Impression PDF Android

Philippe Wambeke - LoliGrUB :backend: revealjs :revealjs_theme: league :revealjs_control: false

L'objectif

Un utlisateur “Lambda” doit pouvoir imprimer simplement un PDF depuis son smartphone / sa tablette Android.

* Sans que Google ne sache ce qui est imprimé * Sans que le document ne fasse le tour du monde * Sans devoir donner tous les droits à Google sur l'imprimante * Sans être obligé d'acheter une imprimante réseau * En utilisant des logiciels libres

Les ingrédients

=== Au niveau matériel

* Une imprimante (réseau ou USB) * Un ordinateur connecté à l'imprimante (un RPI suffit amplement) * La mise en réseau de l'ordinateur et du périphérique Android

=== Au niveau logiciel

[%step] * Sur le serveur: [%step]

- * CUPS (Common Unix Printing Sytem)
- * Un serveur web (Apache par exemple) avec le module “webdav”
- * incron: permet d'être averti sur changements du système de fichiers
- * Un script Bash fait maison

* Sur Android: [%step]

- * Un (bête) client webdav

La recette

Quel va être le parcours du PDF ?

```
. Envoyé depuis Android vers le serveur web à l'aide du protocole webdav
. Incron détectera qu'un nouveau fichier vient d'être "uploadé" et
déclenchera l'exécution du script bash maison
. Le script maison vérifie qu'il s'agit bien d'un PDF et l'imprime via CUPS
```

[NOTE.speaker] – Expliquer brièvement l'extension http: *webdav* –

Mise en place

=== Installation du serveur web * Installer Apache et CUPS: —- sudo apt-get install apache2 cups —- * Activer le module webdav: —- sudo a2enmod dav sudo a2enmod dav_fs —- === Configuration du serveur web (1/3) Dossier de destination:: Le dossier qui recevra les fichiers PDF sera ici /var/www/webdav. Ce dossier doit être créé et appartenir à Apache. —- sudo mkdir -p /var/www/webdav sudo chown -R www-data:www-data /var/www/webdav —- === Configuration du serveur web (2/3) Configurer Apache et le module webdav:: Il faut aussi l'indiquer à Apache dans le dossier sites-available par le fichier /var/www/webdav: —- Alias /webdav /var/www/webdav <Location /webdav> Options Indexes MultiViews DAV On Order deny,allow deny from all allow from 192.168.2 </Location> —- [NOTE.speaker] – Montrer l'arborescence des dossiers /etc/apache2 ATTENTION: si apache version 2.4, le fichier webdav doit se nommer webdav.conf !!! – === Côté serveur web: configuration (3/3) Il faut ensuite activer le module webdav et recharger la configuration de Apache: —- sudo a2ensite webdav sudo service apache2 reload —- [NOTE.speaker] – Oui, même sur debian 8 et systemctl, la commande service fonctionne ! – === Configuration de CUPS (1/2) Port d'écoute:: Permettre d'accéder à CUPS depuis le réseau local en modifiant le fichier de configuration /etc/cups/cupsd.conf. Remplacer la ligne: —- Listen localhost:631 —- par: —- Listen 631 —- === Configuration de CUPS (2/2) Permettre d'accéder à l'interface d'administration de CUPS: —- sudo cupsctl –remote-admin —- Finalement, on redémarre CUPS: —- sudo systemctl restart cups —- === Ajouter l'imprimante En fonction de la marque et du modèle d'imprimante, il peut être nécessaire d'installer des pilotes supplémentaires. Le site https://www.openprinting.org/printers dresse une liste exhaustive des modèles et des drivers correspondant. Une fois le bon driver installé, reste à ajouter l'imprimante: * Ouvrir l'URL http:<ip_serveur_web>:631 * Choisir “Ajouter une imprimante” * Suivre les instructions de l'assistant

=== Incron

* Installer incron et permettre à root de l'utiliser. Taper en tant que root:

---

sudo su # apt-get install incron # echo root > /etc/incron.allow

---

* Activer la surveillance du dossier de destination:

---

sudo incrontab -e

---

* Et ajouter:

/var/www/webdav IN_MOVED_TO /home/wap/printtest.sh $@ $#

[NOTE.speaker] – Expliquer brièvement la syntaxe et les différents modes supportés par incron:

* IN_ACCESS File was accessed (read) * IN_ATTRIB Metadata changed (permissions, timestamps, extended attributes, etc.) * IN_CLOSE_WRITE File opened for writing was closed * IN_CLOSE_NOWRITE File not opened for writing was closed * IN_CREATE File/directory created in watched directory * IN_DELETE File/directory deleted from watched directory * IN_DELETE_SELF Watched file/directory was itself deleted * IN_MODIFY File was modified * IN_MOVE_SELF Watched file/directory was itself moved * IN_MOVED_FROM File moved out of watched directory * IN_MOVED_TO File moved into watched directory * IN_OPEN File was opened –

=== Le script bash Du côté serveur, il ne reste plus qu'à copier le script bash suivant dans le dossier spécifié à l'étape précédente.

[source,bash]

#!/bin/bash # Petit script pour imprimer un pdf en ligne de commande (trop fort GNU/Linux) complete_file_name="$1/$2"

file_name=$(file "${complete_file_name}" | grep "PDF document" | cut -f1 -d':') if [-n "${file_name}"]; then

```
lp -d "HP_DESKJET_990C" "${file_name}"
rm "${file_name}"
```

fi

Côté Android

Il suffit ensuite d'installer n'importe quel client webdav.

L'impression du PDF se fait simplement en "uploadant" le fichier PDF sur le serveur web.

[NOTE.speaker] – Expliquer qu'il est asssez difficile de trouver un client webdav libre sur Android. Et que le client en ligne de commande sous GNU/Linux s'apelle cadaver . –

Pour aller plus loin

Il est possible d'améliorer les choses:

* en implémentant une authentification pour l'envoi des PDF (mode digest) * imprimer des images en faisant une conversion à la volée (mogrify) * imprimer des documents odt en faisant une conversion à la volée (mais nécessite l'installation d'une usine à gaz)

Merci

Questions ?

From:
https://www.loligrub.be/wiki/ - **LoLiGrUB**

Permanent link:
https://www.loligrub.be/wiki/atelier20161119-impression-pdf-android-run?rev=1481828435

Last update: **2016/12/15 19:00**