

---

# RSSKISS : Un lecteur de flux RSS en 100 lignes de code

Copyright © 2014 pabr@pabr.org  
Tous droits réservés. (All rights reserved.)

rsskiss est un bricolage "vite fait mal fait" pour remplacer Google Reader.

**READ THE HYPERTEXT VERSION HERE:**  
<http://www.pabr.org/rsskiss/rsskiss.fr.html>

Historique des versions		
1.0	2014-04-12	Première publication.

---

## Table des matières

1. Motivations .....	3
2. Prérequis .....	3
3. Fonctionnement .....	3
4. Test rapide .....	4
5. Avertissements .....	4

---

# 1. Motivations

Lorsque Google a mis fin à son service Reader en juillet 2013 j'en ai profité pour essayer de passer moins de temps à lire l'information en continu. Évidemment cela n'a pas duré et j'ai dû partir à la recherche d'une solution de remplacement. Les principales alternatives étaient :

- **Services en ligne** . Plusieurs prestataires ont profité du vide laissé par Google Reader pour offrir des solutions équivalentes, mais je recherchais une solution pérenne. Si Google ne peut pas offrir un agrégateur de flux gratuit à long terme, à qui peut-on faire confiance ?
- **Extensions pour navigateurs** . Comme j'utilise des navigateurs très différents sur plusieurs types d'appareils, il aurait été difficile d'assurer la synchronisation.
- **Auto-hébergement d'un serveur** . Il existe de nombreux projets open-source de lecteurs RSS avec une interface web, mais ils requièrent tous Java/PHP/Perl et une base de données transactionnelle.

J'ai fini par bricoler un lecteur de flux minimaliste qui répond à 80 % de mes besoins avec seulement 100 lignes de script shell et de règles XSL.

## 2. Prérequis

- Un serveur UNIX/Linux générique
- **xsltproc** [<http://xmlsoft.org/XSLT/xsltproc.html>] (de libxslt). On pourrait se contenter d'expressions régulières, mais puisque XML est maintenant présent sur toutes les plate-formes, autant utiliser les outils dédiés.
- **wget**.
- Un moyen pour diffuser le HTML produit vers plusieurs terminaux : un serveur web (de préférence), ou un hébergement web, ou simplement sendmail.
- Un lecteur HTML.

## 3. Fonctionnement

- Les abonnements aux flux sont stockés dans un fichier texte (exemple [rskiss-1.0/test.org]). Le format est compatible avec Org-mode sous emacs.

La dernière colonne indique comment chaque flux doit être affiché :

- **rawhtml** : Copier le HTML fourni par les flux (**dangereux**).
- **noscript** : Désactiver les tags <script>.
- **notags** : Masquer tous les tags HTML.
- (par défaut) : Afficher seulement le titre.
- `rss2table.xsl` [rskiss-1.0/rss2table.xsl] et **xsltproc** transforment les flux RSS et Atom du format XML vers un format texte avec un article par ligne.
- `rsskiss_poll.sh` [rskiss-1.0/rsskiss\_poll.sh] lit le fichier des abonnements, récupère les flux et présente les articles non lus sous forme HTML sur sa sortie standard.
- (option) `rsskiss_poll.cgi` [rskiss-1.0/rsskiss\_poll.cgi] invoque **rsskiss\_poll.sh**. On peut l'exécuter soit comme une tâche **cron**, soit comme un script CGI pour déclencher manuellement la relecture des flux.

- (option) `rsskiss_mark_all_read.cgi` [`rsskiss-1.0/rsskiss_mark_all_read.cgi`] permet de marquer les articles comme lus depuis un navigateur web. Alternativement, on peut le déclencher par retour de mail, ou automatiquement par **cron**, ou en ligne de commande.

## 4. Test rapide

```
wget http://www.pabr.org/rsskiss/rsskiss-1.0/rsskiss_poll.sh [rsskiss-1.0/rsskiss_poll.sh]
wget http://www.pabr.org/rsskiss/rsskiss-1.0/rss2table.xsl [rsskiss-1.0/rss2table.xsl]
wget http://www.pabr.org/rsskiss/rsskiss-1.0/test.org [rsskiss-1.0/test.org]
sh ./rsskiss_poll.sh test.org > new.html
firefox new.html
```

## 5. Avertissements

- **rss2table.xsl** a été testé avec quelques dizaines de flux générés par des systèmes de gestion de contenus différents, mais n'est probablement pas compatible avec toutes les variantes de RSS et Atom.
- Le fait d'afficher du code HTML d'origine tierce depuis un serveur HTTP de confiance peut poser des problèmes de sécurité. **rsskiss** essaie de nettoyer les données potentiellement dangereuses, mais sans garantie d'exhaustivité.