



Éditorial

« À l'insu de mon plein gré »

Louis Nyffenegger 

Les attaques de type *Cross Site Request Forgery* (CSRF) sont connues depuis 2001 [1] et pourtant, elles commencent tout juste à être évoquées par les développeurs d'applications Web et les experts en sécurité. C'est ainsi que les CSRF sont entrés en cinquième place du Top 10 de l'OWASP [2].

Cette attaque utilise les accréditations d'un utilisateur sur un site pour se faire passer pour lui. Un intrus peut par exemple héberger une page avec du code HTML et/ou Javascript qui va faire réaliser au navigateur de la victime (à l'insu de son plein gré) des requêtes GET ou POST sur d'autres serveurs (*routeur, application intranet, webmail,...*). Il faut bien sûr que la victime soit authentifiée sur l'application vulnérable, ce qui rend beaucoup de personnes sceptiques sur les impacts réels de cette vulnérabilité.

En effet, cette attaque n'est souvent pas prise en compte, il est ainsi possible dans des applications connues de trouver des CSRF permettant : l'envoi arbitraire d'emails, l'envoi arbitraire de commentaires sur un blog, l'ajout d'une machine dans une DMZ, l'ajout d'un utilisateur *root* sur une interface de gestion de serveur, comme nous l'avons présenté avec Renaud Feil lors du SSTIC [3].

Espérons tout de même que nous n'assisterons pas à l'envoi massif de mails dans les listes de diffusion de sécurité pour des vulnérabilités de ce type (les découvertes de XSS dans d'obscures applications PHP sont déjà largement suffisantes).

Les CSRF sont particulièrement graves quand ils sont présents sur des sites commerciaux, l'exemple du CSRF *1-Click* d'Amazon [4] montre bien que sur un site grand public sur lequel la durée des sessions est longue (pour augmenter l'interactivité), un simple CSRF peut avoir un impact important.

Une autre application connue des CSRF est le *Drive-By Pharming* [5], qui consiste à utiliser un CSRF afin de modifier le serveur DNS utilisé par un routeur (type *box ou routeur wifi) afin de contrôler le trafic de la victime et d'amener celle-ci sur un site malveillant. Les mots de passe par défaut de tous les routeurs étant connues, il est très simple de créer une page malveillante amenant le navigateur à faire une requête GET (par exemple, avec les accréditations par défaut). On peut résumer cette action à la simple inclusion du code suivant dans un page

```
HTML : 
```

Comment se prémunir ? En tant qu'utilisateur, il faut éviter d'utiliser le même navigateur pour les applications sensibles et pour une utilisation journalière. En tant que développeur, il faut s'assurer de l'unicité de chaque requête, pour cela, un jeton non prévisible doit être ajouté dans chaque formulaire et recopié dans une variable de session (stockée côté serveur), certains frameworks (comme *Struts*) le permettent déjà. La vérification du *referer* est une méthode qui peut être utilisée mais n'est pas suffisante car certaines personnes font le choix de ne pas envoyer de *referer* et diverses techniques permettent à un intrus de bloquer cet envoi. Enfin, certains navigateurs tentent de bloquer les attaques CSRF grossières telles que les formulaires envoyés automatiquement par du code Javascript.

Cette ancienne vulnérabilité remise au goût du jour est encore un exemple de la prise en compte tardive d'une technique connue, même si les solutions sont simples à mettre en place (beaucoup plus que le filtrage de XSS par exemple), il y a fort à parier que les CSRF seront à l'origine des prochaines attaques visant les applications web grand public.

- [1] <http://www.tux.org/~peterw/csrf.txt>
- [2] http://www.owasp.org/index.php/Top_10_2007
- [3] http://www.hsc.fr/ressources/presentations/sstic07_CSRF/index.html
- [4] <http://shiflett.org/blog/2007/mar/my-amazon-anniversary>
- [5] http://www.symantec.com/avcenter/reference/Driveby_Pharming.pdf

À propos de l'auteur

Louis Nyffenegger est un consultant en sécurité suisse travaillant chez HSC (Hervé Schauer Consultants - <http://www.hsc.fr>). Il est spécialisé dans la conduite d'audits, d'études et de tests d'intrusion. Il a réalisé une présentation sur les *Cross Site Request Forgeries* lors du SSTIC 2007 avec Renaud Feil. Louis remercie Renaud pour le travail réalisé ensemble et toute l'équipe d'HSC. Louis peut être contacté à l'adresse suivante : Louis.Nyffenegger@hsc.fr.