Conseil
Expertise
Audit & Test
Formation

# OSSIR Juillet 2011

# WebScarab
# Développement de nouveaux modules pour les tests d'intrusion

**Jérémy Lebourdais – EdelWeb**

**(jeremy.lebourdais à edelweb.fr)**

# WebScarab – Développements et améliorations



- **Introduction**
- Présentation de WebScarab
- Développements réalisés
- Conclusion
- Questions

# WebScarab – Développements et améliorations
## Introduction – Qui suis-je ?

- Consultant Sécurité chez EdelWeb depuis 2006
- Réalisation de tests d'intrusion, d'audits (techniques), expertise technique et conseil
- Cibles des tests d'intrusion: majorité de sites web
- Passionné de sécurité, de GNU/Linux et des logiciels libres, mais aussi de programmation, et d'amélioration/optimisation des outils utilisés quotidiennement
- Utilisateur de WebScarab depuis plus de 4 ans
- Pas de « lien » avec l'OWASP
- Oui, j'ai déjà utilisé BurpSuite ;-)

# WebScarab – Développements et améliorations
## Introduction – Objet de la présentation

- **Cette présentation ne concerne pas**
  - La comparaison de différents proxy intrusifs (no troll inside)
  - La « promotion » de WebScarab
  - La présentation « technique » des différentes fonctionnalités de WebScarab

- **Mais présente un retour d'expérience concret sur le développement et l'amélioration d'outils, en l'occurrence WebScarab**

# WebScarab – Développements et améliorations



- Introduction
- **Présentation de WebScarab**
- Développements réalisés
- Conclusion
- Questions

# WebScarab – Développements et améliorations
## Présentation de WebScarab

- Proxy web intrusif, projet de l'OWASP, codé en JAVA, sous licence GPL
- Premières versions du code en 2002
- N'a pas évolué pendant « longtemps » de façon « officielle » (sur le site de l'OWASP)
- Cette « vieille » version avait de nombreux bugs ➔ mauvaise réputation de cet outil auprès des pentesters
- L'auteur (Rogan Dawes) l'a fait évoluer sur son GIT (version non « officielle »)
- Moi aussi, de mon côté ;-)
- Une version NextGen (NG) a été publiée, mais n'est pas (ou très peu) maintenue

# WebScarab – Développements et améliorations
# Présentation de WebScarab

- **Pourquoi j'ai choisi WebScarab?**
- **Choix fait il y a quelques années, en partie pour les raisons suivantes**
  - Format des traces sous forme de fichiers (*-request, *-response)
  - Logiciel libre
  - L'ergonomie de BurpSuite ne me convenait pas complètement, et celui-ci n'avait pas de fonctionnalités indispensables par rapport à WebScarab
- **Quels avantages j'en ai tiré**
  - Evolution des fonctionnalités et correction des bugs, au fur et à mesure
  - Scripts d'analyse « offline » des requêtes/réponses
  - Sujet de cette présentation

# WebScarab – Développements et améliorations Présentation de WebScarab – Comment est-il codé?

- **Conçu comme un framework**
  - Un cœur qui s'occupe de gérer les connexions, le stockage des requêtes/réponses, et plus généralement tous les aspects « bas niveau » et interface graphique
  - Des plugins qui interagissent avec le cœur afin de réaliser leurs tâches : la plupart des onglets de WebScarab sont des plugins

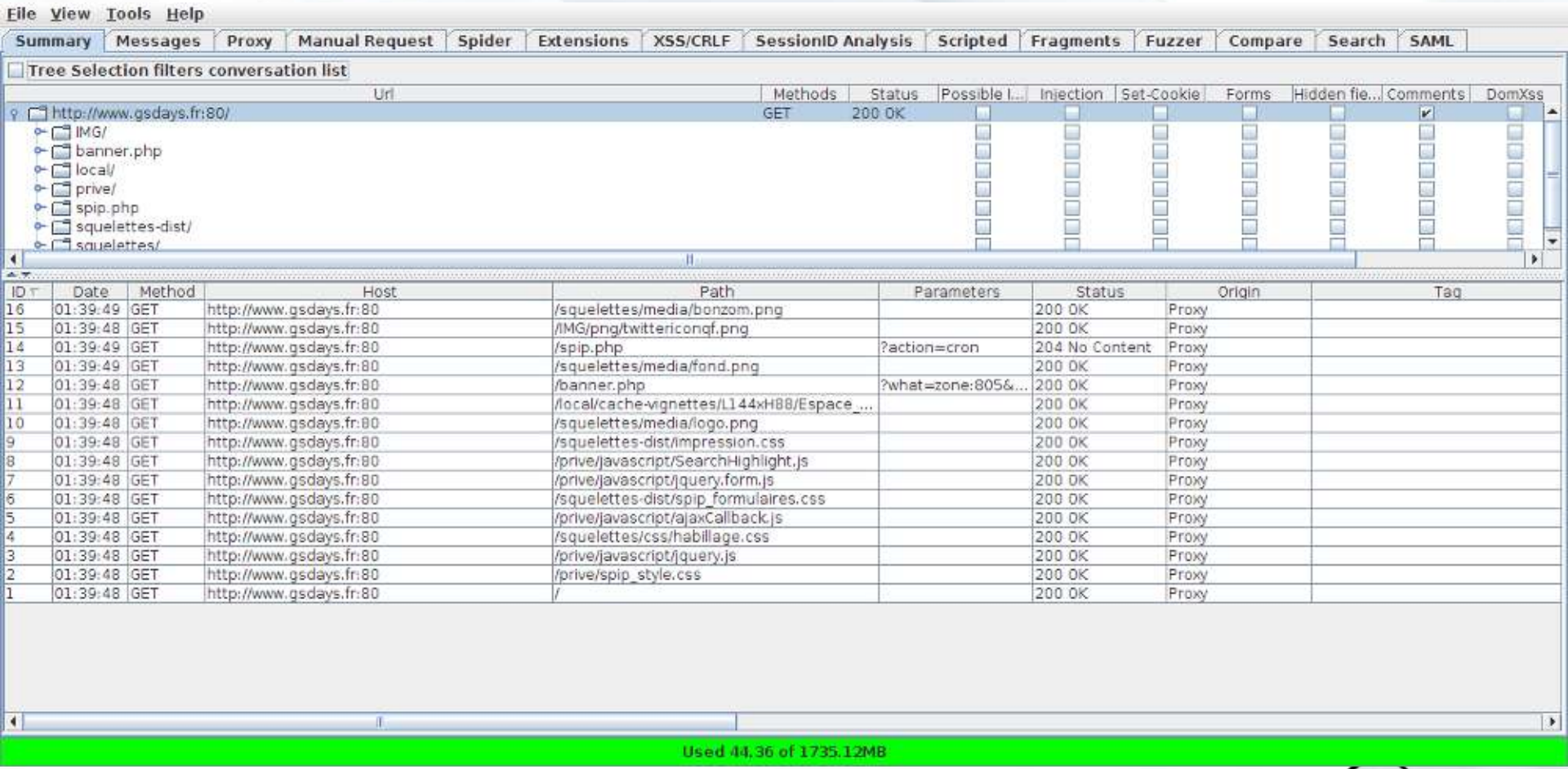
- **Segmentation du code des différentes parties (packages JAVA)**
  - Gestion des requêtes HTTP
  - Cœur du framework
  - Parsers
  - Répertoires spécifiques pour chacun des plugins
  - Interface graphique
  - Classes « utilitaires »

# WebScarab – Développements et améliorations Présentation de WebScarab – Comment est-il codé?

- **Qualité du code?**
  - Variable, d'où la présence de bugs!
  - Problèmes de gestion des cas d'erreurs (exceptions non traitées)
  - Mauvaise conception de certaines fonctions : parcours de tableaux, gestion de certains cas de figures uniquement, …
  - Utilisation d'algorithmes peu optimisés, tel que celui pour la comparaison des requêtes qui prend beaucoup trop de temps
  - Cependant, la segmentation du code est claire et les classes/méthodes/variables ont des noms explicites qui facilitent la compréhension du code

# WebScarab – Développements et améliorations
# Présentation de WebScarab

- Une capture d'écran, pour voir à quoi ressemble WebScarab



# WebScarab – Développements et améliorations



- Introduction
- Présentation de WebScarab
- **Développements réalisés**
- Conclusion
- Questions

# WebScarab – Développements et améliorations
# Développements réalisés – Pourquoi ?

- **Pourquoi avoir développé du code pour WebScarab**
  - Corrections des bugs, histoire de se faciliter la vie de tous les jours…
  - Ajout de fonctionnalités manquantes, au fil de l'eau
  - Adaptation de WebScarab aux applications rencontrées
  - WebScarab est sous licence GPL

# WebScarab – Développements et améliorations
# Développements réalisés – Liste rapide

- **Principales modifications apportées au code sur plus de 2 ans**
  - Création d'un nouveau plugin de recherche d'injections
  - Amélioration du support de l'AMF
  - Amélioration du support des objets Java Serialisés
  - Corrections des bugs
  - Ajouts de fonctionnalités « mineures »

# WebScarab – Développements et améliorations
# Développements réalisés – Plugin d'injections

- **Plugin d'injection – Besoin**
  - Certaines applications Web, généralement complexes, font usage de (très) nombreux paramètres, sur de nombreuses pages
  - Ce même type d'application complexe est généralement très peu testable avec des outils « tout-automatisés »
  - Nécessité d'utiliser l'application par le pentester (opération manuelle) tout en testant les paramètres envoyés à l'application (opération automatisée)

# WebScarab – Développements et améliorations
# Développements réalisés – Plugin d'injections

- **Solution réalisée**
  - Basé sur le plugin XSS de WebScarab
  - Suppression de la partie détection de CSRF
  - Analyse, en temps réel, des requêtes envoyées par le navigateur
  - Création de nouvelles requêtes avec les paramètres modifiés
  - Utilisation d'une chaine de caractères, modifiable, ainsi qu'un identifiant unique pour tester les injections

| Path | Parameters | Status | Origin | XSS-ng | XSS-ng Parameter | XSS-ng Tested Parameter | XSS-ng Injected Parameter Found |
|---|---|---|---|---|---|---|---|
| /WebGoat/attack | ?menu=EDW-4';!--"\<\>%3d%26{()}_EDW | 200 OK | XSS/CRLF ng | ☑ | menu | menu | body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW |
| /WebGoat/attack | ?menu=410 | 200 OK | XSS/CRLF ng | ☐ | | action | |
| /WebGoat/attack | ?menu=EDW-4%27%3B%21--%22%5C%3... | 200 OK | XSS/CRLF ng | ☑ | menu | menu | body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW<br>body:false:menu:EDW-4';!--"\<\>=&{()}_EDW |

# WebScarab – Développements et améliorations
# Développements réalisés – Plugin d'injections

- **Algorithme (simplifié)**
  - Analyse de chaque requête envoyée par le navigateur
  - Recherche de paramètres dans l'URL ou dans le corps
  - Duplication de la requête en deux copies
  - Envoi des deux copies au serveur et analyse des réponses

# WebScarab – Développements et améliorations
# Développements réalisés – Plugin d'injections

- **Points « importants » lors du développement**
  - Création d'une table de « requêtes/paramètres » testés
  - Vérification des requêtes envoyées
  - Utilisation d'un « pattern » distinctif (et modifiable) avec une marque de début et de fin, tel que: `EDW-1';!--"\<\>=&{()}_EDW`
  - Prise en compte des différents contenus de requêtes qui peuvent être envoyés (texte, POST-multipart, requête GWT, …)
  - Filtre pour délimiter le périmètre d'action du plugin
  - Plusieurs modes d'utilisation du plugin

# WebScarab – Développements et améliorations
# Développements réalisés – Plugin d'injections

- **Bilan**
  - Test des paramètres de l'application quelque soit sa complexité
  - Maîtrise complète de l'algorithme de test utilisé et donc des paramètres envoyés et testés
  - Possibilité d'évolution du moteur afin de prendre en compte un nouveau type de contenu (ex: requêtes GWT)
  - A permis de trouver de nombreuses vulnérabilités de type XSS ou SQLi, publiées (ex: Typo3) ou non
  - Permet d'avoir une certaine « exhaustivité »…

# WebScarab – Développements et améliorations
# Développements réalisés – Amélioration du support de l'AMF

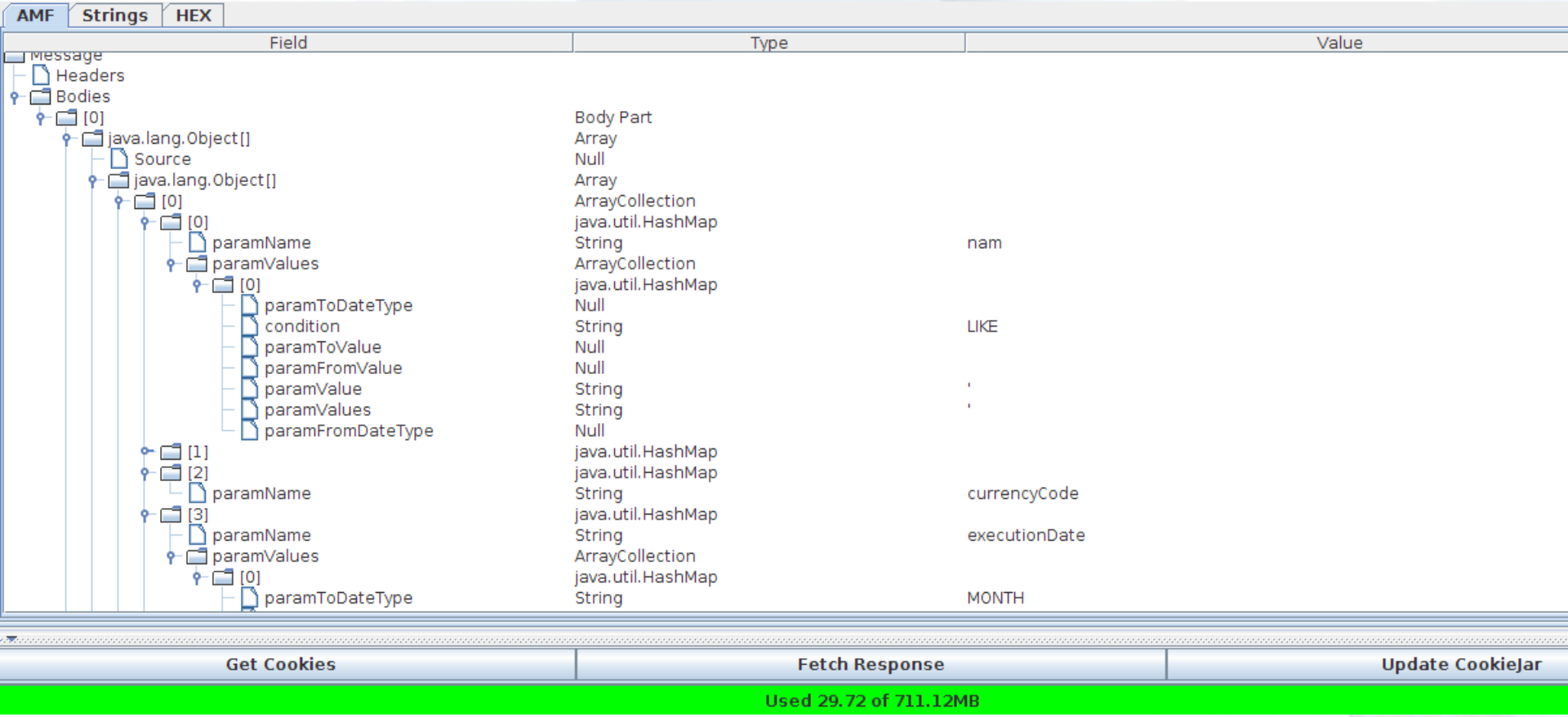
- **Amélioration du support de l'AMF – Besoin**
  - « A l'époque » (début 2009), peu de programmes supportent vraiment l'AMF et permettent de le modifier à la volée
  - Un des seuls outils utilisables est CharlesProxy
  - Support de l'AMF dans WebScarab très peu fonctionnel

# WebScarab – Développements et améliorations
# Développements réalisés – Amélioration du support de l'AMF

- **Solution réalisée**
  - Lecture des spécifications d'AMF par Adobe: facile, il y a à peine 20 pages…
  - Support de l'AMF basé sur l'utilisation des classes de BlazeDS
  - Refonte de la classe utilisée pour afficher/modifier de l'AMF dans WebScarab

# WebScarab – Développements et améliorations
# Développements réalisés – Amélioration du support de l'AMF

- Solution réalisée – Capture d'écran



# WebScarab – Développements et améliorations
# Développements réalisés – Amélioration du support de l'AMF

- **Points « importants » lors du développement**
  - L'AMF est assez « simple »… en théorie
  - L'accès et la possibilité de modification du code source ont été essentiels
  - Temps passé non négligeable
- **Bilan**
  - A permis de trouver de nombreuses vulnérabilités
  - Retour de nos clients: AMF pas ou peu testé
  - Peu de logiciels supportent complètement l'AMF

# WebScarab – Développements et améliorations
# Développements réalisés – Autres développements

- **Support des objets JAVA sérialisés**
  - Support très partiel de WebScarab « de base »
  - Développements pour modifier les objets envoyés ou reçus
  - Développements faits en 2009… depuis quelques publications sur la même fonctionnalité dans BurpSuite (cf. BlackHat EU 2010)
- **Corrections des bugs au fil des découvertes**
- **Ajout de nouvelles fonctionnalités selon nos besoins**

# WebScarab – Développements et améliorations



- Introduction
- Présentation de WebScarab
- Développements réalisés
- **Conclusion**
- Questions

# WebScarab – Développements et améliorations Conclusion

- La licence GPL est une « killer-feature » de WebScarab
- Nombreux bugs dans WebScarab ➔ rejet des utilisateurs
- Langage JAVA ➔ intégration « aisée » de certaines fonctionnalités
- Développer prend du temps…
- Mais les résultats sont là

# WebScarab – Développements et améliorations Conclusion

- **Beaucoup d'idées d'améliorations**
  - Plugin d'injection
  - Support d'AMF
- **Mais aussi de nouvelles fonctionnalités**
  - Intégration aux outils internes
  - Proxy intrusif dans le navigateur

# Questions