	29
Caractéristiques des composants.....	30
Ordinateur PC.....	30
Ordinateur portable.....	31
Ordinateur Linux mode texte.....	32

Simulateur Network-In ! version 2.1.x

Ordinateur serveur.....	33
Routeur.....	34
Routeur NAT réseau réel.....	35
Machine virtuelle VirtualBox.....	36
Pont avec l'hôte.....	37
Commutateur non administrable.....	38
Concentrateur.....	38
Commutateur administrable.....	39

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).

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 2.1.x

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), avec les mêmes inconvénients 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) , lsof , 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 2.1.x

Paramètre	usage	valeurs typiques
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 2.1.x

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 :

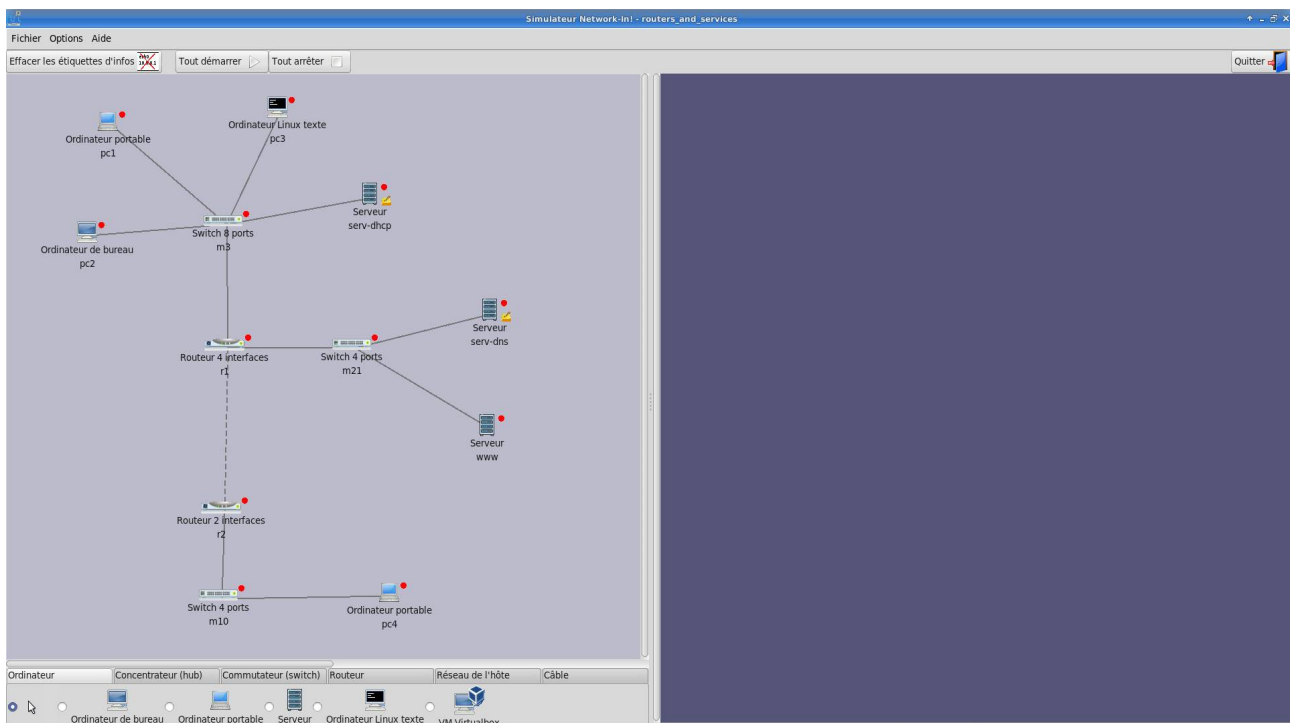
- le répertoire du projet actif *network-in-proj/*
- le répertoire de configuration personnelle de l'utilisateur *.network-in/*

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 2.1.x

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

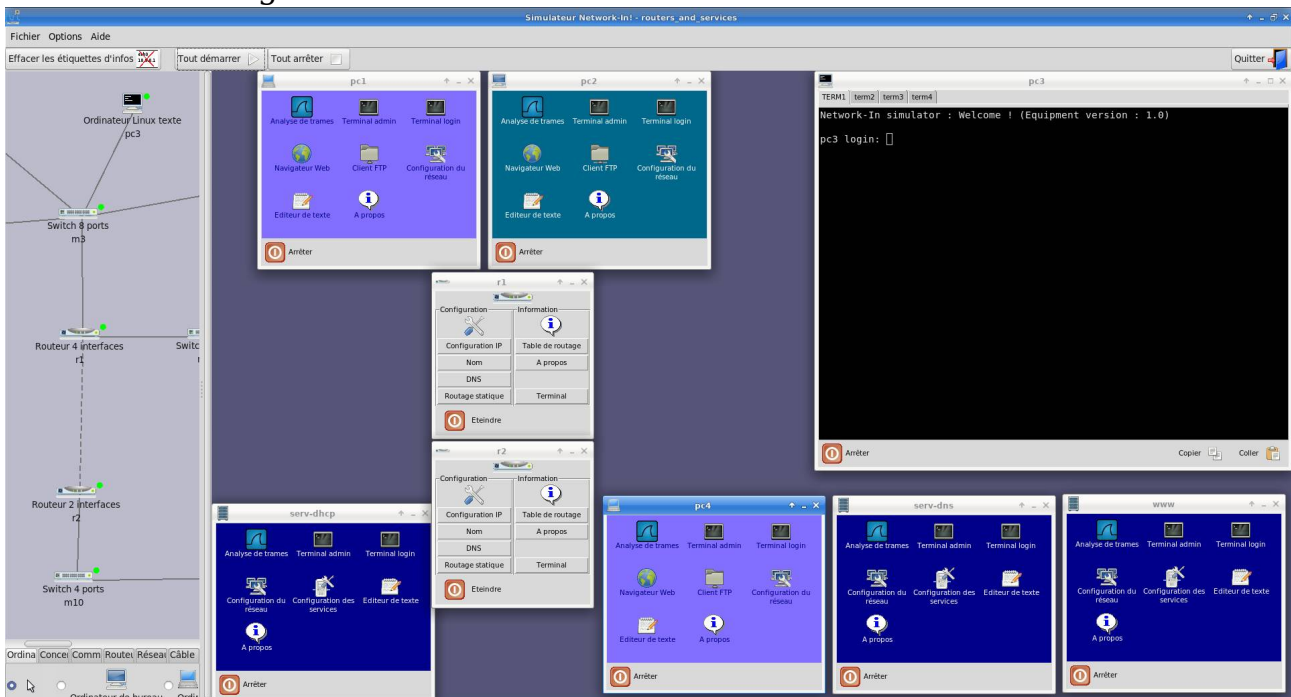


Figure 1: Simulation en cours. La fenêtre de simulation a été agrandie

Démarrage d'un matériel

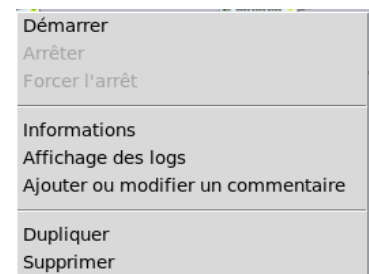
Chaque matériel peut être démarré manuellement.

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.

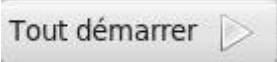
Un matériel a trois états possibles. Il peut être :

- 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

Démarrage de l'ensemble de la simulation

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

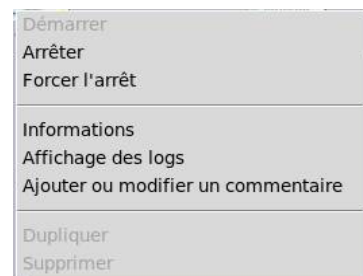
Si VirtualBox n'est pas installé ou que la VM VirtualBox n'est pas accessible, alors le composant ne démarre pas mais aucune erreur n'est déclenchée.

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 virtuelle 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.

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 switches non administrables ;
- Des switches 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 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.



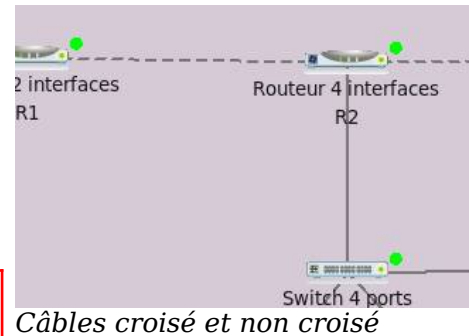
Onglets de choix de matériel

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.



Paramétrer 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.

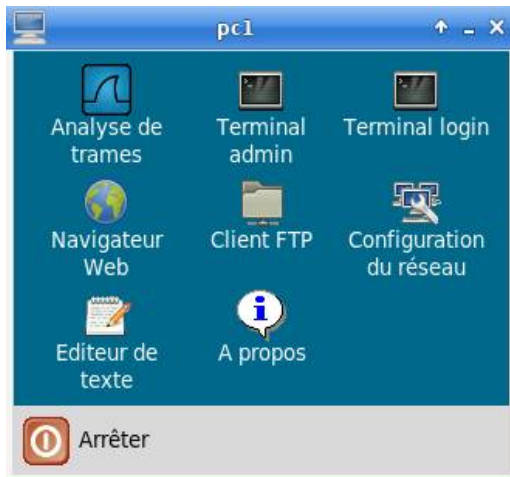
Certains composants ne sont pas configurables et ne proposent pas d'interface graphique :

- 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 2.1.x



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 aux deux extrémités :

- Le nom 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.

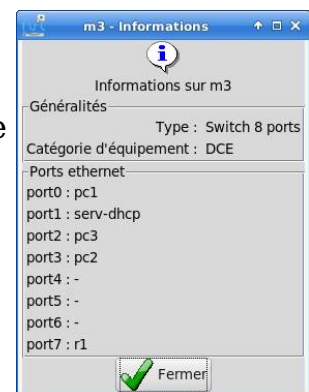


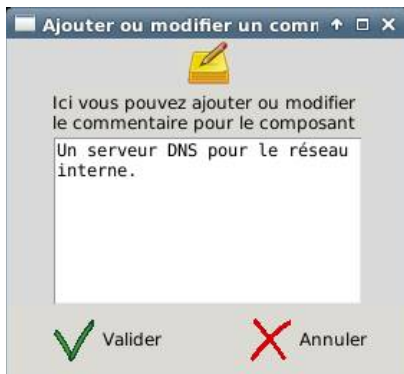
Figure 2: Fenêtre d'information (ici un switch)

Commentaires associés à un matériel

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

Simulateur Network-In ! version 2.1.x

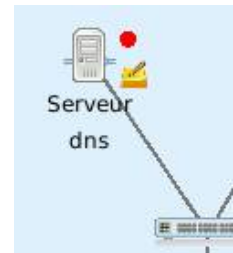
En cliquant sur cet icône, on pourra visualiser la note associée au matériel :



Fenêtre de saisie



Visualisation du commentaire



Présence d'un commentaire

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.

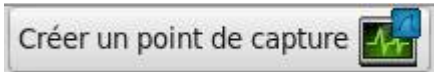
Pour supprimer un commentaire, effacer son texte depuis la fenêtre de saisie.

Outil de capture de trames intégré

Network-In! permet de visualiser très facilement le trafic réseau dans n'importe quel câble de la simulation.

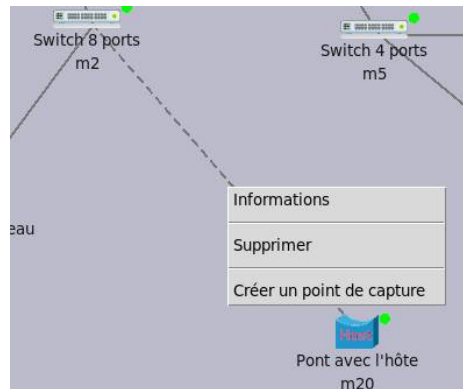
Mise en œuvre d'une capture

Il faut d'abord définir un point de capture. 2 méthodes sont possibles :

En cliquant sur le bouton  de la barre de boutons puis sélectionnant un câble.

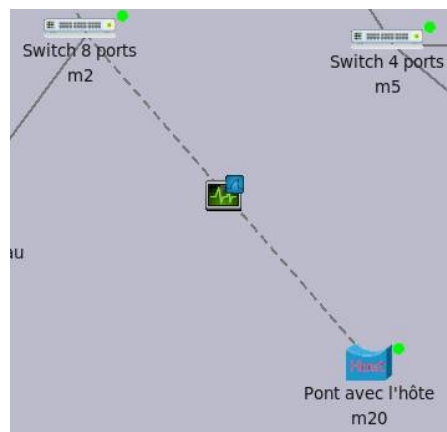
En cliquant avec le bouton droit sur un câble et en utilisant le menu "Créer un point de capture".

Simulateur Network-In ! version 2.1.x

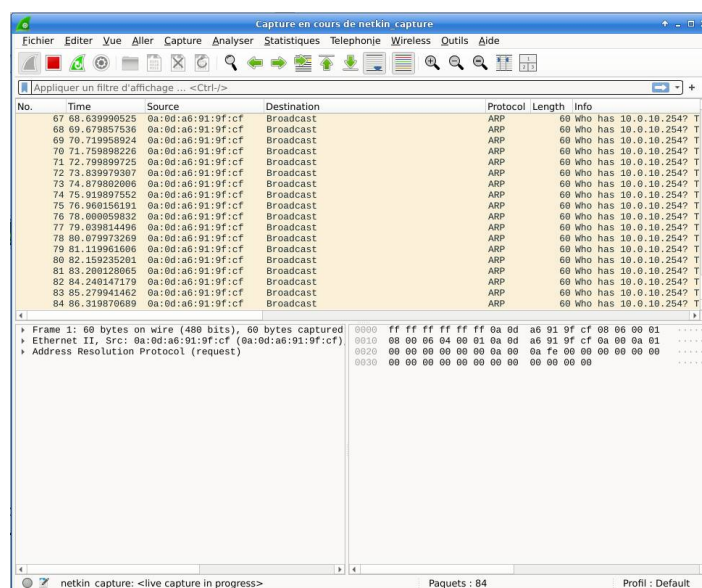


Créer un point de capture depuis le menu contextuel d'un câble

Dans les deux cas, le point est créé et matérialisé au milieu du câble par l'icône .



Point de capture créé



Une capture démarre automatiquement

L'utilitaire de capture Wireshark s'ouvre automatiquement et la capture

Simulateur Network-In ! version 2.1.x

démarre.

Il est possible de créer des points de capture sur plusieurs câbles en même temps.

Il est possible de démarrer autant de fenêtres de captures que nécessaire (depuis le menu contextuel de l'icône de capture), cela permet de faire des comparaisons de manière simple.

La suppression d'un point de capture (depuis le menu contextuel de l'icône de capture) ne supprime pas les fenêtres Wireshark des captures.

Aspects techniques des captures

Une interface virtuelle est créée sur la machine hôte exécutant Network-In!. Par défaut cette interface est nommée **netkin_cap_mXX** (XX est le numéro de câble).

L'outil Wireshark, pour avoir accès aux interfaces réseau du système et donc en particulier à cette interface, doit être exécuté par un compte système possédant les autorisations nécessaires : pour cela, lors de l'installation, Wireshark (lib_pcap exactement) est configuré automatiquement pour permettre les captures par n'importe quel utilisateur (configuration des capabilities). Cela pourrait être considéré comme une faiblesse de sécurité.

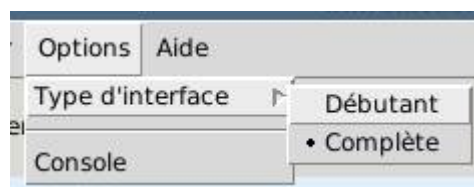
Niveaux de l'interface du simulateur

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.



Menu de choix de l'interface

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

Simulateur Network-In ! version 2.1.x

Interface pour débutant

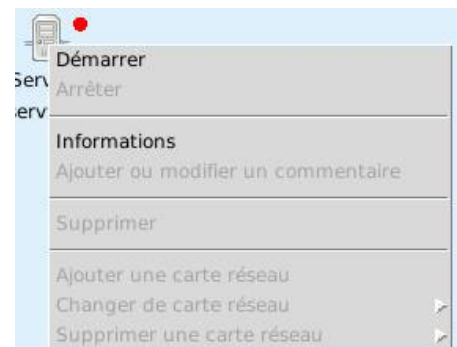
Elle correspond au niveau 1 dans le fichier de configuration du logiciel.

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



Interface débutant : certains onglets et composants ont disparus

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.



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

Interface complète

Elle correspond au niveau 2 dans le fichier de configuration du logiciel.

Toutes les fonctionnalités du logiciel sont utilisables.



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 2.1.x



Figure 3: Noyau manquant

Se référer aux explications fournies en annexe²⁶ 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 annexe²⁶ 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 grisées 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 grisée mais présente un point d'interrogation.



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 script 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 opérée 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 2.1.x

- 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 **stern**
- 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 Stern
- 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 2.1.x

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

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`

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*

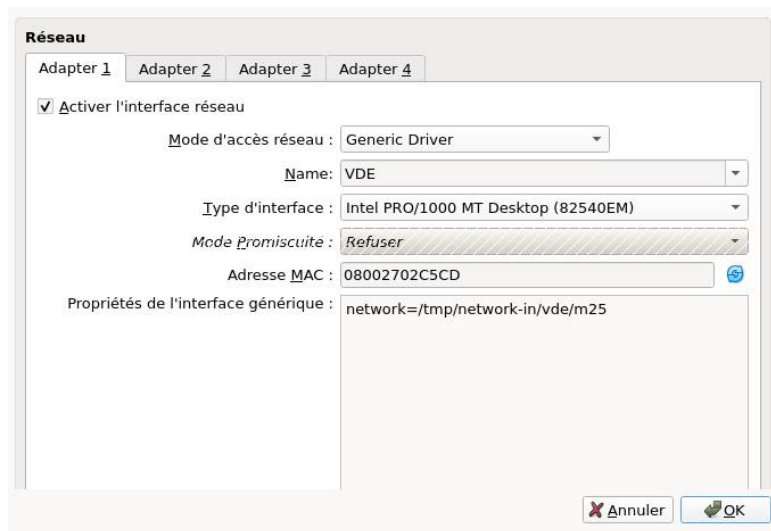


Figure 5: Configuration de l'interface réseau de la VM vue depuis VirtualBox

Portage sous Windows

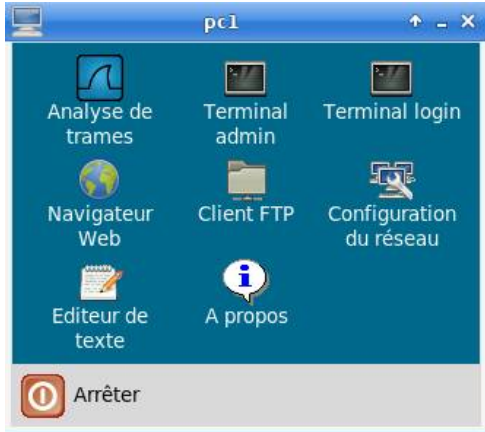
Network-In! est disponible sous Windows en utilisant la technologie WSL (Windows System for Linux) qui permet d'exécuter des machines virtuelles Linux de manière transparente.

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	• Configuration du réseau • Navigateur Web (Netsurf) • Client FTP (Gftp) • Terminal de commandes (basé sur Simple Term) • Capture de trames (basé sur Termshark) • Editeur de texte (Network-In!)

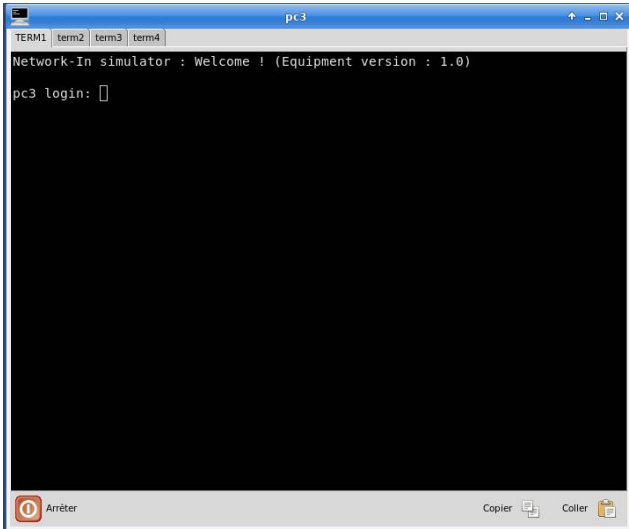
Simulateur Network-In ! version 2.1.x

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	• Configuration du réseau • Navigateur Web (Netsurf) • Client FTP (Gftp) • Terminal de commandes (basé sur Simple Term) • Capture de trames (basé sur Termshark) • Editeur de texte (Network-In!)

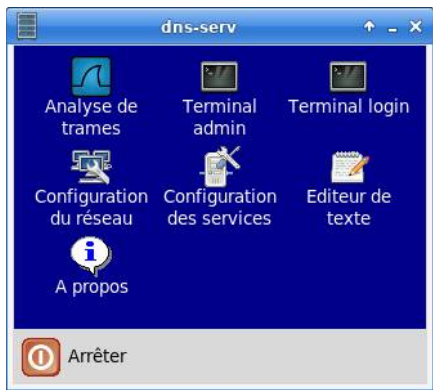
Simulateur Network-In ! version 2.1.x

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	• Commandes Linux usuelles • tcpdump • ssh (client)
Compte administrateur	root Mot de passe par défaut : root

Simulateur Network-In ! version 2.1.x

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	De type bureau avec icônes de lancement des applications 
Connectivité réseau	1 interface ethernet : eth0 Ajout possible d'autres interfaces
Applications / utilitaires disponibles	• Configuration du réseau • Configuration des services • Client FTP (Gftp) • Terminal de commandes (basé sur Simple Term) • Capture de trames (basé sur Termshark) • Editeur de texte (Network-In!)
Services disponibles	• Service DHCP (isc-dhcp-server) • Service DNS (Bind9) • Service FTP (Proftpd) • Service HTTP (Apache 2) • Service SSH (OpenSSH)

Simulateur Network-In ! version 2.1.x

Routeur

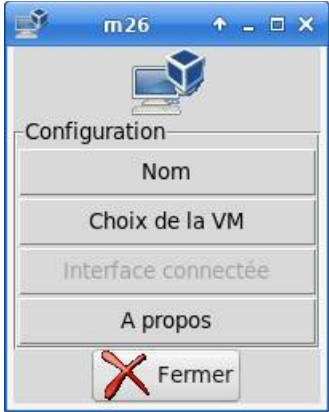
Nom interne du composant	router2 (2 interfaces ethernet) router4 (4 interfaces ethernet)
Description	Routeur proposant 2 ou 4 interfaces ethernet. Routage statique exclusivement.
Type de matériel	DTE
Technologie employée	Noyau Usermode-Linux Exécution système Debian
Persistance de la configuration	Disque 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	• Configuration IP • Configuration du nom • Configuration DNS • Routage statique • Affichage table de routage • Terminal de commandes (basé sur Simple Term)

Simulateur Network-In ! version 2.1.x

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	• Configuration IP • Configuration du nom

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 ²⁹
Applications / utilitaires disponibles	• Configuration du nom • Choix de la VM à connecter • Choix de l'interface ethernet à relier à la simulation

Simulateur Network-In ! version 2.1.x

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	Différents boutons permettant la configuration du pont 
Connectivité réseau	• 1 port ethernet côté simulateur : port 0 • 1 interface TAP pour l'hôte : eth_mXX
Applications / utilitaires disponibles	• Configuration du nom • Configuration IP sur l'hôte

Commutateur non administrable

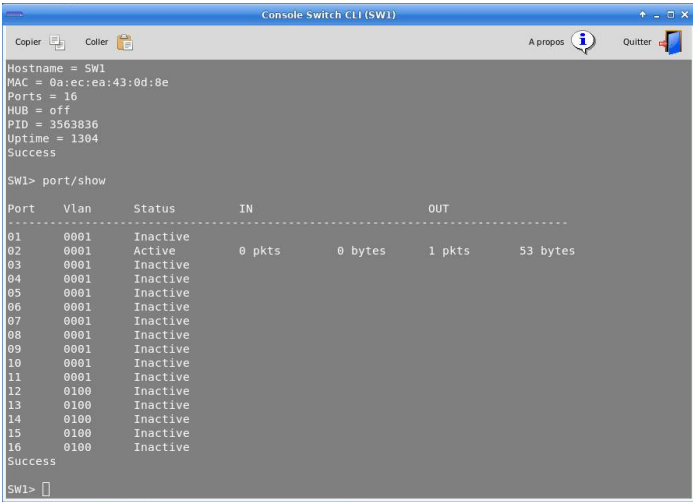
Nom interne du composant	switch
Description	Composant switch non administrable
Type de matériel	DCE
Technologie employée	VDE2 switch
Persistance de la configuration	dans fichier structure.xml
Mémoire vive allouée	~0
Interface utilisateur	Aucune
Connectivité réseau	• 2 ou 4 ports ethernet : port0 à port3 ou port7
Applications / utilitaires disponibles	• Aucune

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	• 2 ou 4 ports ethernet : port1 à port4 ou port8
Applications / utilitaires disponibles	• Aucune

Simulateur Network-In ! version 2.1.x

Commutateur administrable

Nom interne du composant	mswitch
Description	Composant switch administrable
Type de matériel	DCE
Technologie employée	VDE2 switch
Persistance de la configuration	Dans fichier structure.xml pour les connexions des ports et l'@ MAC Dans fichier network-in-proj/datas/mxx/init.conf pour la configuration du switch
Mémoire vive allouée	~0
Interface utilisateur	Interface en mode CLI donnant accès aux commandes de configuration du switch.  <pre> Console Switch CLI (SW1) ----- Copier Coller A propos Quitter Hostname = SW1 MAC = 0a:ec:ea:43:0d:8e Ports = 16 HUB = off PID = 3563836 Uptime = 1304 Success SW1> port/show Port Vlan Status IN OUT ----- 01 0001 Inactive 02 0001 Active 0 pkts 0 bytes 1 pkts 53 bytes 03 0001 Inactive 04 0001 Inactive 05 0001 Inactive 06 0001 Inactive 07 0001 Inactive 08 0001 Inactive 09 0001 Inactive 10 0001 Inactive 11 0001 Inactive 12 0100 Inactive 13 0100 Inactive 14 0100 Inactive 15 0100 Inactive 16 0100 Inactive Success SW1> </pre>
Connectivité réseau	16 ports ethernet : port1 à port16
Applications / utilitaires disponibles	- Configuration spanning tree - Configuration VLAN - Table MAC - Sauvegarde et chargement des configurations du switch

Simulateur Network-In ! version 2.1.x

Commandes disponibles depuis la console CLI d'un switch administrable :

La commande **help** ou **?** donne accès à l'ensemble des commande et leurs explications.

COMMAND PATH	ARGUMENTS	HELP
-----	-----	-----
?	[arg]	Help (limited to arg when specified)
help	[arg]	Help (limited to arg when specified)
hostname	NAME	set switch name
load	PATH	load a configuration script
logout		logout from this mgmt terminal
save	[PATH]	save configuration in file (default file
path if not completed)		
sethub	off(default)/on	enable or disable hub mode
show		show switch version and more info
shutdown		shutdown of the switch
mac	=====	HASH TABLE MENU
mac/setexpire	N	change MAC entries expire time
mac/show		show MAC table
stp	=====	RAPID SPANNING TREE MENU
stp/disable		disable STP
stp/enable		enable STP
stp/portcost	VLAN PORT COST	set the port cost for a VLAN
stp/portfast	VLAN PORT on/off	Define or cancel an edge port for a vlan
stp/priority	PRIORITY	set switch priority (value between 0 and
61440		
stp/show	[VLAN]	show spanning tree data for the defined vlan
port	=====	PORT STATUS MENU
port/resetcounter	[PORT]	reset the port (PORT) counters
port/setvlan	PORT VLAN	set port VLAN (untagged)
port/show	[PORT]	print the port table
vlan	=====	VLAN MANAGEMENT MENU
vlan/addtrunkport	VLAN PORT	add trunk port to the vlan N
vlan/create	VLAN	create the VLAN with tag N
vlan/deltrunkport	VLAN PORT	del trunk port to the vlan N
vlan/name	VLAN NAME	set the NAME for VLAN tag N
vlan/show	[VLAN]	print the list of defined vlan
vlan/remove	VLAN	remove the VLAN with tag N