

Maquette du projet Green IT

Ce wiki sert de support à l'installation de la maquette du projet tutoré Green IT.

Câblage du réseau

Branchement

Configuration du commutateur

- Présentation d'EnergyWize

Afin de contrôler l'alimentation POE des équipements branchés au commutateur, nous utilisons une technologie développée par Cisco, EnergyWize.

Reposant d'abord sur une implémentation SNMP (appelée « phase 1 », à partir de l'IOS 12.2(50)SE), la version actuelle fonctionne avec son propre protocole (« phase 2 »). Nous en sommes actuellement à la phase 2.5. Le protocole permet à la fois d'interroger le réseau (« Combien consomme mon bâtiment ? », « Combien consomment mes équipements non critiques ? », « Combien consommerai-je si je passais au niveau d'énergie supérieure ») et de le piloter: • soit en contrôlant l'énergie envoyée sur les ports PoE ce qui permet de gérer directement n'importe quel équipement PoE (caméra/téléphone IP/Point d'accès,...) • soit en envoyant des commandes aux équipements compatibles EnergyWize qui vont réagir à cette commande (pour une multiprise, il pourra s'agir d'éteindre ou de modifier la puissance délivrée à la prise par exemple). EnergyWize est installé de base avec l'IOS (sans coût de licence supplémentaire).

- Activation d'EnergyWize

Pour activer complètement EnergyWize, il faut configurer un « domaine » EnergyWize. Un équipement (commutateur/routeur) ne peut appartenir qu'à un seul domaine EnergyWize. Les équipements appartenant à un même domaine EnergyWize peuvent s'interroger entre eux (« Quelle est la consommation électrique du domaine ? ») ou bien agir les uns sur les autres (« Eteindre tous les appareils de faible criticité »). Il est recommandé de configurer un domaine EnergyWize pour un bâtiment, ce qui correspond à une continuité de distribution électrique, mais vous êtes libre d'organiser autrement votre gestion électrique. Vous êtes limité par le nombre de points à gérer dans un même domaine. Au delà de 10 000 équipements, il est conseillé de subdiviser le domaine.

Pour configurer complètement EnergyWize, il faut entrer 3 commandes: 1. une commande pour créer le domaine ce qui va permettre aux équipements de communiquer entre eux. Les équipements ont besoin d'un mot de passe pour vérifier l'intégrité des messages. Ce mot de passe est donc commun à tous les commutateurs du domaine. 2. une commande pour attribuer un mot de passe propre à la communication entre les agents (comme des téléphones/PDU/PC) directement connectés au commutateur et le commutateur. Il peut être différent pour chaque commutateur d'un même domaine. 3. Un troisième mot de passe est nécessaire pour la communication entre un serveur d'administration EnergyWize et le commutateur. Il peut être également différent au sein des commutateurs d'un même domaine.

```
// création du domaine
(config)# energywise domain green security shared-secret 0 bonjour
// Configure le mot de passe EnergyWise pour les communications entre les
agents et le commutateur
(config)# energywise endpoint security shared-secret 0 bonjour
```

On peut attribuer, une importance, un nom, un rôle à un commutateur EnergyWise. Il faut néanmoins retenir qu'on ne peut pas éteindre le commutateur via EnergyWise. Evidemment il faut être en mode configuration. Pour configurer le commutateur, on entrera donc (peu importe l'ordre):

```
(config)# energywise name commutateurGreen
(config)# energywise role Cisco8ports
(config)# energywise importance 100
```

- Configuration des interfaces PoE

La configuration des interfaces suit la même logique que pour le commutateur, il faut juste se placer au bon niveau de configuration de l'IOS : - si l'on désire configurer une interface : (config)# interface fa0/1 - si l'on désire configurer un ensemble d'interface : - contiguës : (config)# interface range fa0/1 – 10 par exemple - éparses : (config)# interface range fa0/1 , fa0/10 , fa0/12 On peut ensuite configurer l'interface comme décrit précédemment, en lui attribuant un rôle, un nom, des mots-clés etc... On a accès à une commande supplémentaire energywise activitycheck qui va indiquer au commutateur de vérifier s'il n'y a pas d'appel en cours avant d'éteindre un téléphone PoE.

```
(config)# interface FastEthernet0/2
(config-if)# energywise name lampe
(config)# interface FastEthernet0/3
(config-if)# energywise activitycheck
(config-if)# energywise name telDuPDG
(config-if)# energywise keywords tel
(config)# interface FastEthernet0/5
(config-if)# energywise activitycheck
(config-if)# energywise name telDuSecrétaire
(config-if)# energywise keywords tel
(config)# interface FastEthernet0/7
(config-if)# energywise wifi
(config-if)# energywise name bornewifi
(config-if)# energywise keywords wifi
```

- Requêtes EnergyWise permettant de couper l'alimentation des interfaces

Une requête s'applique uniquement sur le domaine EnergyWise auquel appartient le commutateur d'où elle est lancée. Les requêtes fonctionnent essentiellement selon 2 modes: - un mode pour récupérer la consommation des équipements - un mode pour agir sur les équipements

Nous avons travaillé sur le mode permettant d'agir sur les équipements. Dans une évolution du projet, il sera possible de mettre à profit celles permettant de récupérer la consommation afin de mesurer en temps réel la consommation. Il serait ainsi possible de mesurer les économies d'énergies réaliser par rapport à une gestion des équipements sans domotique. Les requêtes commencent par filtrer les équipements auxquels elles vont s'appliquer. Elles filtrent d'abord par importance :

energywise query importance <importanceMaxDesEquipementsAFiltrer> ...

Ici on sélectionne tous les équipements qui ont une criticité (« importance ») inférieure ou égale à importanceMaxDesEquipementsAFiltrer. ainsi « energywise query importance 100 » sélectionne tous les équipements avec une importance ≤ 100 or 100 étant l'importance maximum, cela sélectionne tous les équipements. Une fois que l'on a filtré par équipement, on peut filtrer soit par nom soit par mot clé (par commodité, on choisit importanceMaxDesEquipementsAFiltrer = 100) :

```
energywise query importance 100 name <nomDesEquipements> ...
energywise query importance 100 keywords <motCle> ...
```

La fonction set level permet d'appliquer un niveau d'énergie, c'est celle que nous utilisons pour activer ou désactiver les interfaces. Le niveau 0 permet de couper l'alimentation, le niveau 1 l'active.
● set level <niveauDEnergie> permet d'appliquer un niveau d'énergie

Commande pour allumer la lampe :

```
energywise query importance 100 name lampe set level 1 timeout 1
```

Commande pour éteindre la lampe :

```
energywise query importance 100 name lampe set level 0 timeout 1
```

- Mise en place une politique récurrente

Il est possible d'installer des politiques récurrentes sur les commutateurs, c'est-à-dire changer les niveaux d'énergie sur une interface selon la date (semaine,jour, mois,heure, etc). Les récurrences sont associées à une interface (dès lors, nous les configurerons en mode interface « (config-if)# ») et non à des filtres d'importance ou de nom comme les requêtes. Il ne faut donc pas changer l'équipement de port après configuration, ce qui peut s'avérer plutôt contraignant. En revanche, une fois la récurrence programmée, le commutateur peut perdre la communication avec la plateforme d'administration, la récurrence continuera à s'appliquer. On peut voir les récurrences disponibles via la commande show energywise recurrences :

On a ici une récurrence qui éteint (« set level 0 » à 17h33) et rallume (« set level 10 ») le port PoE à 19h06. En dessous, il y a aussi une partie « Alarms » sur laquelle on ne peut pas agir. La syntaxe de la commande est la suivante:

```
(config-if)# energywise level 0 recurrence importance 10 at 19 06
```

Exemple :

```
energywise level 10 recurrence importance 100 at 30 23 1-31 1,3,4 *
```

Cette récurrence se déclenche à 23h30 (30 23), du 1er au 31 du mois (1-31), les 1ère, 3ème et 4ème mois de l'année, pour tous les mois de l'année(*). Si le PoE est désactivé sur l'interface, la commande EnergyWise est tout de même exécutée. Si le niveau d'énergie EnergyWise est 0 au moment où l'administrateur réactive le PoE sur le port, celui-ci restera éteint.

```
GreenIT#sh run
Building configuration...
```

Current configuration : 2910 bytes

```
!  
version 15.0  
no service pad  
service timestamps debug datetime msec  
service timestamps log datetime msec  
no service password-encryption  
!  
hostname GreenIT  
!  
boot-start-marker  
boot-end-marker  
!  
!  
enable password bonjour  
!  
username user password 0 bonjour  
no aaa new-model  
system mtu routing 1500  
!  
!  
!  
energywise domain green security shared-secret 0 bonjour  
energywise name commutateurGreen  
energywise role Cisco8ports  
!  
energywise endpoint security shared-secret 0 bonjour  
!  
crypto pki trustpoint TP-self-signed-2901296768  
  enrollment selfsigned  
  subject-name cn=IOS-Self-Signed-Certificate-2901296768  
  revocation-check none  
  rsakeypair TP-self-signed-2901296768  
!  
!  
crypto pki certificate chain TP-self-signed-2901296768  
  certificate self-signed 01  
    3082022B 30820194 A0030201 02020101 300D0609 2A864886 F70D0101 05050030  
    31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274  
    69666963 6174652D 32393031 32393637 3638301E 170D3933 30333031 30303031  
    30335A17 0D323030 31303130 30303030 305A3031 312F302D 06035504 03132649  
    4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32 39303132  
    39363736 3830819F 300D0609 2A864886 F70D0101 01050003 818D0030 81890281  
    8100AA73 B187EB21 E5E65F36 996F00D5 DA6C0F46 3670D1DF 23B4A4DE 91F0EE3B  
    51BA7216 21A83ACD 1ED7571E 3D3C97B4 E4AA45B4 6F585420 978C5D7A 245925C2  
    10219670 B86136C6 5FCCCB84 B45B6BB5 42FA88FE FC43C906 F9A1A124 D6F16B07  
    5C81767F 1D566598 CEBE99DB E85B3E8B E773A9DC BC6A5889 33C3A1C0 08B861FB  
    BEA10203 010001A3 53305130 0F060355 1D130101 FF040530 030101FF 301F0603
```

551D2304	18301680	142D7906	A94A09A1	C96B329B	DCD21C94	8A5BFC49	EC301D06
03551D0E	04160414	2D7906A9	4A09A1C9	6B329BDC	D21C948A	5BFC49EC	300D0609
2A864886	F70D0101	05050003	8181004A	1DD831C9	D5DC5DE4	AAD3F708	E91C337A
158E6193	21704F9B	9AE7F088	AFE98E80	E0A0031F	D6DA4077	B893632E	C9C4F542
8F156C04	1F8CF28C	41CF27DB	BD729748	4FBF9DBF	407B66FB	BFE0E5C5	436819A8
DE3A5A7E	A07A6F55	08C5801F	1ABAC6E6	51BD63EE	EA67F3DB	18FC27F9	99CC9D4E
D2761C1E	A5619769	F5E2FE01	E1F479				

quit !

```
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
```

```
interface FastEthernet0/1
!
interface FastEthernet0/2
    energywise name lampe
!
interface FastEthernet0/3
    energywise activitycheck
    energywise keywords tel
    energywise name telDuPDG
!
interface FastEthernet0/4
!
interface FastEthernet0/5
    energywise activitycheck
    energywise keywords tel
    energywise name telDuSecretaire
!
interface FastEthernet0/6
!
interface FastEthernet0/7
```

```
energywise role wifi
energywise keywords wifi
energywise name bornewifi
!
interface FastEthernet0/8
!
interface GigabitEthernet0/1
!
interface Vlan1
 ip address 192.168.1.1 255.255.255.0
!
ip http server
ip http secure-server
!
!
!
!
!
!
line con 0
line vty 0 4
 password bonjour
 login
line vty 5 15
 login
!
end
```

Configuration de la borne wifi

Mise en place d'un serveur téléphonique Asterisk

Afin de toujours plus économisé en énergie nous avons besoin d'utiliser un client léger ou un serveur économe en énergie. Pour répondre à ce besoin nous n'avons pas hésiter, la sortie récente du RaspBerry Pi était une bonne raison pour pouvoir tester ses compétences.

Le premier but du RaspBerry était d'accueillir tout nos serveurs que ce soit serveur Web, serveur téléphonique et Serveur DHCP. Mais par manque de temps et de puissance du RaspBerry nous avons installer qu'un seul serveur téléphonique permettant à nos deux téléphone PoE de pouvoir communiquer ensemble. Le serveur Web et DHCP étant sur notre ordinateur de test sous Linux.

Nous allons maintenant détailler toutes les étapes de réalisations :

Premier démarrage du RaspBerry

Pour simplifier l'installation et la configuration première des RaspBerry son créateur a développé un petit installateur nommé Noobs (New Out Of Box Software = Nouveau logiciel qui fonctionne à la

sortie de la boîte) qui permet d'installer un système d'exploitation de notre choix.

Dans un premier temps nous avons besoin de formater la carte SD qui va accueillir notre système. Pour ce faire rien de plus simple cliquer droit sur Windows et formater.

Notre carte SD étant formatée nous pouvons commencer l'installation, dans un premier temps nous devons suivre cette étape afin d'installer Noob sur notre Raspberry :

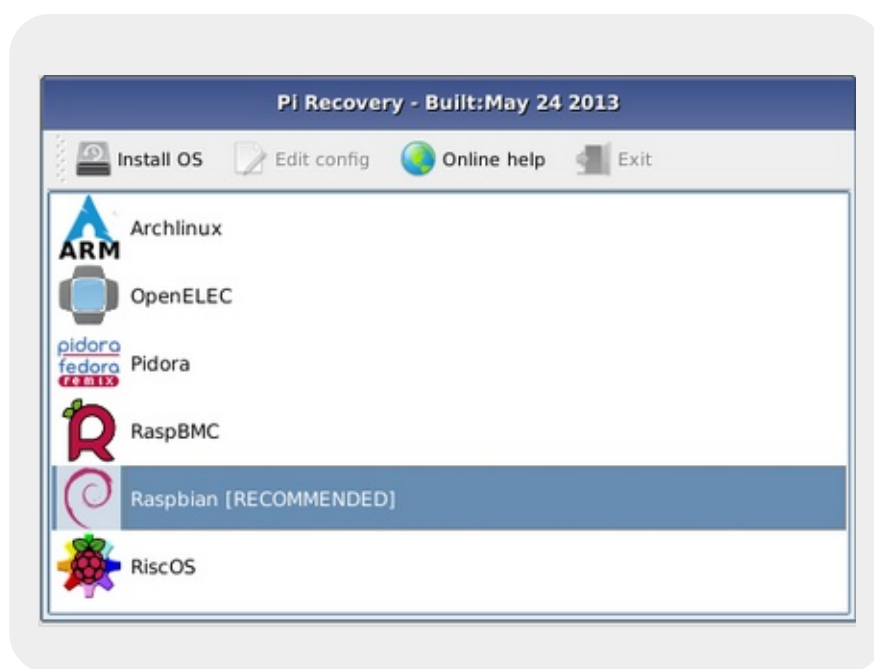
1. Télécharger Noob
2. Le décompresser
3. Copier tous les fichiers obtenus sur la carte SD
4. Insérer la carte dans le Raspberry et l'allumer



NOOB disponible ici :

http://downloads.raspberrypi.org/NOOBS/NOOBS_v1_3_2.zip

Tout ceci étant fait nous pouvons installer le système de notre Raspberry. Voici l'écran que nous avons sur le Raspberry, ici nous avons le choix de plusieurs systèmes à installer, Raspbian, Pidora, RaspBMC, RiscOS, Arch Linux ou OpenELEC.



Une fois le système installé, NOOB reste présent sur la carte et vous pouvez le réactiver simplement en maintenant la touche SHIFT appuyée pendant le démarrage.

Une fois le système choisi nous avons plus rien à faire, tout se fait tout seul. Pour notre part nous avons choisi Raspbian qui est un dérivé de Debian mais adapté au Raspberry PI.

Installation du serveur Asterisk

Maintenant que nous avons notre système opérationnel, il faut installer le serveur Asterisk. Asterisk est un autocommutateur téléphonique privé (PABX) open source et propriétaire (publié sous licence GPL et licence propriétaire¹) pour systèmes GNU/Linux.

Pour répondre à ce besoin nous avons trouvé un logiciel libre et puissant du nom de FreePBX qui est un serveur Asterisk léger.

Pour installer ce logiciel il faut suivre plusieurs étapes que nous détaillons plus bas.

Dans un premier temps nous devons télécharger les fichiers d'installation de FreePBX afin de l'installer sur notre carte SD.

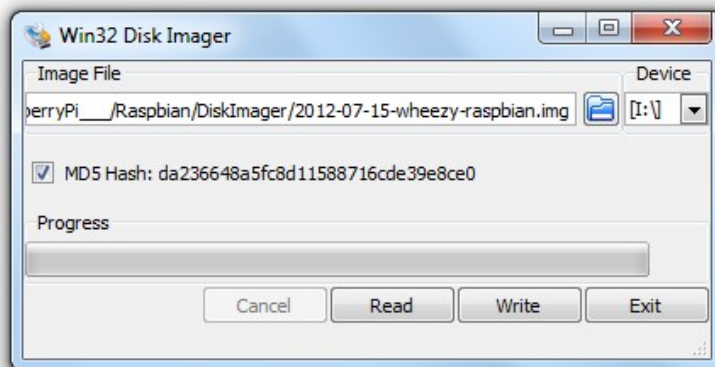
Ensuite il faut installer les fichiers téléchargés dans notre carte SD pour cela nous devons utiliser un logiciel du nom de Win32DiskImager qui va permettre d'installer des images et ainsi installer FreePBX.



Win32DiskImager disponible gratuitement ici :

<http://sourceforge.net/projects/win32diskimager/>

Pour ce faire il suffit d'ouvrir le logiciel sélectionné l'image à installer ainsi que la carte SD dans le device et de cliquer sur write pour écrire sur la carte du RaspBerry:



Ensuite nous allumons notre RaspBerry avec l'image fraîchement installée et FreePBX est enfin installée et accessible.

Configuration du serveur Asterisk

Maintenant que nous avons notre image installée nous pouvons configurer notre serveur.

Dans un premier temps il est utile de rappeler les identifiant et mot de passe utilisé pour cette image :

- Pour root sur Mysql : raspberry
- Login SSH :
- user : **root**
- pwd : **raspberry**
- Login FreePBX par défaut :
- user : **admin**
- pwd : **admin**



Il faut bien faire attention puisque le clavier est en qwerty.

Un menu est accessible en tapant raspi-config en ligne de commande.

Grâce à raspi-config nous avons accès à beaucoup de configuration système :

- Informations
- Partition de la carte SD
- Configuration du clavier
- Configuration du mot de passe
- Configuration de l'heure
- Overclockage
- Mise à jour etc ...

Création d'un serveur DHCP

[Syntaxe dokuwiki](#)

From:

<http://wiki.lahoud.fr/> - **wikiroute**

Permanent link:

<http://wiki.lahoud.fr/doku.php?id=greeniut&rev=1395859897>

Last update: **2014/03/26 19:51**

