Advanced CSRF / Have fun and profit

Manfred Touron — Vincent Guasconi

Epitech Security Laboratory

5 juin 2008

CSRF en trois blocks beamer

- En gros :

  Faire exécuter une requête HTTP à un tiers sans son accord

CSRF en trois blocks beamer

- En gros :

  Faire exécuter une requête HTTP à un tiers sans son accord

- Deux avantages :
  - Casser la Cross Domain Policy
  - Agir à l'insu de l'utilisateur

CSRF en trois blocks beamer

- En gros :

  Faire exécuter une requête HTTP à un tiers sans son accord

- Deux avantages :
  - Casser la Cross Domain Policy
  - Agir à l'insu de l'utilisateur

- Un inconvénient :

  Impossibilité de récupérer ou d'intéragir avec la réponse

Time based blind SQL injection

Quoiqu'est-ce ?

Injection SQL dont aucun retour, pas même booléen n'est renvoyé à l'attaquant à travers la requête.

Time based blind SQL injection

Quoiqu'est-ce ?

Injection SQL dont aucun retour, pas même booléen n'est renvoyé à l'attaquant à travers la requête.

Une des techniques pour résoudre ce problème :

- Ralentir le serveur SQL lorsque la requête a réussi (ou non).
- IF(premiere lettre password = 'a', sleep 5, aucune action)

Time based blind SQL injection

Quoiqu'est-ce ?

Injection SQL dont aucun retour, pas même booléen n'est renvoyé à l'attaquant à travers la requête.

Une des techniques pour résoudre ce problème :

- Ralentir le serveur SQL lorsque la requête a réussi (ou non).
- IF(premiere lettre password = 'a', sleep 5, aucune action)
- Technique applicable sur toutes les injections

- Le temps d'exécution d'une requête HTTP est facilement calculable sur une CSRF.

- Le temps d'exécution d'une requête HTTP est facilement calculable sur une CSRF.
- Couplons les deux vecteurs...

- Le temps d'exécution d'une requête HTTP est facilement calculable sur une CSRF.
- Couplons les deux vecteurs...
  - Décentralisation complète de l'attaque (merci les autres)

- Le temps d'exécution d'une requête HTTP est facilement calculable sur une CSRF.
- Couplons les deux vecteurs...
  - Décentralisation complète de l'attaque (merci les autres)
  - Bypass de la Cross Domain Policy (interfaces admin, etc...)

- Le temps d'exécution d'une requête HTTP est facilement calculable sur une CSRF.
- Couplons les deux vecteurs...
  - Décentralisation complète de l'attaque (merci les autres)
  - Bypass de la Cross Domain Policy (interfaces admin, etc...)
  - Possible depuis un mail, simple et anonyme

Exemples concrets

Exemples concrets

- Dump complet d'une database depuis plusieurs milliers de machines qui ne sont que de simples visiteurs d'un site web, ou lecteurs d'une mailing-list.

Exemples concrets

- Dump complet d'une database depuis plusieurs milliers de machines qui ne sont que de simples visiteurs d'un site web, ou lecteurs d'une mailing-list.
- Michel lit ses mails au boulot, et se retrouve sans le savoir à attaquer le serveur de la boite concurrente.

Exemples concrets

- Dump complet d'une database depuis plusieurs milliers de machines qui ne sont que de simples visiteurs d'un site web, ou lecteurs d'une mailing-list.
- Michel lit ses mails au boulot, et se retrouve sans le savoir à attaquer le serveur de la boite concurrente.
- Rendre exploitable une injection SQL sur un panel d'administration, ou zone restreinte **depuis n'importe quel domaine (les injections SQL deviennent Cross Domain)**

Les seuls logs qui permettraient de remonter à l'attaquant sont dans la mémoire des clients (lié aux redirections 302 du serveur, disparaissent rapidement).

Les seuls logs qui permettraient de remonter à l'attaquant sont dans la mémoire des clients (lié aux redirections 302 du serveur, disparaissent rapidement).

Les logs sur les serveurs HTTP attaqués portent uniquement la marque de la victime.

Un couteau suisse

Un couteau suisse

- Serveur multiplex en C

Un couteau suisse

- Serveur multiplex en C
- Facilite, et organise l'exploitation

Un couteau suisse

- Serveur multiplex en C
- Facilite, et organise l'exploitation
- Unifie les requêtes

Un couteau suisse

- Serveur multiplex en C
- Facilite, et organise l'exploitation
- Unifie les requêtes
- Permet de cibler un utilisateur

DÉMO

Solution simple

Sensibilisation des développeurs web :

- Aux injections SQL

Solution simple

Sensibilisation des développeurs web :

- Aux injections SQL
- Aux protections contre les CSRF côté serveur (tokens, referer)

Solution simple

Sensibilisation des développeurs web :

- Aux injections SQL
- Aux protections contre les CSRF côté serveur (tokens, referer)

N'interpréter sous aucune raison les mails en HTML

Difficilement envisageable

- Se faire à l'idée de naviguer sous netcat

Difficilement envisageable

- Se faire à l'idée de naviguer sous netcat
- Temps de réponse aléatoire côté serveur

Difficilement envisageable

- Se faire à l'idée de naviguer sous netcat
- Temps de réponse aléatoire côté serveur
- Lire ses mails avec gnus

Remerciements

Merci pour votre attention.

Clap clap

... des questions ?

... des questions ?

Anticipons :

- Le papier complet, les slides et les sources du serveur sont disponibles à l'adresse suivante : http://esl.epitech.net/acsrf
- NoScript ne sert strictement à rien.
- Le referer ne change pas sur un redirection 302.
- Le serveur est en C dans un soucis de performances.
- Le scripting du serveur est possible très simplement.
- Aucun enfant n'a été tué dans le cadre de ces recherches.
- ESL != LSE