



WWW.MACYBER.FR
CYBERSÉCURITÉ - AUDIT - PENTEST

Confidentiel et Réservé WAYNES ENTERPRISES

Audit de Sécurité Externe

Compte rendu d'audit - 19/01/22

WAYNES ENTERPRISES

— À l'attention de

M. Bruce WAYNE	PDG Wayne Enterprises	+33 6 27 30 03 93	bruce.wayne@wayne.com
----------------	--------------------------	-------------------	-----------------------

— Votre interlocuteur MA Cyber

Antoine MARTIN	Pentester - Consultant Cybersécurité Sénior	+33 7 66 63 04 51	antoire.martin@macyber.fr
----------------	--	-------------------	---------------------------

— Circuit de validation interne MA Cyber

	Nom-Prénom	Fonction	Date	Visa
Rédigé par	Antoine Martin	Pentester - Consultant Cybersécurité Sénior	01/01/22	
Approuvé par	Antoine Martin	Pentester - Consultant Cybersécurité Sénior	01/01/22	

Version	Date	Objet de la version
Vo.1	01/01/22	Création du document
Vo.2	01/01/22	Rédaction du document
V1.0	01/01/22	Validation du document

1 - Table des matières

1 -	Table des matières	3
2 -	Introduction	4
2.1.	Contexte	4
2.2.	Modalité de l'évaluation	4
2.3.	Périmètre de l'audit	4
3 -	Synthèse de l'audit	5
3.1.	Opinion des auditeurs	5
3.2.	Criticité des vulnérabilités	6
3.3.	Synthèse des vulnérabilités identifiées	7
3.4.	Scénarios de risque identifiés	8
3.5.	Exploitabilité en fonction des scénarios de risque	9
3.6.	Synthèse des recommandations	11
4 -	Détail des vulnérabilités	12
V1 :	L'un des champs de la page de logs est vulnérable aux injections SQL - ****	12
V2 :	Certains des mots de passe utilisateurs sont stockés en clair dans la base - ****	14
V3 :	L'application ne possède pas de politique pour la gestion des mots de passe - ***	15
V4 :	La base de données est mutualisée entre toutes les sociétés utilisant BATMANAPP - ***	17
V5 :	Des failles XSS sont présentes dans plusieurs pages du site - ***	18
V6 :	Aucune analyse antivirus n'est effectuée lors de l'importation de fichiers par l'administrateur - **	21
V7 :	En cas d'inactivité, la session reste active pendant plus d'une heure - **	23
V8 :	Plusieurs utilisateurs peuvent se connecter simultanément avec un même compte - **	24
V9 :	Lors de la génération des graphiques en image/pdf, "highcharts" récupère les données via canal non chiffré - **	25
V10 :	Notre utilisateur de moindre privilège a pu se connecter à la messagerie "admin/messages" - *	26
V11 :	Un utilisateur peut, lors d'une commande, manipuler des quantités négatives pour obtenir des montants négatifs - *	27
V12 :	Une page intitulée "/tests" est présente dans l'application - *	28

2 - Introduction

2.1. Contexte

La société WAYNE ENTERPRISES a fait appel à Axicom Security pour mener une évaluation de sécurité sous forme de test d'intrusion de l'application en mode SaaS BatmanApp, développée par WayneDevCorp.

Ce document de compte rendu recense les vulnérabilités découvertes et les recommandations permettant de réduire les risques associés.

2.2. Modalité de l'évaluation

Le test d'intrusion a été effectué sur la plateforme web de test (utilisée pour réaliser les démonstrations du service de gestion de flotte mobile disponible à l'adresse suivante : <https://bat.man.app/>).

Cet audit a pour objectif d'évaluer le niveau de sécurité de la plateforme web, et de valider le bon cloisonnement des différents comptes utilisateurs en fonction de leurs droits d'accès sur les données.

Note :

- Lors de cet audit, la messagerie interne de l'application était désactivée
- La fonction d'import de document n'a pas pu être correctement testée en raison d'erreur ou de durées d'importation infinie

2.3. Périmètre de l'audit

De manière à pouvoir tester les fonctionnalités de l'application web, WayneDevCorp a mis à disposition de Axicom Security deux comptes applicatifs ayant différents profils :

- Un compte administrateur fonctionnel
- Un compte utilisateur final



3 - Synthèse de l'audit

3.1. Opinion des auditeurs

Bien qu'aucune vulnérabilité ne soit exploitable sans être authentifié sur l'application, le niveau de sécurité de l'application est jugé insuffisant au regard du niveau attendu d'une application exposée sur internet.

Les vulnérabilités critiques permettant d'accéder à l'ensemble des données stockées en base ne sont exploitables que depuis un compte ayant un profil « administrateur applicatif ».

Le filtrage des champs manipulable par les utilisateurs n'est pas systématiquement effectué, entraînant des vulnérabilités de type Injection SQL et Injection XSS.

- Une vulnérabilité de type injection SQL présente dans la partie administrateur (lecture des logs) nous a permis d'accéder aux données de la base de données et des utilisateurs (login, mot de passe, commandes...)
- Des vulnérabilités de type XSS sont présentes sur les champs des pages utilisateurs et administrateurs, ces vulnérabilités peuvent permettre de cibler les utilisateurs en exécutant du code Javascript dans leur navigateur en vue d'usurper leur identité ou d'obtenir des informations sensibles (login, mots de passe).

Certains mots de passe dans la base, dont ceux de nos comptes de test, n'ont pas été hashés avant d'être enregistrés, et sont stockés en clair.

Les données sensibles des différentes sociétés clientes de BATMANAPP sont mutualisées dans la base, la compromission d'un des clients entraîne la compromission des comptes de l'ensemble des utilisateurs (sur la plateforme de test).

La gestion des sessions est perfectible, bien qu'un mécanisme anti-bruteforce verrouille automatiquement les comptes au bout de 3 essais, l'application ne limite pas le nombre d'utilisateurs pouvant se connecter simultanément à un même compte. Par ailleurs, en cas d'inactivité, les sessions restent valides pendant plus d'1h50.

Enfin, la fonction permettant de générer des graphiques en pdf ou images est externalisée chez une société tierce « highcharts ». Des données potentiellement sensibles bien que non nominatives sont envoyées à ce service pour permettre la génération du document intégré dans l'application BATMANAPP. De plus, ces données sont transmises sur internet sans chiffrement des communications (HTTP).

3.2. Criticité des vulnérabilités

Chaque vulnérabilité présentée dans le rapport a été pondérée par une valeur allant de 1 (mineure) à 4 (critique) en fonction de l'impact qu'elle peut avoir sur l'infrastructure ou l'activité.

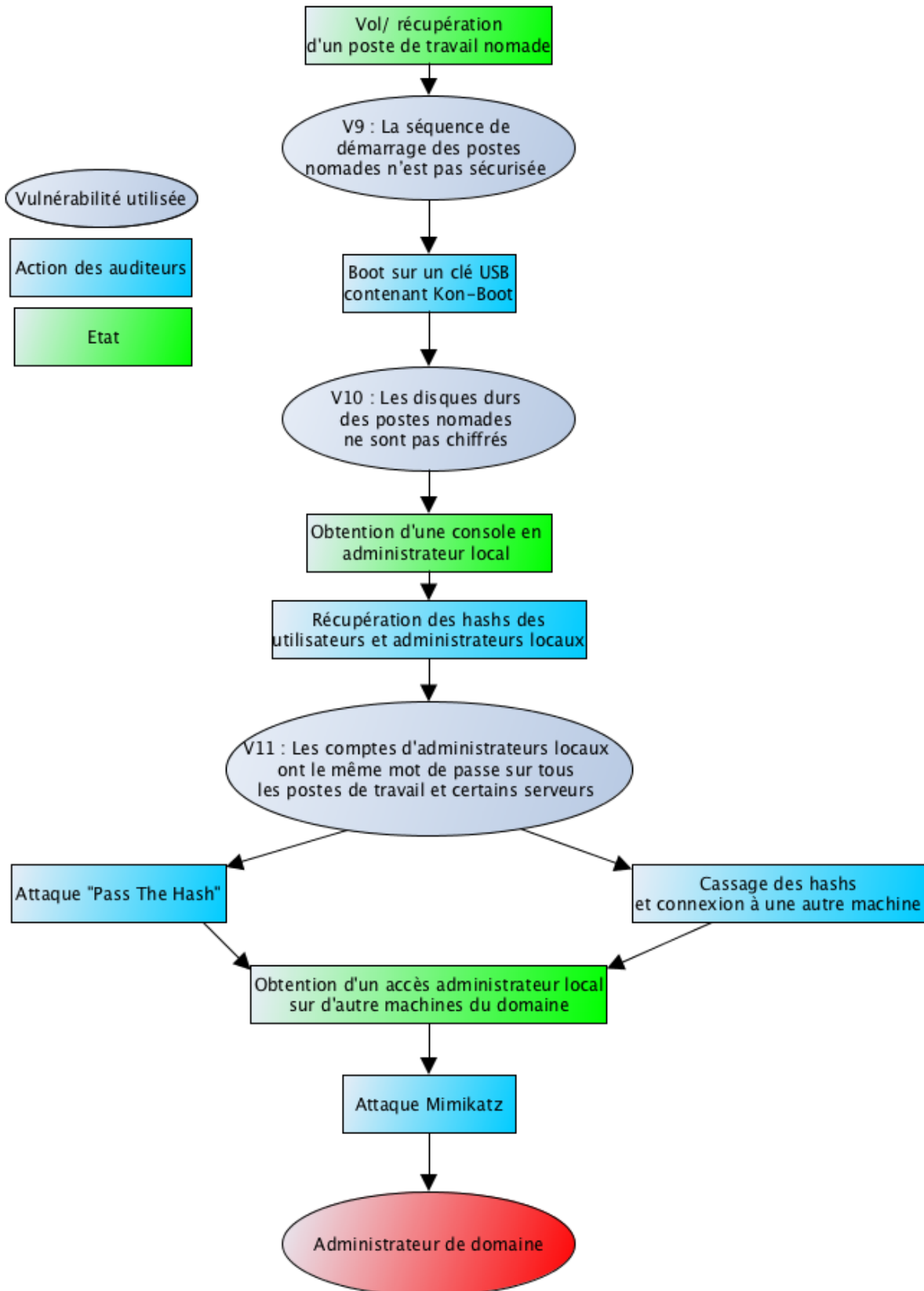
Valeur	Niveau de risque	Description	Intitulé
4	****	La vulnérabilité identifiée est « critique » : il peut en résulter une prise de contrôle totale du service ou de l'équipement. Note : Une vulnérabilité critique ne permet pas systématiquement de prendre le contrôle du système sous-jacent, mais peut impacter le service audité lui-même.	Critique
3	***	La vulnérabilité crée un risque limité. Seule une partie du service ou de l'équipement est sous contrôle après exploitation.	Majeure
2	**	La vulnérabilité crée un risque limité, par exemple la divulgation d'informations sous-jacente pourrait être utilisée pour exploiter des vulnérabilités dont le niveau de risque est plus élevé.	Moyenne
1	*	Deux cas de figure se présentent : <ul style="list-style-type: none"> • La vulnérabilité est potentielle, mais ne peut mettre en cause l'intégrité ou la confidentialité du système en l'état ; • La vulnérabilité permet l'accès à des informations techniques qui, utilisées seules, n'impactent pas la sécurité. 	Mineure

Nous recommandons l'application rapide des recommandations liées aux vulnérabilités de criticité * et **** afin d'éviter des divulgations d'informations qui pourraient avoir un impact pour WAYNES ENTREPRISES.**

3.3. Synthèse des vulnérabilités identifiées

Vuln	Description	Périmètre	Criticité
V1	L'un des champs de la page de logs est vulnérable aux injections SQL	Application web	4
V2	Certains des mots de passe utilisateurs sont stockés en clair dans la base	Application web	4
V3	L'application ne possède pas de politique pour la gestion des mots de passe	Application web	3
V4	La base de données est mutualisée entre toutes les sociétés utilisant BATMANAPP	Base de données	3
V5	Des failles XSS sont présentent dans plusieurs pages du site	Application web	3
V6	Aucune analyse antivirus n'est effectuée lors de l'importation de fichiers par l'administrateur	Application web	2
V7	En cas d'inactivité, la session reste active pendant plus d'une heure	Application web	2
V8	Plusieurs utilisateurs peuvent se connecter simultanément avec un même compte	Application web	2
V9	Lors de la génération des graphiques en image/pdf, "highcharts" récupère les données via canal non chiffré	Application web	2
V10	Notre utilisateur de moindre privilège à pu se connecter à la messagerie "admin/messages"	Application web	1
V11	Un utilisateur peut, lors d'une commande, manipuler des quantités négatives pour obtenir des montants négatifs	Application web	1
V12	Une page intitulée "/tests" est présente dans l'application	Application web	1

3.4. Scénarios de risque identifiés



3.5. Exploitable en fonction des scénarios de risque

Réf :	Description	Scénario 1 : Utilisateur non -authentifié	Scénario 2 : Utilisateur authentifié avec un compte standard	Scénario 3 : Utilisateur authentifié avec un compte administrateur applicatif
V1	L'un des champs de la page des logs est vulnérable aux injections SQL	Non exploitable, aucun impact	Non exploitable, aucun impact	Accès aux données stockées en base : Impact : Disponibilité, Confidentialité, Intégrité
V2	Certains des mots de passe utilisateurs sont stockés en clair dans la base	Non exploitable, aucun impact	Non exploitable, aucun impact	En cas d'exploitation de la vulnérabilité V1 : Accès aux mots de passe en clair, et possibilité de mener une attaque ciblée sur les utilisateurs via d'autres applications (métier, boîtes mails, VPN etc.)
V3	L'application ne possède pas de politique pour la gestion des mots de passe	Non exploitable, aucun impact	Un utilisateur authentifié peut modifier son mot de passe et mettre un nouveau mot de passe composé d'un seul caractère.	Un utilisateur authentifié peut modifier son mot de passe et mettre un nouveau mot de passe composé d'un seul caractère.
V4	La base de données est mutualisée entre toutes les sociétés utilisant BATMANAPP	Non exploitable, aucun impact	Non exploitable, aucun impact	En cas d'exploitation de la vulnérabilité V1 : Un autre compte d'administration n'ayant pas autorité sur le périmètre WAYNE ENTERPRISES peut accéder aux données WAYNE ENTERPRISES en base.
V5	Des failles XSS sont présentes dans plusieurs pages du site	Non exploitable, aucun impact	Utilisation de l'application pour cibler d'autres utilisateurs de l'application en vue d'usurper leur identité, ou d'obtenir des informations sensibles	Utilisation de l'application pour cibler d'autres utilisateurs de l'application en vue d'usurper leur identité, ou d'obtenir des informations sensibles
V6	Aucune analyse antivirus n'est effectuée lors de l'importation de fichiers par l'administrateur	Non exploitable, aucun impact	Non exploitable, aucun impact	Utilisation de l'application BATMANAPP pour diffuser des pièces jointes malveillantes
V7	En cas d'inactivité, la session reste active pendant plus d'une heure	Non exploitable, aucun impact	Accès à l'application depuis un poste de travail non verrouillé. Réutilisation d'une session volée (via l'exploitation de la vulnérabilité V5)	Accès à l'application depuis un poste de travail non verrouillé. Réutilisation d'une session volée (via l'exploitation de la vulnérabilité V5)

V8	Plusieurs utilisateurs peuvent se connecter simultanément avec un même compte	Non exploitable, aucun impact	Possibilité de conserver une session volée indéfiniment, que l'utilisateur légitime se re-connecte ou non	Possibilité de conserver une session volée indéfiniment, que l'utilisateur légitime se re-connecte ou non
V9	Lors de la génération des graphiques en image/pdf, "highcharts" récupère les données via canal non chiffré	Non exploitable, aucun impact	Transmission des données "metier" vers un tiers, sans chiffrement	Transmission des données "metier" vers un tiers, sans chiffrement
V10	Notre utilisateur de moindre privilège à pu se connecter à la messagerie "admin/messages"	Non exploitable, aucun impact	Information : valider qu'un utilisateur n'a pas la possibilité d'accéder aux messages de l'administrateur.	Non exploitable, aucun impact
V11	Un utilisateur peut, lors d'une commande, manipuler des quantités négatives pour obtenir des montants négatifs	Non exploitable, aucun impact	Information : valider que les commandes avec des montants négatifs ne sont pas honorées	Information : valider que les commandes avec des montants négatifs ne sont pas honorées
V12	Une page intitulée "/tests" est présente dans l'application	Non exploitable, aucun impact	Non exploitable, aucun impact	Non exploitable, aucun impact

3.6. Synthèse des recommandations

#	Recommandation	Priorité
R1	Il est recommandé d'utiliser des requêtes préparées (prepared statement) pour limiter les risques d'injections SQL. Par ailleurs, les champs utilisateurs doivent être encodés avant d'être utilisés par l'application.	P1
R2	Les mots de passe doivent être hashés et de préférence salés avant d'être insérés dans la base. La fonction "Password_bcrypt()" intégrée à PHP, combinée avec une politique de mot de passe renforcée offre un niveau de sécurité convenable	P1
R3	Les mots de passe doivent être validés côté serveur pour satisfaire une taille d'au moins 8 caractères et utiliser 3 parmi 4 types de caractère (min, maj, num, spé). Une liste noire de mots à ne pas utiliser doit être mise en place pour ne pas faire apparaître le nom des sociétés, de l'application, de l'utilisateur ou tout autre mot fréquemment utilisé (password)	P2
R4	Les données propres à une société ne doivent pas pouvoir être accédées par une autre. Il est préférable de créer des bases propres aux sociétés (hébergés au sein d'une même instance MySQL), l'application y accédant au travers d'un compte applicatif propre à la société et possédant des droits de lecture et écriture sur sa base et de manière extrême en lecture seule sur les données de la base BATMANAPP.	P2
R5	Il est recommandé de systématiquement filtrer et/ou encoder les entrées utilisateurs avant que celles-ci ne soient retournées au navigateur.	P2
R6	Une analyse antivirus des fichiers uploadés doit être mise en place pour protéger les utilisateurs contre d'éventuelles attaques virales. Ce mécanisme peut être effectué en amont (brique de filtrage sécurité type WAF) ou directement à l'aide d'un antivirus type ClamAV sous Linux. La détection de pièce jointe malveillante doit faire l'objet d'une alerte de sécurité.	P3
R7	Il est recommandé de diminuer la durée des sessions utilisateurs à 30 minutes. PHP met à disposition les fonctions "ini_set('session.gc_maxlifetime', 1800)", pour la durée des sessions côté serveur, et "session_set_cookie_params(1800)", pour la durée des cookies de sessions utilisateurs	P3
R8	L'accès à un compte doit être limité à un seul utilisateur simultanément. La connexion à l'application doit détruire les autres sessions existantes pour cet utilisateur.	P3
R9	Les fonctionnalités d'export mises à disposition par highcharts doivent être revues afin d'être exécutées dans la mesure du possible en local (backend BATMANAPP). Dans le cas où ce n'est pas réalisable, il est nécessaire d'envoyer les données à minima sur un canal chiffré (HTTPS). Par ailleurs, il est recommandé d'informer les utilisateurs que l'application utilise les services d'une entreprise tierce (non française) pour la génération des graphes et PDF.	P3
R10	Il est nécessaire de vérifier qu'un utilisateur à faible privilège n'a pas la possibilité d'accéder aux messages de l'administrateur	P3
R11	Les valeurs envoyées par l'utilisateur doivent être vérifiées du côté serveur. Cette fonction doit être revue avant le passage en production de l'application.	P3
R12	Cette page doit être retirée lors du passage en production de l'application	P3

4 - Détail des vulnérabilités

V1 : L'un des champs de la page de logs est vulnérable aux injections SQL

- ****

Périmètre :

Application web

Description :

Le champ « eventId » de la page des logs d'activité accessible depuis le profil administrateur applicatif est vulnérable aux injections SQL. Un utilisateur peut exploiter ce champ pour exécuter des commandes directement sur la base MySQL.

```
"https://[redacted]/support/log/details?eventId=658846"
```

Figure 1 – Url et champ exploitable

```
technique test
[18:36:47] [INFO] target URL appears to have 14 columns in query
injection not exploitable with NULL values. Do you want to try with a random integer value for
[18:41:16] [INFO] GET parameter 'eventId' is 'Generic UNION query (NULL) - 1 to 20 columns' in
[18:41:16] [WARNING] applying generic concatenation with double pipes ('||')
[18:41:16] [WARNING] parameter length constrainting mechanism detected (e.g. Suhosin patch). P
[GET parameter 'eventId' is vulnerable. Do you want to keep testing the others (if any)? [y/N]
```

Figure 2 – Capture d'écran du résultat de l'outil sqlmap

```
available databases [5]:
[*] f4_[redacted]
[*] f4_[redacted]_raw_data
[*] f4_raw_data
[*] information_schema
[*] ordering_[redacted]
```

Figure 3 – Liste des bases de données disponibles

L'application accède à la base de données à partir d'un compte applicatif à faible privilège. L'utilisation de ce compte protège les données du moteur de base et ne nous a pas permis de devenir administrateur de la base (DBA), ou d'exécuter des commandes système.

Impact :

Cette vulnérabilité permet de récupérer l'ensemble des données de la base (login, mots de passe, informations métier). Il nous a été possible de récupérer les données concernant environ de 13000 comptes applicatifs présents sur l'application, dont 755 possédants un mot de passe (50% en clair, 50% sous forme de Hash). Cette vulnérabilité permet de compromettre (altérer ou supprimer) les données stockées en base.



Recommandation :

Il est recommandé d'utiliser des requêtes préparées (prepared statement) pour limiter les risques d'injections SQL. Par ailleurs, les champs utilisateurs doivent être encodés avant d'être utilisés par l'application.

Le langage PHP propose pour répondre à ces deux cas de figure des classes et fonctions :

- Les requêtes préparées peuvent être écrites à partir de la classe PDO (PHP Data Object) qui permet de ne pas manipuler directement des chaînes de caractères SQL.
- La fonction « htmlspecialchars() » permet d'encoder les caractères spéciaux propres au langage SQL pour les rendre non interprétable par le moteur de base.

V2 : Certains des mots de passe utilisateurs sont stockés en clair dans la base - ****

Périmètre :

Application web

Description :

Certains des mots de passe présents dans la base de données, dont ceux de nos comptes de test, sont stockés en clair.

```
[19:13:21] [INFO] fetching SQL SELECT statement query output: 'select login,password from
[19:13:22] [INFO] the SQL query used returns 1 entries
[19:13:22] [INFO] retrieved: "[REDACTED]"
select login,password from app_user where login like '%[REDACTED]%' [1]:
[*] [REDACTED]
```

Figure 4 – Capture d'écran du résultat de l'outil sqlmap

Néanmoins, il semble qu'environ 50% des comptes ne soient pas stockés avec un mot de passe en clair, mais à l'aide de la fonction de hashage bcrypt. Ce mode de stockage offre un niveau de protection adéquate des mots de passe.

L'utilisation de ces mots de passe stockés en clair nous a permis de nous authentifier à l'application au travers d'un autre compte.

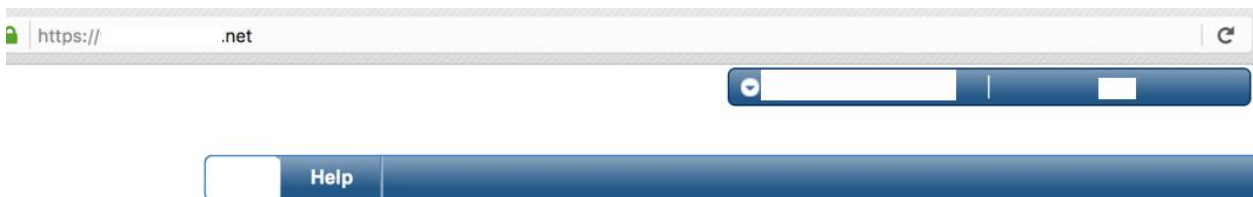


Figure 5 – Accès à la plateforme à partir d'un compte WAYNEDEVCORP

Impact :

Le stockage des mots de passe en clair dans la base ne permet pas de protéger ces informations sensibles en cas de vulnérabilité applicative permettant d'extraire des données stockées en base (Injection SQL).

L'utilisation du login d'un utilisateur et de son mot de passe stocké en clair peut permettre à une personne illégitime de s'identifier en son nom à l'application.

Recommandation :

Les mots de passe doivent être hashés et de préférence salés avant d'être insérés dans la base. La fonction "Password_bcrypt()" intégrée à PHP, combinée avec une politique de mot de passe renforcée offre un niveau de sécurité satisfaisant.

V3 : L'application ne possède pas de politique pour la gestion des mots de passe - ***

Périmètre :

Application web

Description :

Les mots de passe des utilisateurs ne sont pas sujets à une politique de mot de passe, certains utilisent le nom de l'application et l'année de création du compte, un utilisateur se sert de son identifiant de connexion également comme mot de passe.

```
Total entries = 401
Total unique entries = 372
```

Figure 6 – Nombre de mots de passe analysés

```
Top 10 base words
[redacted] = 22 (5.49%)
cgbyse = 5 (1.25%)
courbevoie = 5 (1.25%)
[redacted] = 3 (0.75%)
[redacted] = 3 (0.75%)
[redacted] = 3 (0.75%)
orange = 3 (0.75%)
password = 3 (0.75%)
cpprog = 2 (0.5%)
[redacted] = 2 (0.5%)
```

Figure 7 – Principaux mots utilisés dans un mot de passe

Par ailleurs, nous avons pu utiliser un mot de passe de 1 caractère pour nous connecter à l'application.

```
POST /login HTTP/1.1
Host:
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:46.0) Gecko/2010
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate, br
Referer: https://[redacted]/login
Cookie: locale=fr-FR; sessionid=287db6a990d6e20dfacaf76dc008a2fa
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 73

identifiant=[redacted] password=1&submit=|
```

Figure 8 – Authentification à l'aide d'un mot de passe de 1 caractère

L'application possède néanmoins un mécanisme anti-bruteforce qui permet de bloquer l'utilisateur au bout de trois tentatives infructueuses. Ce qui renforce l'accès à l'application.

En combinaison avec une politique de mot de passe suffisante, ce mécanisme peut être limité à une durée d'exclusion de 5 minutes. Ce délai reste suffisant pour garantir l'efficacité du mécanisme anti-forcebrute sans que l'utilisateur ne soit contraint de contacter un administrateur pour demander le déblocage de son compte.



Impact :

Il est envisageable d'essayer certains motifs de création des mots de passe, comme l'utilisation du nom de l'application combiné avec la date de création du compte pour obtenir des accès applicatifs. Il devient alors possible de s'authentifier en une tentative, rendant les mécanismes de protection inefficaces.

Recommandation :

Les mots de passe doivent être validés côté serveur pour satisfaire une taille d'au moins 8 caractères et utiliser 3 parmi 4 types de caractère (min, maj, num, spé). En complément une liste noire de mots à ne pas utiliser peut-être mise en place pour ne pas faire apparaître le nom des sociétés, de l'application, de l'utilisateur ou tout autre mot fréquemment utilisé (Top 10 des pires mots de passe).

V4 : La base de données est mutualisée entre toutes les sociétés utilisant BATMANAPP - ***

Périmètre :

Base de données

Description :

Lors de l'exploitation de la vulnérabilité Vo1 (injection SQL) permettant d'accéder aux données stockées en base, il est possible de récupérer les données de nombreux clients de BATMANAPP, en particulier les identifiants de connexions des utilisateurs, les commandes passées, les identifiants téléphoniques des employés. La table User contient plus d'1 million d'entrées.

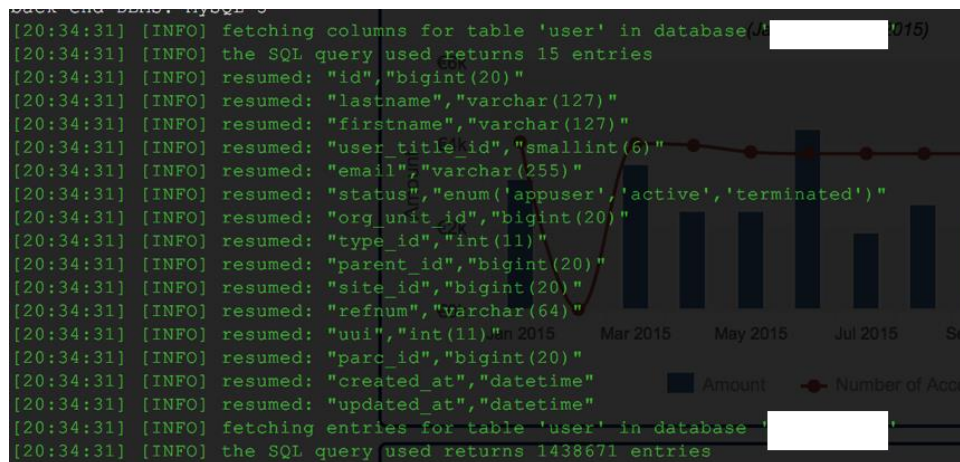


Figure 9 – La lecture de la table "user" a retourné plus d'1 million d'utilisateurs

Impact :

Le compte applicatif d'accès à la base de données à la possibilité de lire les données de l'ensemble des sociétés. La mutualisation de ces données ne permet pas de se protéger contre un vol massif de données. L'exploitation d'une faille à partir du compte d'une société met à risque l'ensemble des sociétés.

Recommandation :

Afin de limiter les risques en cas d'exploitation de vulnérabilité applicative, les données propres à une société ne doivent pas pouvoir être accédées par une autre.

Il est préférable de créer des bases propres aux sociétés (hébergées au sein d'une même instance MySQL), l'application y accédant au travers d'un compte applicatif propre à la société et possédant des droits de lecture et écriture sur sa base et de manière extrême en lecture seule sur les données de la base BATMANAPP.

V5 : Des failles XSS sont présentes dans plusieurs pages du site - ***

Périmètre :

Application web

Description :

Des failles XSS sont présentes dans certains champs utilisateurs.

Le champ « sort » présent dans de nombreuses pages de référence (mobile, utilisateur, panier, message, articles, détails d'appels, ...) n'est pas correctement encodé avant son affichage et permet d'injecter du code Javascript qui sera exécuté dans le navigateur du client. Cela est dû au fait que cette variable, passe en paramètre de la requête GET, et est réaffichée par la suite dans la page.

La valeur servant d'exemple est la suivante :

```
!&sort="">><script>alert('XSS')</script><"!
```

Figure 10 – Paramètre "sort" vulnérable ajouté à la requête GET

Ce champ a été retrouvé dans les modules suivants :

- /admin/messages ; contracts ; custom-fields ; custom-views ; import-export ; parcs ; profile ; site ; workflow-attachment/...
- /enduser/messages/...
- /order/cart ; orders/...
- /reporting/dashboards ; files ; interactive ; invoice ; query ; static ; template ; usage ; usage-cdr/...
- /resource/access-profile ; device ; policy-agreement ; usage-profile/...
- /support/log

Cette faille est persistante dans l'outil de messagerie internet de l'application (testé sur /enduser/messages/), ce qui la rend plus importante, car elle sera rappelée à chaque fois que l'utilisateur effectue un GET de la page.



Figure 11 – Lors d'un listing, le champ "sort" est vulnérable aux XSS

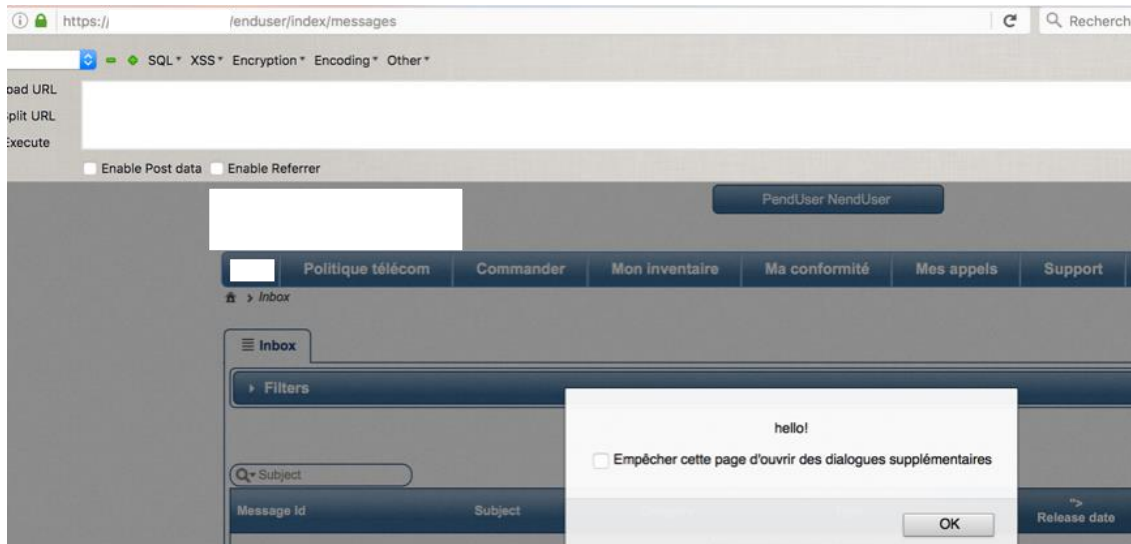


Figure 12 – L'injection XSS du champ "sort" est persistante dans la partie messagerie interne (pour l'utilisateur)

Une faille XSS peut être exploitée dans les commentaires de création de requête sous /reporting/query.

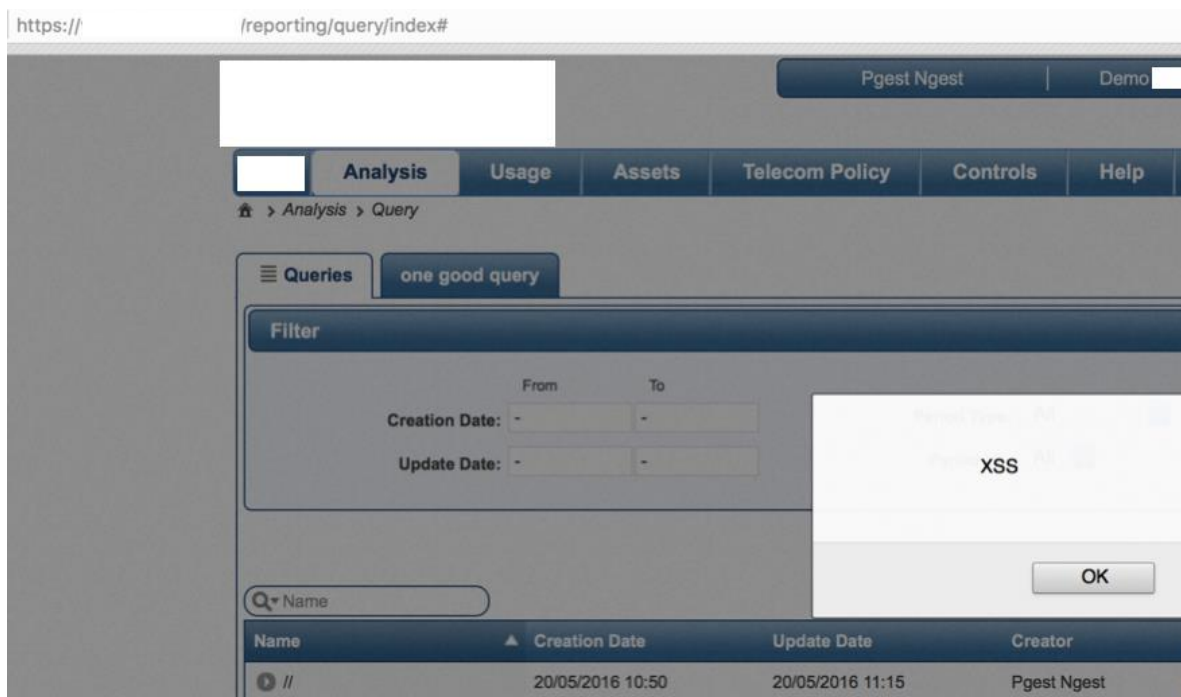


Figure 13 – Présence d'une faille XSS lors de l'ajout d'un commentaire dans l'éditeur de "query"

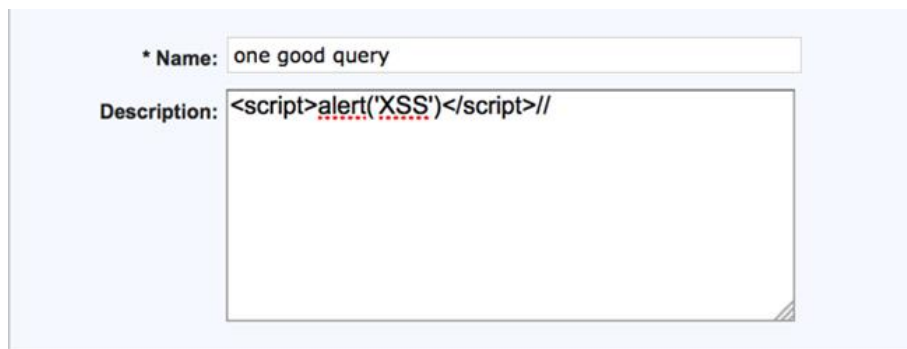


Figure 14 – L'utilisation des commentaires "//" nullifie la protection anti-XSS

L'administrateur a la possibilité de modifier la traduction des champs de l'application. Cette partie n'est pas protégée contre les failles XSS.

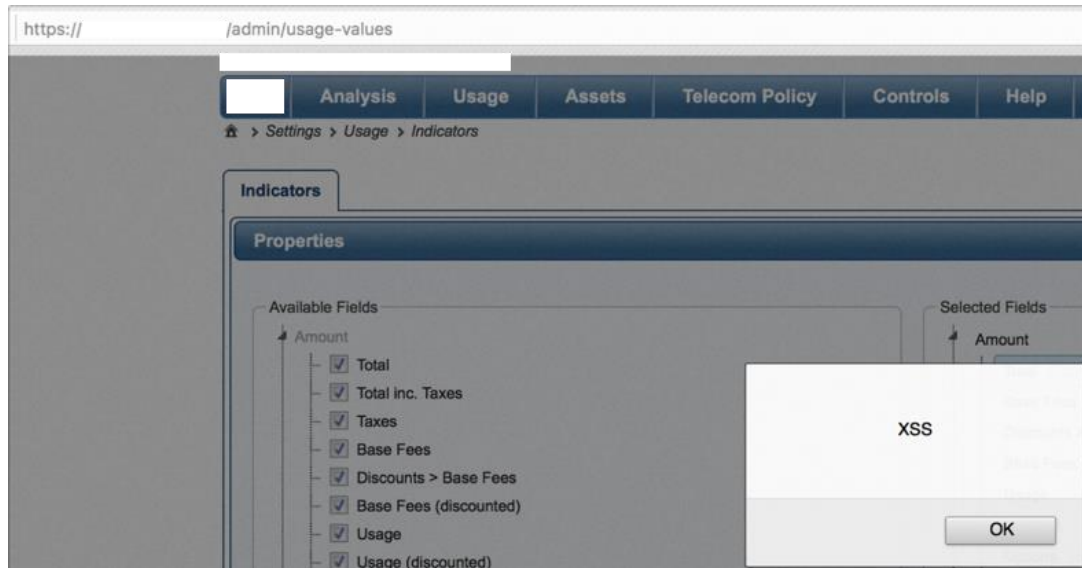


Figure 15 – Faille XSS dans /admin/usage-values



Figure 16 – Faille XSS

Impact :

L'exploitation de faille XSS peut permettre à un individu d'injecter du code javascript à l'insu d'un utilisateur et de le mener à exécuter des actions sans que celui-ci en ait conscience.

Ces actions peuvent rediriger l'utilisateur sur un autre site, lui demander de se réauthentifier pour récupérer par la suite ses identifiants, etc.

Recommandation :

Il est recommandé de filtrer ou d'encoder systématiquement les entrées utilisateurs.

PHP met à disposition différentes fonctions, telles que "htmlspecialchars()" pour permettre d'encoder les données utilisateurs transmises au serveur et de rendre l'utilisation de caractères spéciaux ("<", ">"...) non interprétables lors de leur affichage dans le navigateur.

V6 : Aucune analyse antivirus n'est effectuée lors de l'importation de fichiers par l'administrateur - **

Périmètre :

Application web

Description :

Lors de l'ajout de pièces jointes par un administrateur sur le site (exemple : ajout d'une pièce jointe à un contrat), aucune analyse antivirus n'est effectuée.

Le fichier de test est composé d'une chaîne de caractères utilisée pour tester la présence d'antivirus (EICAR) a bien été uploadé sans être effacé ou bloqué par la suite.

```
$ cat ~/Utils/antivirus.txt  
X5O!P%@AP[4\PZX54(P^ 7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
```

Figure 17 – Fichier EICAR de test pour antivirus

SHA256: ebc8e03e399a33a7a8c13de514639fa23d03d5a3739f472a75f5474da5899586
Nom du fichier : antivirus.zip
Ratio de détection : **47 / 56**
Date d'analyse : 2016-05-23 08:32:39 UTC (il y a 1 heure, 2 minutes)

Figure 18 – Analyse du fichier de test sur virustotal.com, la signature est bien reconnue par 47 des 57 antivirus présents sur virustotal.

L'ajout de ce fichier en pièce jointe d'un contrat n'a pas été interdit par le serveur, il est possible par la suite de le télécharger.

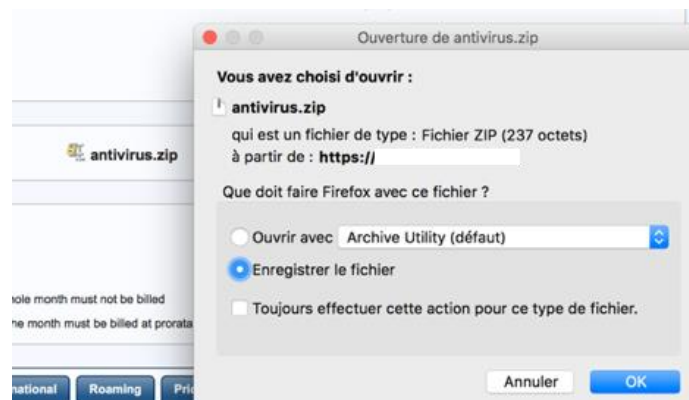
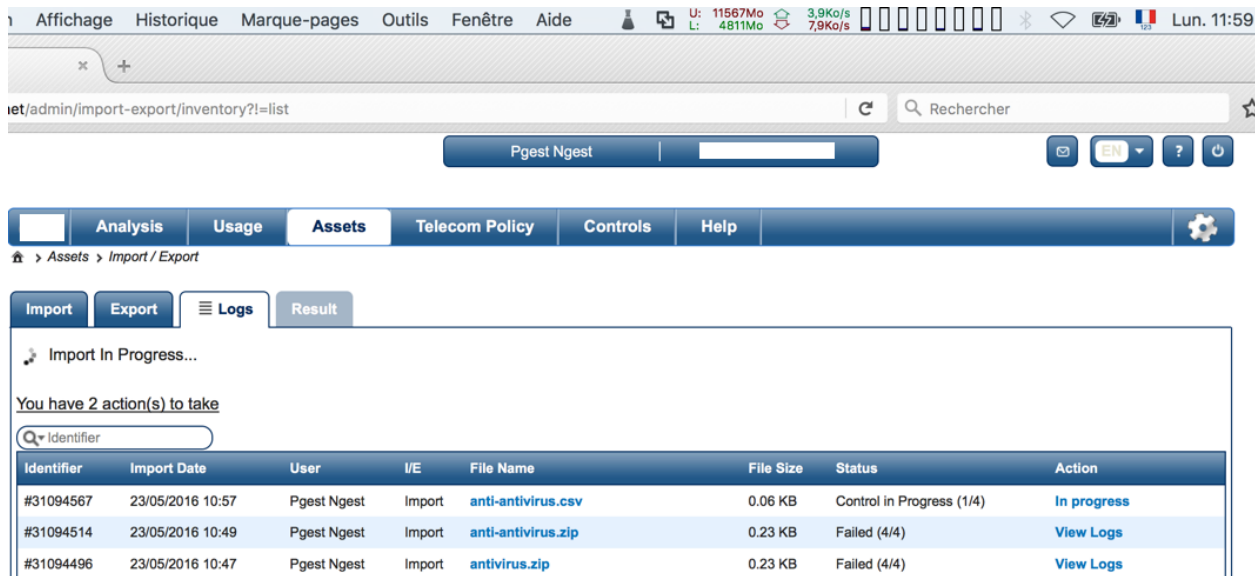


Figure 19 – téléchargement de l'archive de test

Ce test n'a pas pu être mené sur la fonction d'importation présente à l'url « admin/import-export » en raison d'erreurs ou de temps d'upload infinis.



The screenshot shows a web browser window with the URL `net/admin/import-export/inventory?!=list`. The interface includes a navigation menu with tabs for Analysis, Usage, Assets, Telecom Policy, Controls, and Help. Below the menu, there are tabs for Import, Export, Logs, and Result. The main content area displays "Import In Progress..." and a message: "You have 2 action(s) to take". A search bar for "Identifier" is present. A table lists the import actions:

Identifier	Import Date	User	I/E	File Name	File Size	Status	Action
#31094567	23/05/2016 10:57	Pgest Ngest	Import	anti-antivirus.csv	0.06 KB	Control in Progress (1/4)	In progress
#31094514	23/05/2016 10:49	Pgest Ngest	Import	anti-antivirus.zip	0.23 KB	Failed (4/4)	View Logs
#31094496	23/05/2016 10:47	Pgest Ngest	Import	antivirus.zip	0.23 KB	Failed (4/4)	View Logs

Figure 20 – "import in progress" après une heure

Impact :

Un utilisateur malveillant peut utiliser l'application BATMANAPP comme vecteur de diffusion d'un trojan ou malware en pièce jointe d'un document dans le but d'infecter d'autres utilisateurs. Il est cependant nécessaire de respecter le mécanisme de filtrage sur les types mime / les extensions des fichiers. Mais les archives au format .zip peuvent potentiellement permettre d'uploader des exécutables.

Recommandation :

Il est recommandé de protéger les utilisateurs en intégrant une analyse antivirus des fichiers uploadés contre d'éventuelles attaques virales. Ce mécanisme peut être effectué en amont (brique de filtrage sécurité type Firewall Applicatif WAF) ou directement à l'aide d'un antivirus type ClamAV sous Linux. Par ailleurs, la détection de pièce jointe malveillante doit faire l'objet d'une alerte de sécurité.

V7 : En cas d'inactivité, la session reste active pendant plus d'une heure -

**

Périmètre :

Application web

Description :

Les sessions utilisateurs ne sont pas détruites après plus d'une heure d'inactivité.

2414	https://	GET	/js/_cache/22a6ef0437578902797ceb2c15c691e5.js			200	14:16:37 20...
2413	https://	GET	/support/log			200	14:16:35 20...
2412	https://	GET	/js/_cache/22a6ef0437578902797ceb2c15c691e5.js			200	12:19:50 20...
2410	https://	GET	/support/log			200	12:19:15 20...

Figure 21 – Capture de deux requêtes faites à 2 heures d'intervalle avec la même session

Impact :

Il est peu commun qu'un utilisateur demande explicitement à se déconnecter d'une application web avant de fermer son navigateur web, et la fermeture du navigateur ne permet pas non plus de supprimer la session de l'utilisateur.

Sa session reste alors active pendant la durée maximale autorisée par l'application.

Dans le cas où l'utilisateur s'absenterait de son poste de travail, ou dans le cas où son cookie de session serait compromis, un autre utilisateur peut utiliser la session toujours existante de l'application.

Recommandation :

Il est recommandé de diminuer la durée des sessions utilisateurs afin de respecter la politique de sécurité Wayne Enterprises. Habituellement ce paramètre est positionné pour une valeur inférieure à 30 minutes.

PHP met à disposition les fonctions "ini_set('session.gc_maxlifetime', 1800)", pour la durée des sessions côté serveur, et "session_set_cookie_params(1800)", pour la durée des cookies de sessions utilisateurs.



V8 : Plusieurs utilisateurs peuvent se connecter simultanément avec un même compte - **

Périmètre :

Application web

Description :

Lors de nos tests, nous avons eu la possibilité d'authentifier plusieurs navigateurs à un même compte utilisateur.

Impact :

Une personne malintentionnée peut parvenir à récupérer le cookie de session d'un utilisateur actif. Cette personne va alors se connecter à l'application sans que l'utilisateur légitime en soit informé.

La seconde session peut alors rester ouverte aussi longtemps que la personne le souhaite. Le changement du mot de passe utilisateur ne permet pas non plus de déconnecter la seconde session.

Recommandation :

L'accès à un compte doit être limité à un seul utilisateur simultanément. La connexion à l'application doit détruire les autres sessions existantes pour cet utilisateur.

V9: Lors de la génération des graphiques en image/pdf, "highcharts" récupère les données via canal non chiffré - **

Périmètre :

Application web

Description :

Lors d'une demande de génération de graphique aux formats pdf et images, la bibliothèque javascript highcharts envoie les données à son backend pour permettre la génération du document. De plus, ces données sont envoyées sur un canal de communication non chiffré.

```
r.exporting = {
  type: "image/png",
  url: "http://export.highcharts.com/",
  buttons: {
    contextButton: {
      menuClassName: "highcharts-contextmenu",
      symbol: "menu",
      _titleKey: "contextButtonTitle",
      menuItems: [{
        textKey: "printChart",
        onclick: function() {
          this.print()
        }
      }, {
        separator: !0
      }, {
        textKey: "downloadPNG",
        onclick: function() {
          this.exportChart()
        }
      }
    ]
  }
}
```

Figure 22 – Les données sont envoyées à l'url export.highcharts.com

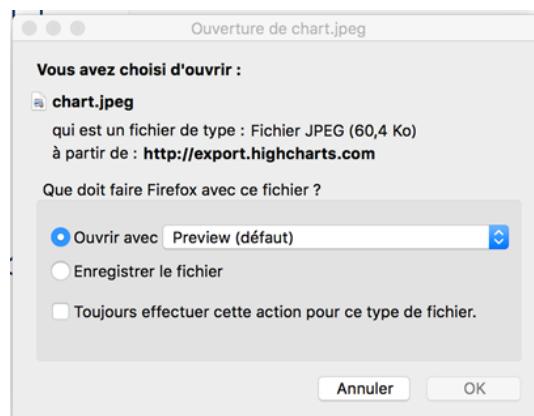


Figure 23 – La réponse est émise par export.highcharts.com

Impact :

Lors de la génération d'un document, les données contenues dans le graphique sont envoyées à l'url « export.highcharts.com ».

Highcharts a la possibilité de stocker ces données, potentiellement sensibles bien que non nominatives.

Recommandation :

Les fonctionnalités d'export mises à disposition par highcharts doivent être revues afin d'être exécutées dans la mesure du possible en local (backend BATMANAPP). Dans le cas où ce n'est pas réalisable, il est nécessaire d'envoyer les données à minima sur un canal chiffré (HTTPS). Par ailleurs, il est recommandé d'informer les utilisateurs que l'application utilise les services d'une entreprise tierce (non française) pour la génération des graphes et PDF.

V10 : Notre utilisateur de moindre privilège à pu se connecter à la messagerie "admin/messages" - *

Périmètre :

Application web

Description :

Le compte « utilisateur final » à le droit d'accès à la page « admin/messages ».

La messagerie interne n'étant pas activée lors de nos tests, nous ne pouvons pas porter de conclusion sur cette constatation.



Figure 24 – "admin/messages" accessible par "PEndUser"

Impact :

Si cette page permet bien de lister les messages adressés à un administrateur, alors un utilisateur final serait en mesure de les lire.

Recommandation :

Il est nécessaire de vérifier qu'un utilisateur à faible privilège n'a pas la possibilité d'accéder aux messages de l'administrateur

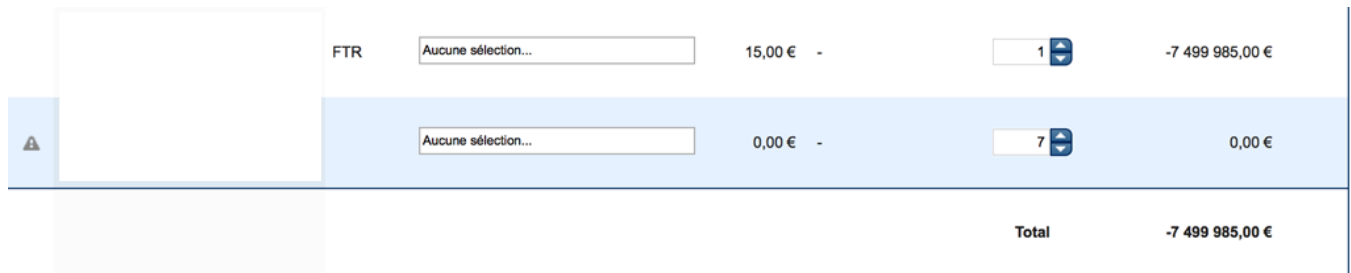
V11 : Un utilisateur peut, lors d'une commande, manipuler des quantités négatives pour obtenir des montants négatifs - *

Périmètre :

Application web

Description :

Un utilisateur a la capacité d'entrer des montants négatifs dans l'application. Il n'est néanmoins pas possible de prouver que la somme sera bien remise à l'utilisateur une fois sa commande passée.



The screenshot shows a table with the following data:

FTR	Aucune sélection...	15,00 € -	1	-7 499 985,00 €
	Aucune sélection...	0,00 € -	7	0,00 €
Total				-7 499 985,00 €

Figure 25 – Total négatif lors d'un passage de commande

Impact :

Les valeurs stockées en base pouvant être par la suite traitées de manière automatisée, cette faille peut être utilisée dans des tentatives de détournement de fonds.

Recommandation :

Les valeurs envoyées par l'utilisateur doivent être vérifiées du côté serveur. Cette fonction doit être revue avant le passage en production de l'application.

V12 : Une page intitulée "/tests" est présente dans l'application - *

Périmètre :

Application web

Description :

Lors de nos tests, une page « /tests » a été retrouvée. Les tests y sont désactivés.

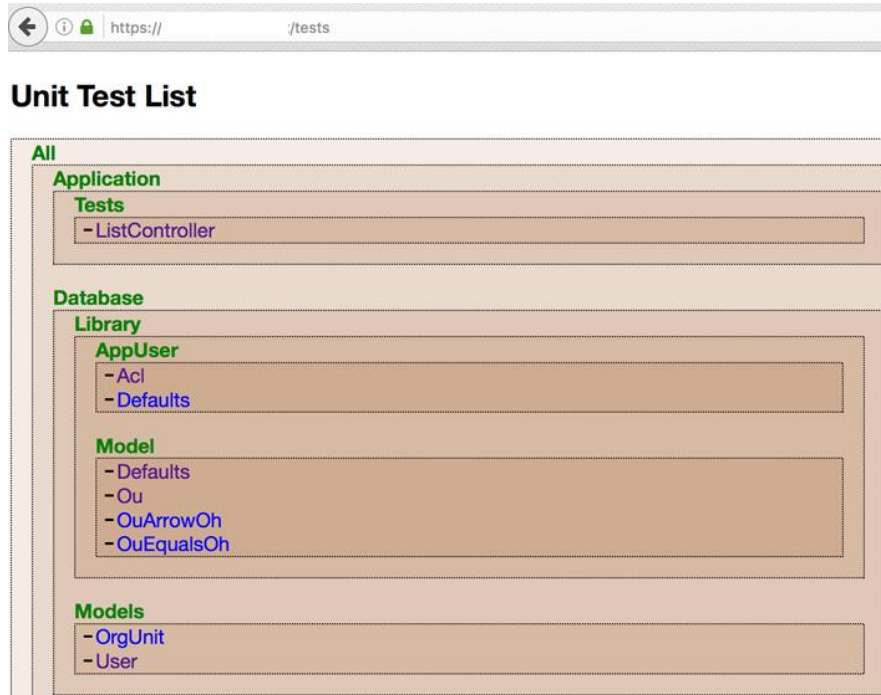


Figure 26 – Présence d'une page de tests

Impact :

Les fonctionnalités de cette page n'étant pas activées, cette vulnérabilité ne présente aucun impact.

Recommandation :

Cette page doit être retirée lors du passage en production de l'application



WWW.MACYBER.FR
CYBERSÉCURITÉ - AUDIT - PENTEST

MA Cyber
17 Boulevard Garibaldi
750015 Paris
+33(0) 7 66 63 04 51
www.macyber.fr