La sécurité sous Drupal

Tris Acatrinei

Adrien Urban dit Ze

Avant-propos

Ce mini-manuel s'adresse prioritairement à celles et ceux qui font leurs premiers pas avec Drupal, les développeurs à la souris, les débutants, qui souhaitent travailler Drupal sans pour autant être des brutes en PHP ni être des experts chevronnés en matière de sécurité.

Il peut également servir de rappel pour celles et ceux qui sont déjà familiarisés avec Drupal sans pour autant avoir la sécurité de leur plateforme comme priorité ou objectif.

C'est surtout un retour d'expériences, de situations vécues, compilées en un seul endroit et qui peut être amélioré.

Enfin, ce mini-manuel est entièrement dédié à l'univers de Drupal 7. Le volet particulier de Drupal utilisé dans le cadre du e-commerce ne sera pas particulièrement traité, même si certaines solutions évoquées peuvent tout à fait s'y appliquer. De la même manière, les expériences ont été faites sur des plateformes Drupal hébergées sur des serveurs mutualisés.

Par ailleurs, il n'a pas pour vocation à remplacer l'excellent travail de Cracking Drupal mais plutôt à venir en complément ou en initiation.

Juin 2013

Table des matières

La notion de sécurité

Les critères de la sécurité

Lorsque l'on parle de sécurité informatique, on entend majoritairement :

- Disponibilité ;
- Intégrité ;
- Confidentialité.

De ces trois critères découlent la notion d'authentification ou de véracité et la non-répudiation. Ces critères s'appliquent aussi bien au domaine de la sécurité physique, de l'exploitation, logique et applicative qu'aux infrastructures des télécommunications. Concernant Drupal, on va surtout s'intéresser à la sécurité de l'infrastructure, c'est-à-dire celle qui concerne le développement de solutions logicielles pertinentes.

De là, si on devait transposer ces critères en question concernant une plateforme sous Drupal, on pourrait se demander si :

- Ma plateforme est-elle suffisamment bien construite pour résister et être accessible en permanence ?
- Ma plateforme est-elle suffisamment bien construite pour ne pas que les données qui y sont insérées ne soient altérées ou détruites ?
- Ma plateforme est-elle suffisamment bien construite pour protéger les données contre une divulgation non autorisée ?
- Ma plateforme est-elle suffisamment bien construite pour que l'identification et l'authentification des utilisateurs soient correctes et que les accès soient adéquats ?

Les dommages directs et collatéraux d'un défaut de sécurité

On sait que la sécurité des sites Web – qu'ils soient construits avec Drupal ou avec un autre framework ou CMS – n'est pas la priorité, ni des entreprises ni de leurs clients, y compris dans le secteur bancaire. Le design, la portabilité, la lisibilité, des blocs de publicité ou encore le SEO sont les demandes les plus récurrentes, au détriment du bons sens.

Or, cela coûte plus cher de ne pas sécuriser une infrastructure dès le départ. En effet, outre l'altération des données, il y aura la perte de confiance des clients et si la faille exploitée a été diffusée – notamment sur les réseaux sociaux – il faudra également prendre en compte les autres tentatives d'attaques qui suivront. Il existe un phénomène de longue traîne ou de fonction hyperbolique en la matière. Une personne ou une entité attaque une infrastructure, publie la méthode et les résultats de l'attaque et d'autres personnes tentent à leur tour d'exploiter cette même faille voire d'en découvrir d'autres.

Une faille de sécurité peut donc avoir un impact sur l'intégrité d'une infrastructure, sur son marketing et sur sa communication. Par ailleurs, plus l'infrastructure est importante ou connue, plus l'impact sera négatif et difficile à endiguer. A titre d'exemple, tout le monde a encore à l'esprit l'attaque de Sony datant de 2011.

Par ailleurs, en raison du fait que de plus en plus d'agences et de développeurs ont recours aux CMS pour leurs sites Web, les failles sont rapidement propagées. Certaines bases de données aisément accessibles, recensent quotidiennement les nouvelles vulnérabilités.

En raison de sa popularité, Wordpress est le CMS qui fait l'objet du plus grand nombre d'attaques et de publications, suivi de près par Joomla et ensuite Drupal. En ce qui concerne le e-commerce, la répartition est plus éclatée. Mais le point commun de tous ces frameworks est leur langage de base : le PHP. Langage de développement Web dynamique, il est celui qui est le plus couramment utilisé, notamment par les CMS.

On l'aura compris : utiliser une structure commune, largement étudiée, disséquée, basé sur un langage populaire renforce la vulnérabilité d'une plateforme. Ce n'est pas pour autant une raison valable pour ne pas y recourir, surtout lorsqu'un peu de bon sens et quelques recherches suffisent à enrayer une majeure partie des difficultés.

Les failles les plus courantes

Néanmoins, il convient d'être absolument clair sur un point : toutes les protections existantes ne pourront rien contre un 0day. Ce que l'on nomme 0day en sécurité informatique désigne une faille de sécurité inconnue, non recensée, non détectée ou non divulguée. De ce fait, elle ne peut être corrigée ou patchée.

Certaines failles sont tout à fait courantes et évitables, comme les injections. Pour l'année 2013, le collectif OWASP a établi un top 10 des failles de sécurité les plus courantes :

- L'injection notamment SQL mais pas uniquement ;
- Le vol de session/mot de passe ;
- Les XSS ;
- La visualisation des données dans une requête HTTP ;
- Les paramètres par défaut des outils utilisés ;
- L'absence de protection des données sensibles ;
- L'absence de contrôle d'accès aux éléments sensibles ;
- Les CSRF ;
- L'utilisation de components tiers ;
- Les redirections invalides.

L'injection de code peut se faire soit dans une URL soit dans un formulaire et consiste à insérer du code de façon à ce que le système l'interprète et ressorte des informations censées rester secrètes.

Le vol de session/mot de passe peut se faire notamment grâce aux vols de cookies et permet d'usurper l'accès à une plateforme d'un utilisateur.

La XSS ou Cross Site Scripting est le fait d'injecter du code et d'exécuter un script dans le navigateur d'une victime voire de défigurer un site.

La visualisation des données de structure d'un site ou d'une infrastructure est faisable lorsque le développeur expose une référence dans un objet, un dossier, un fichier, sans y apposer une forme de contrôle, permettant ainsi à un attaquant de manipuler les données sans avoir d'accès autorisé à ces données.

Le paramétrage par défaut n'est pas tant un problème technique qu'un problème humain. Il faut savoir qu'un grand nombre d'attaques sont perpétrées parce que les personnes laissent les paramètres par défaut de leurs matériels ou de leurs logiciels. Par exemple, l'outil dont le login/mot de passe est admin/admin.

L’absence d’accès sécurisé aux données sensibles vient souvent du fait que la plupart des applications Web ne protège pas suffisamment les données sensibles. Il faut construire une chaîne de sécurité complète, de part et d’autre, notamment avec un SSL et du chiffrement.

De la même manière, l’absence de contrôle des données uploadées fait peser des risques commerciaux et juridiques sur une entreprise.

Les CSRF ou Cross Site Request Forgery est une attaque qui oblige le navigateur d’une personne à envoyer une requête HTTP forgée, en y incluant les cookies et toutes les informations d’authentification de façon automatique, à une application Web vulnérable.

L’utilisation de components tiers telles que les librairies ou autres applications logicielles qui fonctionnent avec certains privilèges d’administration, peuvent également être source d’insécurité.

Enfin, les redirections et transferts invalides peuvent rediriger les victimes vers des pages de phishing ou contenant des malwares.

On a donc un panorama plus ou moins large des différentes vulnérabilités. Entrons dans le vif du sujet.

Les différents niveaux d'actions

Pour protéger une infrastructure, il n'y a pas de mystère : il faut se mettre à la place d'un attaquant. Précédemment, on a brièvement brossé quelles pouvaient être ses armes. Intéressons-nous maintenant aux points d'entrées.

Les fondamentaux

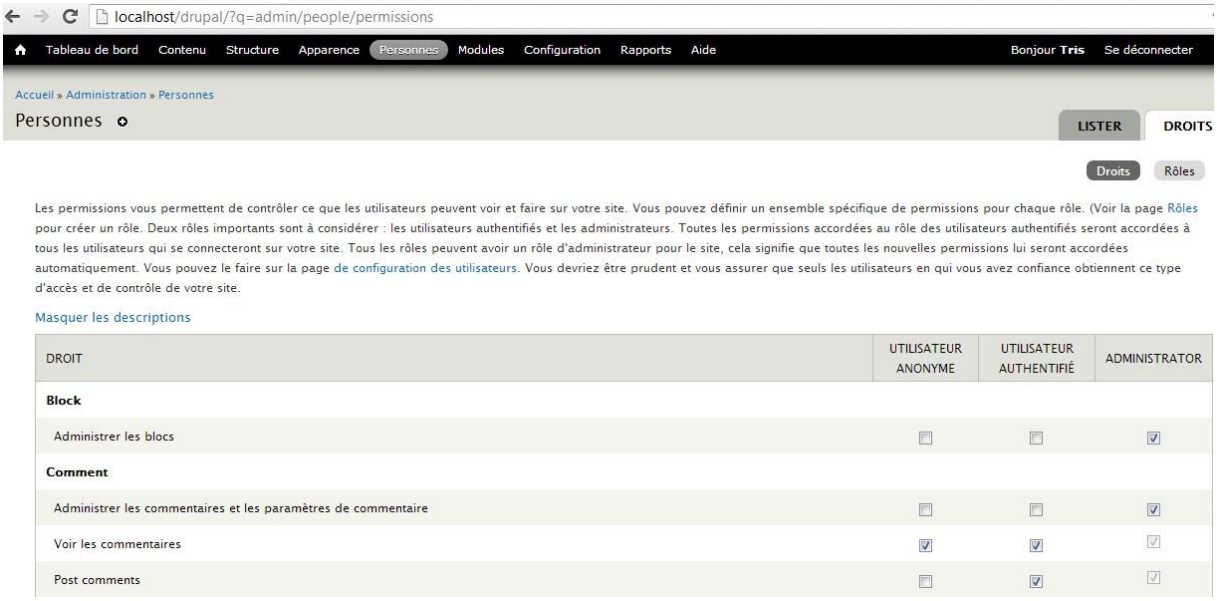
Par défaut, Drupal établit trois types d'utilisateurs :

- L'administrateur du site qui peut être l'user 1 ;
- L'utilisateur authentifié ;
- L'utilisateur anonyme.

Ces différents utilisateurs possèdent différents droits, l'administrateur ayant basiquement tous les droits et privilèges et l'anonyme en ayant le moins. L'une des premières choses à vérifier sont les formats autorisés pour les utilisateurs.

On voit également que l'on peut ajouter des rôles et en créer les droits.

Pour cela, il faut se rendre dans Personnes, onglet Droits.



A partir de là, il suffit simplement de cocher les bonnes cases, afin d'attribuer les bonnes permissions. On a les trois types d'utilisateurs par défaut dans chaque colonne et lorsque l'on ajoute un rôle d'utilisateur, une colonne supplémentaire s'ajoutera. Parlons justement de ces fameux droits.

Les permissions des utilisateurs : une histoire de format

Par défaut, il existe quatre formats de textes utilisables par les utilisateurs : le texte brut, le HTML filtré, le HTML tout court et le PHP. Le plus sage et le plus facile est de limiter l'usage des formats. Pour faire simple : l'anonyme n'a recours qu'au texte brut et l'administrateur à tous les formats. S'il y a des utilisateurs authentifiés, avec des permissions particulières notamment pour la parution des contenus, on procède au cas par cas.

Si le site est de type blog – institutionnel ou commercial – il est possible de limiter les utilisateurs authentifiés au format texte brut et HTML filtré afin d'éviter les mauvaises surprises et les utilisateurs anonymes au texte brut.

Voici un exemple de configuration :

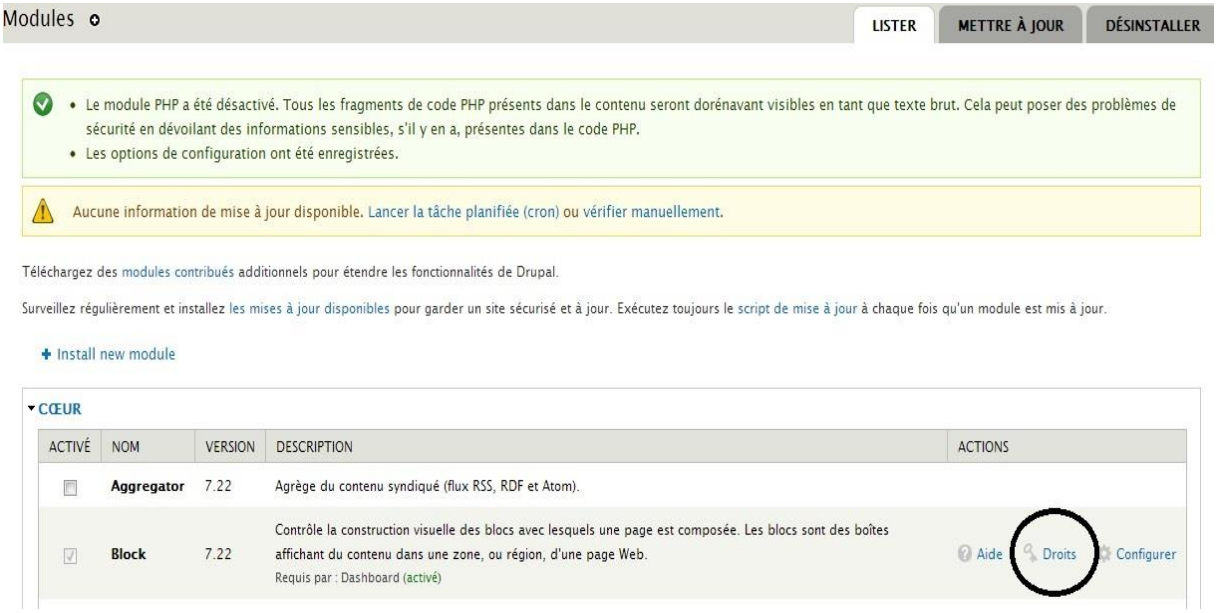
DROIT	UTILISATEUR ANONYME	UTILISATEUR AUTHENTIFIÉ	ADMINISTRATOR
Customizing the dashboard requires the Administrer les blocs permission.	☐	☐	☑
Filter			
Administrer les formats de texte et les filtres *Attention : ne l'attribuer qu'à des rôles de confiance ; cette permission touche à la sécurité.*	☐	☐	☑
Utiliser le format de texte Filtered HTML *Alerte : ce droit peut avoir un impact sur la sécurité en fonction de la manière dont le format de texte est configuré.*	☐	☑	☑
Utiliser le format de texte Full HTML *Alerte : ce droit peut avoir un impact sur la sécurité en fonction de la manière dont le format de texte est configuré.*	☐	☐	☑
Utiliser le format de texte PHP code *Alerte : ce droit peut avoir un impact sur la sécurité en fonction de la manière dont le format de texte est configuré.*	☐	☐	☑

Ici, nous avons choisi de laisser les utilisateurs authentifiés, ayant donc un compte sur le site, avoir la possibilité d'utiliser du HTML filtré. Concrètement, s'ils laissent des liens dans leurs contenus ou commentaires, ils seront cliquables, ce qui ne sera pas le cas en texte brut. Par défaut, les anonymes n'ont aucun droit concernant les filtres. Cela peut sembler, non seulement anecdotique, mais également simpliste mais il s'agit d'une bonne pratique.

L'administration des modules

Sur Drupal, on différencie le core – qui compose donc le cœur de base de Drupal – et les modules complémentaires, par exemples Views qui doit certainement être l'un des modules complémentaires les plus utilisés. L'administrateur a – de base – tous les droits d'administration sur les modules. Mais s'il existe des utilisateurs authentifiés qui ont accès au back-office, le mieux est de réduire leur possibilité d'administration. Cela peut paraître très sévère mais, pour reprendre l'expression d'un ami, « un bon administrateur est un dictateur. Il décide et les autres ferment leur gueule. ».

Drupal est relativement bien fait parce qu'à chaque fois qu'on ajoute et qu'on active un module, selon le module en question, il y a non seulement une petite icône en forme de clef, nous rappelant qu'il y a des droits à définir dans la section Personnes, mais au cas où on l'oublierait, un joli bandeau, situé au-dessus de la liste des modules apparaît après l'activation.



Dans l'exemple ci-dessus, nous avons activé le module PHP et comme nous le voyons, un bandeau vert et venu nous rappeler nous devions manier la fonctionnalité avec précaution et relancer la tâche cron pour être sûr que le système est à jour.

Que se passe-t-il si on autorise certains utilisateurs authentifiés, ayant accès au back-office, à avoir accès à l'administration des modules ? On court le risque qu'ils désactivent – même de façon involontaire – certains modules essentiels au bon fonctionnement du site et entraînant la disparition de certaines fonctionnalités.

Les contenus et les commentaires

Selon le degré d'ouverture de votre site, vous devrez modérer vos commentaires d'une façon ou d'une autre. Au départ, on est dans un gentil monde bisounours et on laisse le site ouvert aux quatre vents niveau commentaires. Fondamentalement, c'est une erreur car le site, selon la façon dont son référencement est fait, va rapidement être repéré par des robots spammeurs, qui vont joyeusement polluer la plateforme. En étant confiné dans le back-office, ils feront moins de dégâts et encore moins s'ils sont cantonnés au texte brut pour le dépôt de commentaires. Mais on peut affiner encore plus les choses en utilisant les fonctionnalités de base de Drupal.

Encore une fois, on se rend dans Personnes puis l'onglet Droits et on coche les cases correspondantes.

DROIT	UTILISATEUR ANONYME	UTILISATEUR AUTHENTIFIÉ	ADMINISTRATOR
Block			
Administrer les blocs	☐	☐	☑
Comment			
Administrer les commentaires et les paramètres de commentaire	☐	☐	☑
Voir les commentaires	☑	☑	☑
Post comments	☑	☑	☑
Skip comment approval	☐	☑	☑
Modifier ses propres commentaires	☐	☐	☑

Dans la configuration de l'image ci-dessus, les anonymes peuvent poster des commentaires mais sont obligés d'attendre la modération avant que le commentaire ne soit public. Les utilisateurs authentifiés peuvent poster des commentaires sans avoir besoin d'attendre une validation et la seule personne qui peut modifier ses propres commentaires, est l'administrateur.

Les URLS

Par défaut, dans Drupal 7, vous verrez que la plupart des URL sont construites ainsi : www.site.com/?q=. Lorsque vous construirez un contenu, vous aurez alors un numéro dans l'URL. L'une des premières choses à faire est donc d'activer Clean Urls/URL sipmlifiées, qui se trouve dans le panneau de configuration.

Non seulement ce sera plus lisible pour vous et vos visiteurs mais cela évitera également de faire croire qu'il existe des possibilités d'injections de code.

Néanmoins, vous n'échapperez pas aux différents scanners de vulnérabilité. En regardant vos logs, vous verrez alors apparaître un certain nombre d'URL demandées, qui n'existent pas sur votre site, par exemple, les pages permettant d'authentification sur d'autres CMS comme Wordpress. A partir de là, vous pouvez jouer avec les redirections d'URL. Vous pouvez orienter vos visiteurs – même malveillants – sur une page 403 ou 404 ou simplement vers la page d'accueil du site ou vers le moteur de recherche. A vous de voir ce que vous estimerez le plus pertinent. Mais vous pouvez également les rediriger vers des pages extérieures à votre plateforme.

La gestion des nuisibles

Le blocage d'IP

Sous les versions précédentes de Drupal, le BanIP était un module complémentaire. A présent, il est intégré dans le core de Drupal ce qui facilite grandement la vie. Il convient donc de l'activer, de même que le syslog, tracker, trigger qui vous permettront de suivre presque à la trace ce qui se passe sur le site.

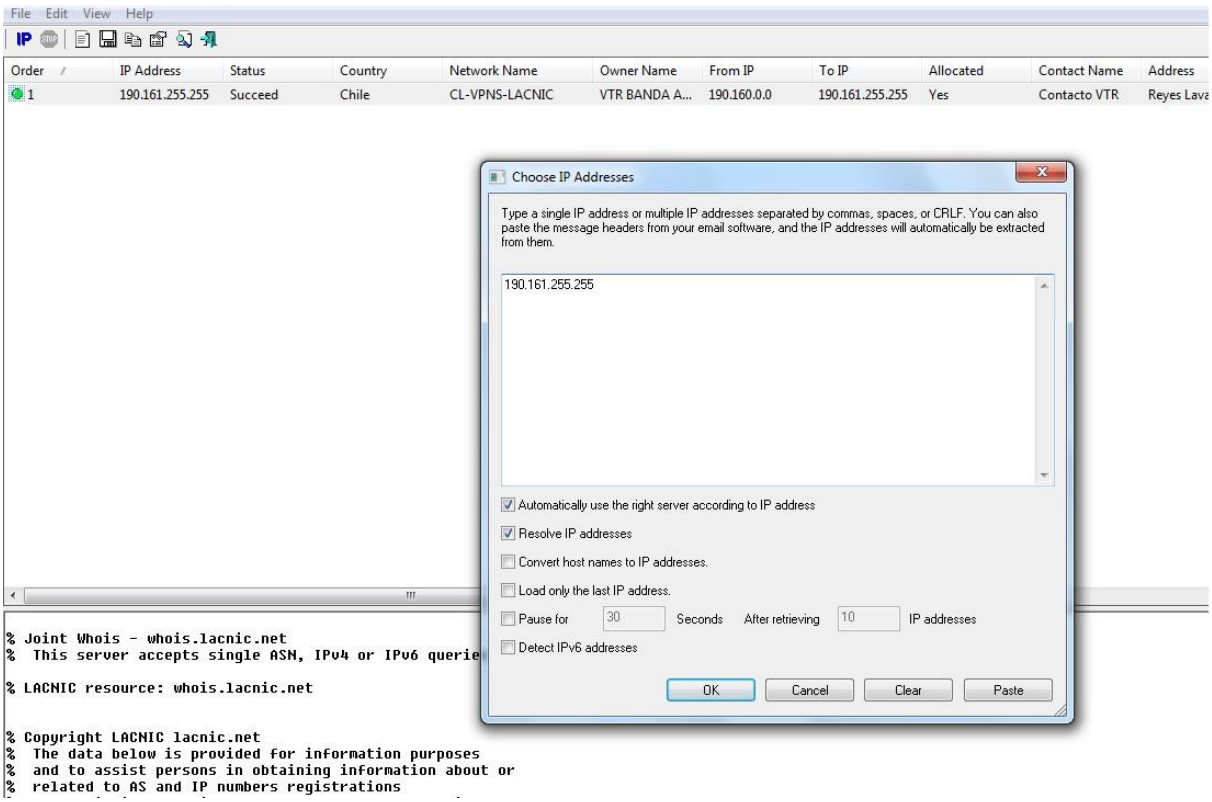
Pour le trouver, on se rend dans Configuration et sur la partie supérieure gauche, dans la section Personnes, se trouve l'option blocage d'adresse IP.

Si on souhaite se faciliter la vie, on peut repérer les rangs d'IP récurrents et les bannir. Certains robots spammeurs ont systématiquement recours aux mêmes rangs d'IP. A l'aide d'un outil de traçage comme IP NET Info, on peut déterminer la provenance géographique et l'intégrité – par regroupements d'informations – de certains visiteurs, ainsi que la plage d'IP utilisé et les bannir en bloc du site. Auquel cas, plutôt que de remplir IP par IP la table, on peut recourir à IP range bans qui permet de bannir tout une plage d'IP.

Il se trouve dans la même section que Ban IP et permet de définir des listes noires – les robots spammeurs – et des listes blanches avec les utilisateurs légitimes.

Cette méthodologie comporte néanmoins le risque de bannir des utilisateurs légitimes, il faut donc veiller à ne pas avoir la main trop lourde. Par ailleurs, si le site est multilingue, il conviendra de ne pas bannir de façon trop brutale des utilisateurs légitimes en provenance de certains pays. C'est justement à ce moment-là qu'IP Net Info permet d'y voir plus clair.

Il s'agit d'un logiciel, extérieur à Drupal, fonctionnant sous Windows, permettant d'obtenir un WHOIS complet à partir d'une adresse IP.



Avec une interface très simple, il permet d'obtenir un WHOIS complet. A partir de là, il devient plus simple de faire le « ménage ».

Les Linuxiens pourront se contenter d'ouvrir un simple terminal et de taper WHOIS 127.0.0.0 pour obtenir ces mêmes informations.

Malheureusement, que ce soit le IP Ban – dans le core – ou IP range ban – en module complémentaire – aucun d'entre eux ne gèrent les IPv6. D'après les dernières informations, la version 8 intègrera une solution permettant de gérer pleinement les IPv6.

Le spam

Comme tous les sites – sauf s'il s'agit de sites ne permettant ni contact, ni inscription, ni commentaire – vous subirez du spam. Mais vous ne serez pas totalement démuni. Sous Wordpress, Akismet fait bien le job de filtre, sur Drupal, le résultat n'est pas folichon. Mais il existe plusieurs autres systèmes bloquant totalement le spam.

Le premier est Bad Behavior qui joue les garde-fous mais ma préférence va à Honeypot.

Après son installation, la configuration se trouve dans Configuration, dans la section rédaction de contenus.

Comme vous le savez certainement, les robots spammeurs remplissent de façon automatique tous les champs qui se présentent à eux.

L'illustration ci-dessus montre un espace de commentaire tel qu'il apparaît pour un utilisateur normal et légitime.

Honeypot fournit une petite ligne supplémentaire, « invisible » aux internautes légitimes, que les robots complètent automatiquement. En effet, un utilisateur légitime va ouvrir une page du site dans son navigateur Web et même s'il ouvre cette même page dans un terminal, il verra le champ « leave this field blank ».

Le robot – et c'est ainsi qu'Honeypot le détecte – va compléter tous les champs qu'il va trouver.

Honeypot bloque alors la soumission même du commentaire et si on est farceur, on peut aussi mettre une limitation de temps dans la soumission des commentaires, bloquant toute nouvelle soumission depuis une même adresse IP pendant un certain laps de temps, que le webmaster peut configurer comme il le souhaite.

Il est à noter qu'il est utilisé sur la plateforme officielle de Drupal.org.

Le firewall

A l'origine, Fail2Ban est un outil qui se place sur le serveur, afin de détecter les activités suspectes sur ce dernier. Le module du même nom est une configuration complémentaire à cet outil. Il nécessite des droits d'administration et ne pourra pas fonctionner sur un serveur mutualisé. Le module pour Drupal est configuré pour envoyer des logs au système directement. Il va regarder et analyser ce qu'il reçoit et s'il voit un grand nombre d'erreur, il bannira automatiquement.

Si votre site n'est pas sur un serveur dédié, sur lequel vous avez complétement la main, ce module ne vous sera pas d'une grande utilité.

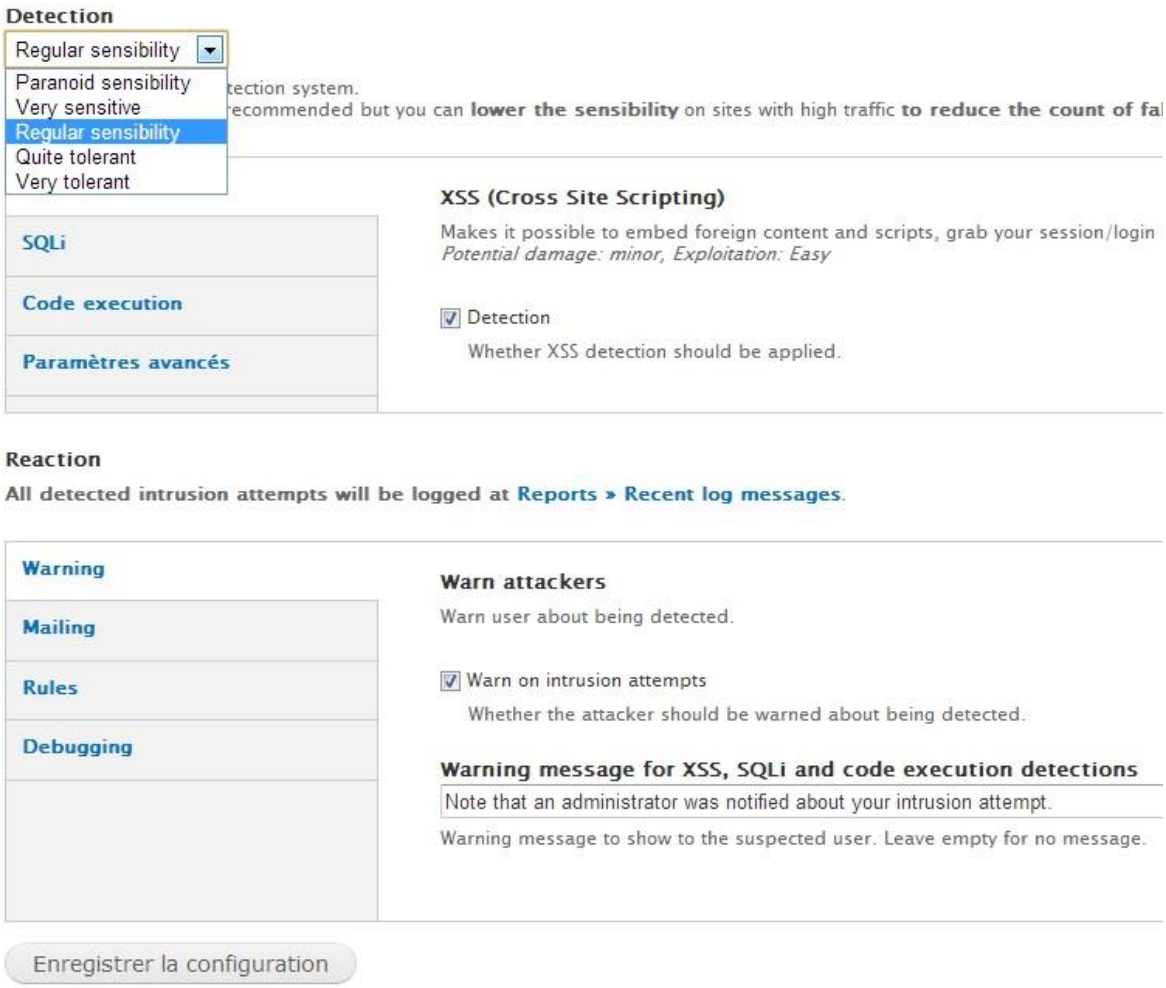
L'IDS

C'est au détour d'un cache de site qui n'est plus disponible que je suis tombée sur Tiny-IDS, un système de détection d'intrusion, adapté à Drupal.

Il n'est pas techniquement correct de parler d'IDS. Mais dans la mesure où c'est le nom choisi par les développeurs de Drupal, pour des raisons de confort, nous garderons la même terminologie.

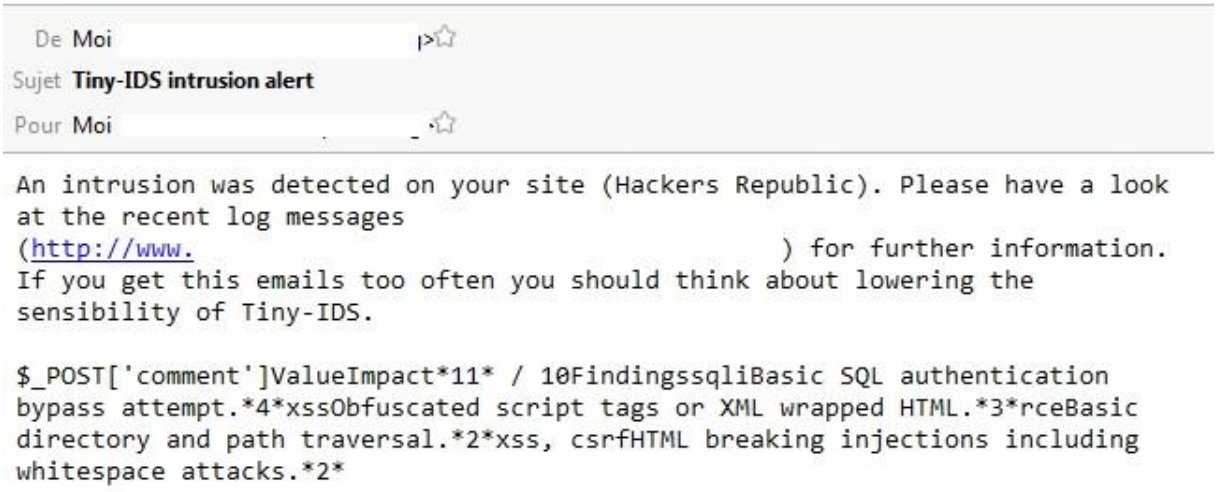
Une fois qu'il est activé, il est configurable depuis Configuration, section système sur la partie supérieure droite de votre panneau.

Il possède cinq niveaux de sensibilité et génère automatiquement un rapport par email lorsqu'une activité curieuse est détectée.



Attention avec le mode paranoïaque car l'IDS enverra alors des alertes emails pour toutes les activités du site – y compris celles qui ne sont pas nécessairement suspectes. Nous verrons dans le chapitre suivant les différents niveaux d'alertes.

Cela va sans dire mais cela va toujours mieux en le disant, il conviendra d'installer les dépendances nécessaires pour que le site puisse envoyer automatiquement les différents emails d'alerte.


De Moi
Sujet Tiny-IDS intrusion alert
Pour Moi

An intrusion was detected on your site (Hackers Republic). Please have a look at the recent log messages
(http://www.) for further information.
If you get this emails too often you should think about lowering the sensibility of Tiny-IDS.

$_POST['comment']ValueImpact*11* / 10FindingssqliBasic SQL authentication bypass attempt.*4*xssObfuscated script tags or XML wrapped HTML.*3*rceBasic directory and path traversal.*2*xss, csrfHTML breaking injections including whitespace attacks.*2*

Sur le plan de l'efficacité, il a pour avantage de bien bloquer les robots spammeurs qui essaient d'envoyer des commentaires bardés de liens publicitaires par paquets de douze. A proprement parler, il ne bloque pas réellement les véritables attaques. Ainsi, lors d'un test réalisé avec un scanner très offensif, il n'a pas spécialement réagi. Par ailleurs, lorsque le spam comporte un grand nombre d'adresses de sites Web, il va avoir tendance à les considérer comme des SQLi.

Cette première couche d'information vous permet d'avoir quelques outils reconnus, qui ont fait leur preuve. Mais cela n'est évidemment suffisant et aucun système n'est 100% étanche. Il convient donc d'acquérir une série de bonnes pratiques.

Les bonnes pratiques

On a tendance à oublier que la sécurité ne passe pas nécessairement par l'acquisition et l'utilisation d'outils coûteux mais simplement par l'apprentissage de bonnes pratiques. Que l'on parle de Web, d'administration système ou de réseaux, il existe un certain nombre de standards qui sont communs à tous.

Drupal ne fait pas exception à la règle et à l'aide de quelques modules, il est aisé de respecter les dites bonnes pratiques.

Le monitoring

Monitorer un site signifie étudier méticuleusement son activité et surtout les activités qui peuvent s'avérer dangereuse pour un site. Par défaut, Drupal propose d'inscrire dans son journal toutes les activités qui ont lieu sur une plateforme, l'administrateur n'ayant qu'à configurer le nombre de messages à sauvegarder.

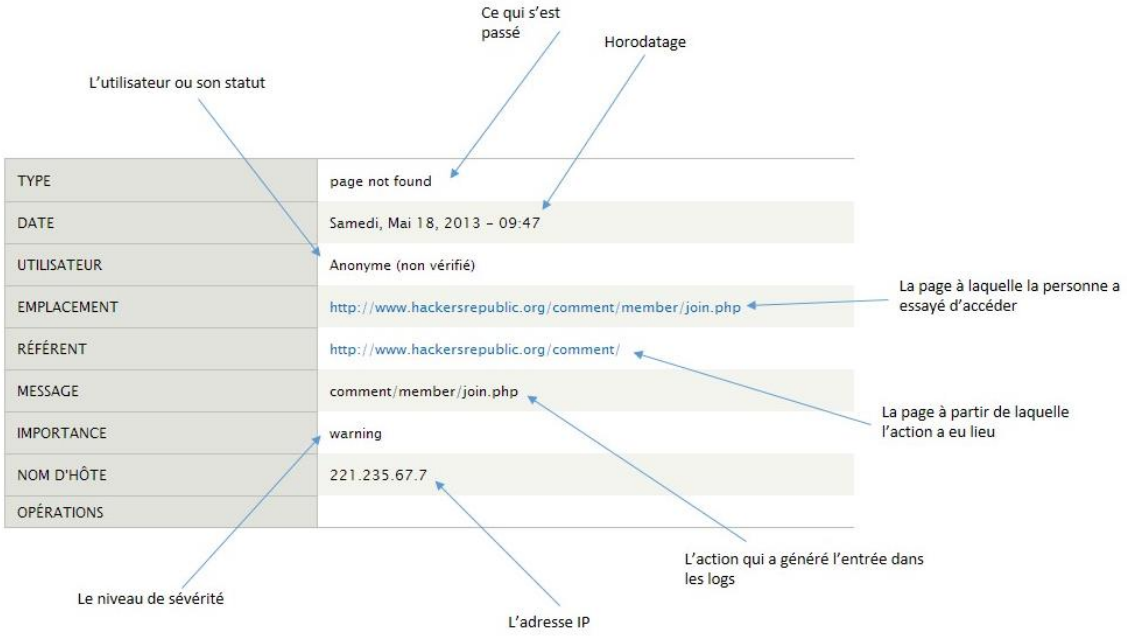
Il répertorie toutes les actions et procède à un classement des actions, selon une échelle de sévérité.

Il existe différents niveaux de sévérité dans les messages relatifs à l'administration de la plateforme, le niveau 1 étant le moins grave et le niveau 7, le plus inquiétant.

La catégorie	Le niveau
Débogage	1
Info	2
Notice	3
Warning	4
Erreur	5
Critique	6
Urgence	7

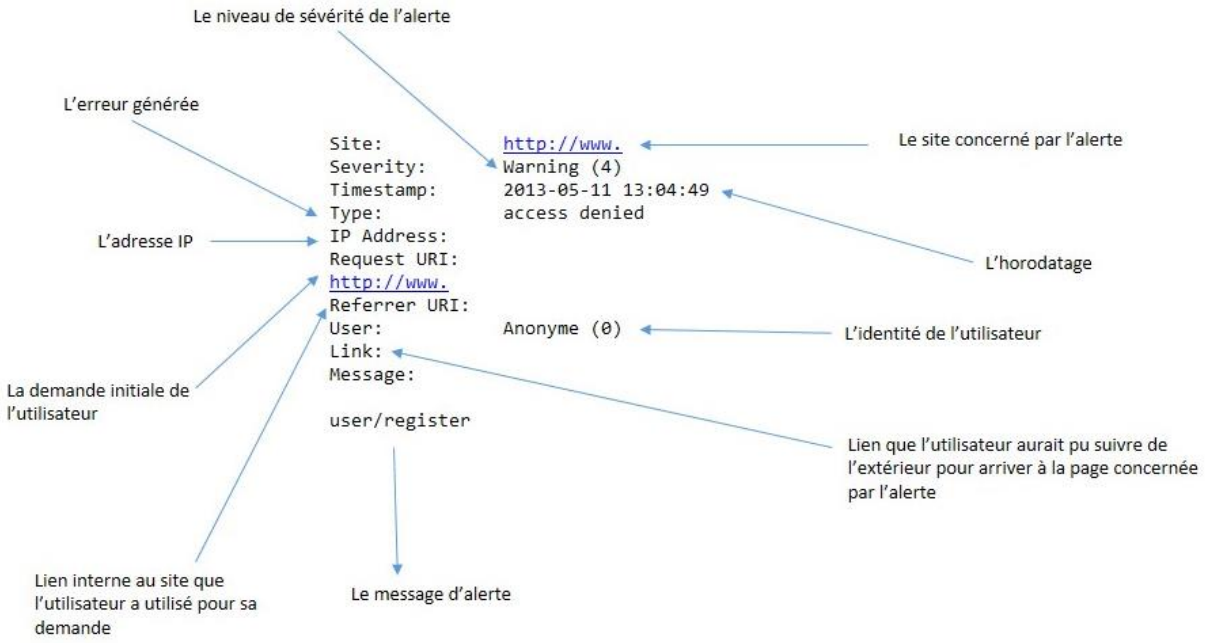
Les niveaux les plus courants – pour l'activité normale d'une plateforme sont les notices (3), les warning(4) et les erreurs (5). On les retrouve dans les logs de Drupal, sous la partie Rapports > Recents log message. Ces logs ne recensent que les aspects « problématiques » et événements qui auraient pu avoir un impact critique sur la plateforme. Les accès aux contenus sont compilés dans Recents Access.

Dans les logs, les messages sont archivés du plus récents au plus anciens et les entrées sont suffisamment détaillées pour comprendre ce qui s'est passé.



A partir de là, il convient d'être honnête avec soi-même. A-t-on le temps nécessaire pour regarder quotidiennement les activités de la plateforme ou non ? Si tel n'est pas le cas, il convient d'installer une solution de log-management. Il existe un module dans Drupal 7 qui envoie par email toutes les entrées du journal, ne l'espèce Watchdog Triggers.

Non seulement, il va reprendre les mêmes informations du journal mais il permet de suivre en temps réel ce qui se passe. Dès qu'un évènement a lieu, la plateforme envoie un email et reprend les informations principales du journal.



Il aura besoin de Trigger pour fonctionner ainsi que Mail System et HTML Mail pour les envoyer les emails.

L'avantage principal de cette fonctionnalité est le gain de temps. Au lieu d'être fixé en permanence sur le journal, on regarde les emails reçus et on prend les mesures nécessaires. Par ailleurs, en cas d'attaques, on gagne un temps précieux. En effet, la plupart des attaques sont faites à partir d'outils préexistants, de scanners, qui laissent des traces importantes dans les logs. Avec ce module, on est alerté en temps réel qu'un scanner est en train de tourner sur un site et on peut le bannir aussitôt.

Enfin, afin de ne pas être saturé, il conviendrait de dédier une boîte mail qui sera uniquement consacrée

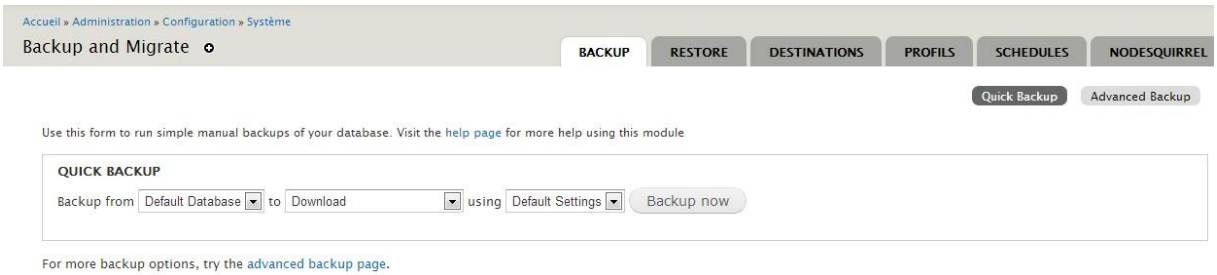
La sauvegarde et la restauration

Il peut arriver qu'un site crashe, soit parce qu'une attaque sévère a eu lieu, soit qu'une installation a été mal faite soit parce que la loi de Murphy a frappé. Quoiqu'il en soit, si une sauvegarde a été faite de façon de régulière, les dégâts peuvent être minimes. Pour cela, on se sert de deux choses : des snapshots et Backup & Migrate. Pour les snapshots, c'est relativement simple, on se tourne vers son hébergeur et on met en place un système qui va régulièrement prendre « une photo » à un moment prédéterminé de l'état d'un système, en l'espèce d'un site et surtout de sa base de données. En cas de problème, ce snapshot va permettre de restaurer rapidement un site.

Sinon, une des choses faisable est d'installer Backup & Migrate, qui est un module qui va sauvegarder toutes les informations d'un site, les organiser en bases de données et permettre des migrations très rapides mais aussi des restaurations rapides.

A la base, Backup & Migrate n'était pas nécessairement conçu dans une optique de sécurité. L'idée était de permettre de migrer rapidement les données d'une plateforme Drupal à une autre, sans que ces dernières ne soient impactées. Mais la facilité d'installation, l'efficacité de ce module et sa possibilité à programmer les sauvegardes en ont fait un outil participant à la sécurité, surtout si on se réfère à l'un des critères de la sécurité, à savoir la disponibilité des données.

Concrètement, on installe le module, puis on le configure en lui indiquant comment faire la sauvegarde et on peut également programmer des sauvegardes. Les sauvegardes seront alors disponibles dans un dossier sur le serveur. En cas de crash, il suffira de réinstaller le core de Drupal, les modules complémentaires et simplement se rendre dans configuration > système > Backup & Migrate > Restore.



L'une des dernières versions de Backup & Migrate vous permet de placer vos sauvegardes dans un service de Cloud, en l'espèce, NodeSquirrel. Sur ce point, il n'y aura pas de parti pris. Certaines personnes ont recours à ce type de services et sont très à l'aise avec d'autres. D'autres préféreront avoir un contrôle total sur leurs données, à n'importe quel moment de la chaîne, sans être dépendant d'un service.

En combinant les snapshots et Backup & Migrate, on a un filet de sécurité en cas de pépins.

Les droits d'accès des dossiers et des fichiers

Installer un Drupal, c'est bien. Eviter qu'il soit ouvert aux quatre vents, c'est mieux. Pour cela, il faut paramétrer les permissions des dossiers. Pour les développeurs à la souris, on peut faire cela en utilisant tout simplement un client FTP de type Filezilla. En piochant à droite et à gauche et en demandant à des personnes connaissant à la fois bien Drupal et la sécurité, voici un conseil concernant les permissions des dossiers Drupal :

Nom du dossier/fichier	Droit
includes	755
misc	755
modules	755
profiles	755
scripts	755
sites	755
themes	755
.gitignore	644
htaccess	644
authorize	644
cron	644
index	644
probots	644
update	644
web	644
xmlrpc	644

Enfin, supprimez les fichiers suivants :

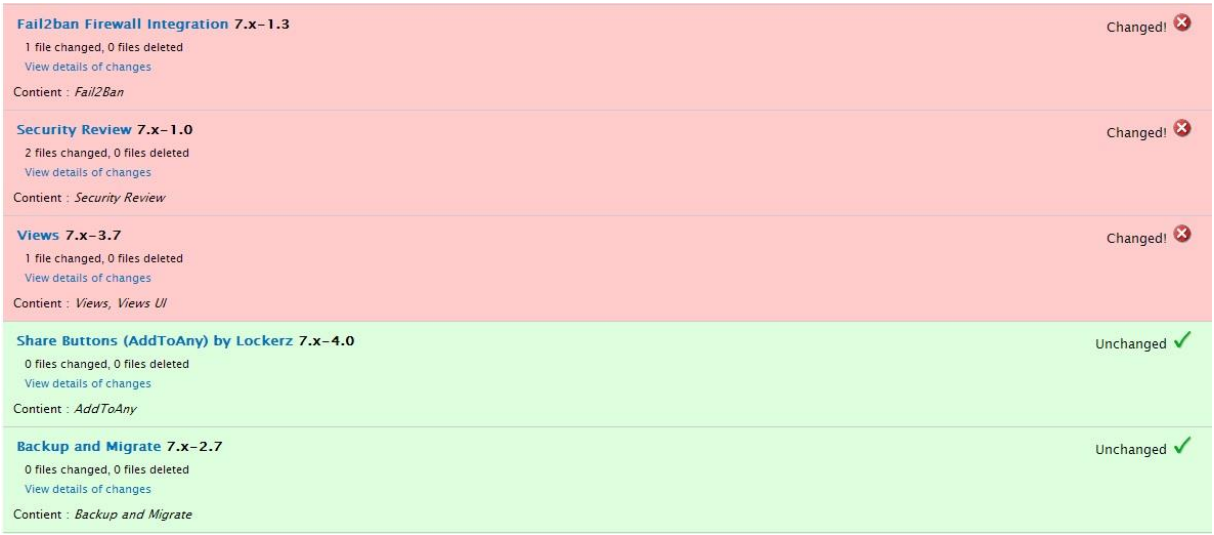
- Install.php,
- Changelog,
- Les fichiers d'installation des bases de données.
- Les modules dont vous n'avez pas besoin ainsi que les modules que l'on utilise en phase de développement comme simpletest.

On n'oubliera pas de cocher la case « appliquer ces droits aux sous-dossiers ». Quoiqu'il en soit, on vérifie qu'aucun dossier ou fichier n'a pour droit le 777.

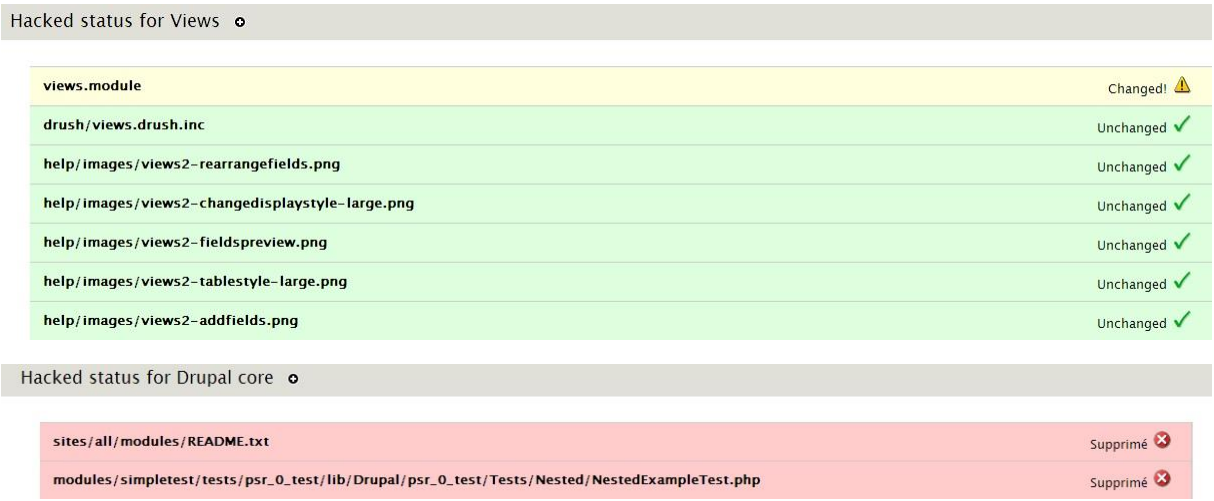
Le monitoring des dossiers et des fichiers

Il existe un module complémentaire qui peut vous aider au quotidien pour voir si des dossiers ou des fichiers ont été modifiés, il s'agit de Hacked. Il porte relativement mal son nom mais remplit bien sa fonction. Il ne vous avertit pas si vous avez fait l'objet d'une intrusion mais vous signale si un fichier appartenant à un module, de core ou complémentaire, a été modifié. Cela demande évidemment de se rappeler ce que l'on a corrigé mais cela peut-être utile d'autant qu'il fournit le détail des éléments qui ont été modifiés. On peut tenter de combiner ce module avec une règle personnalisé mais il faut savoir que cela risque de ralentir sévèrement le chargement du site.

Une fois le module installé et activé, allez dans Rapports puis cliquez sur l'onglet Hacked. Il fera alors une revue complète des éléments du site et vous fournira un rapport assez complet facile à comprendre.



Sur la page principale du rapport, on obtient la liste de tous les modules qui ont été modifié. On observe qu'un lien est inséré pour permettre de visualiser les changements.



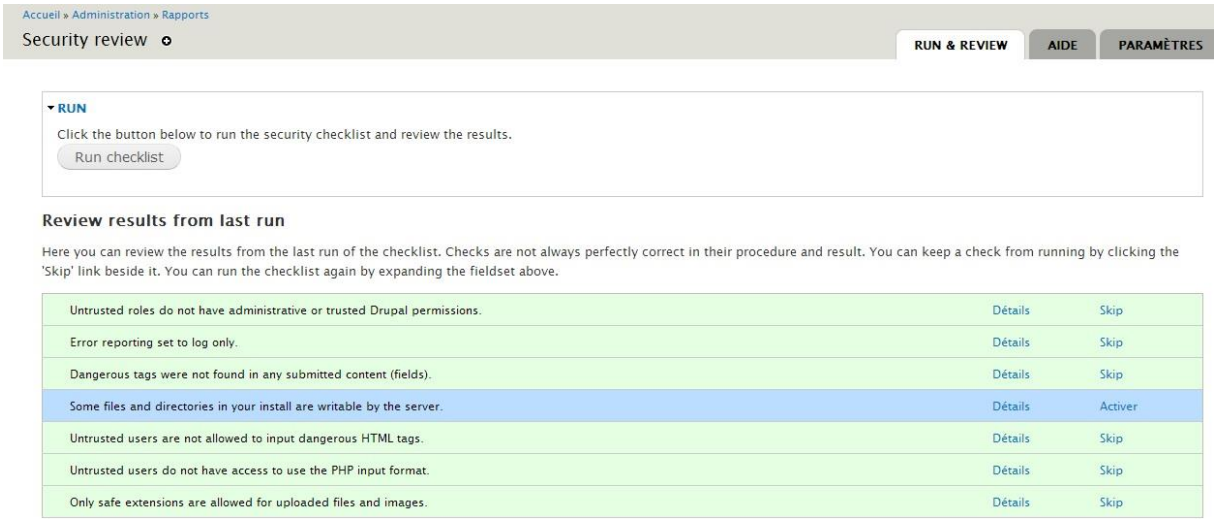
Le code couleur est relativement simple : vert lorsque rien n'a été modifié, jaune lorsqu'il y a un changement et rouge lorsqu'un élément a été supprimé.

La modification peut être une correction dans le code.

L'autre module qui peut venir en complément de Hacked est Security Review.

En effet, il vérifie un certain nombre de points dont les rôles, les erreurs, les fichiers, les formats autorisés en fonction des rôles.

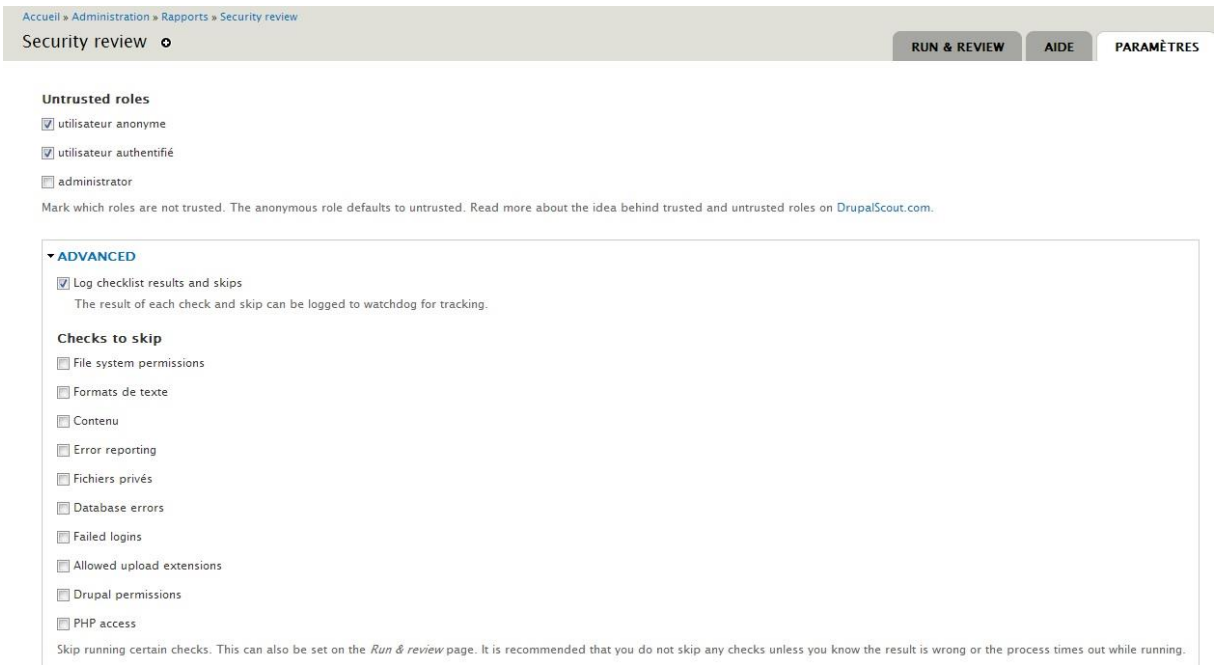
Tout comme Hacked, une fois le module installé et activé, vous le trouverez dans Rapports.



Comme on peut le voir, certains points peuvent être enlevés de la checklist.

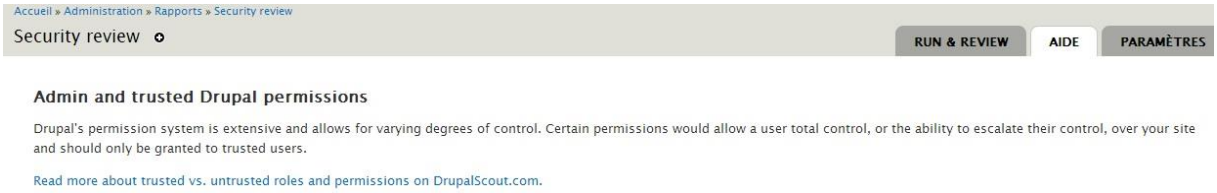
En cliquant sur détails, vous obtiendrez les informations qui vous permettront de corriger ce qui est considéré comme une faille par l'outil.

L'onglet paramètres vous permettra de personnaliser les éléments que vous souhaitez ne pas prendre en compte lors de votre analyse.



Sa faiblesse réside dans le fait que s'il vérifie effectivement un certain nombre de points, il demande de bonnes connaissances pour comprendre la façon dont les problèmes peuvent être résolus. En effet, il peut induire en erreur un utilisateur. Dans la revue de sécurité opérée ici, l'accès aux fichiers a été supprimé car si on suit les recommandations préconisées, on supprime tout simplement la possibilité à la plateforme de fonctionner correctement.

Néanmoins, il est un outil très pédagogique car en cliquant sur Détails dans le rapport, on obtient une page qui explique brièvement l'importance de l'élément analysé ainsi qu'un lien vers de la documentation officielle Drupal.


Accueil » Administration » Rapports » Security review
Security review
RUN & REVIEW
AIDE
PARAMÈTRES
Admin and trusted Drupal permissions
Drupal's permission system is extensive and allows for varying degrees of control. Certain permissions would allow a user total control, or the ability to escalate their control, over your site and should only be granted to trusted users.
Read more about trusted vs. untrusted roles and permissions on DrupalScout.com.

L'identification et l'authentification

Le certificat SSL

Selon le degré de sensibilité du site, mettre en place un certificat SSL et donc la possibilité de se connecter en https peut être essentiel. A ce stade, vous avez deux solutions : le faire vous-même ou vous tournez vers votre hébergeur. Si ces questions sont obscures, laissez la main à votre hébergeur qui s'en chargera. Pour autant, il n'est pas toujours pertinent de forcer la connexion en https. En effet, dans certaines entreprises ou administrations, il ne serait pas possible de se connecter sur les sites en https. Si vous forcez les connexions en https, vous perdrez du trafic. Sur ce point, tout dépend du but du site, de sa fonction et de son public.

Néanmoins, vous pouvez mélanger les deux, par exemple, laisser le choix de la connexion à l'utilisateur et forcer le https uniquement pour les connexions à l'espace d'authentification. Le module Secure Login permet ce type de configuration. Vous pouvez également utiliser Secure Pages qui vous permettra de déterminer un certain nombre de pages uniquement accessibles en https, ce qui peut être intéressant si le site est développé pour être à la fois un extranet et un site Internet.

L'authentification des utilisateurs

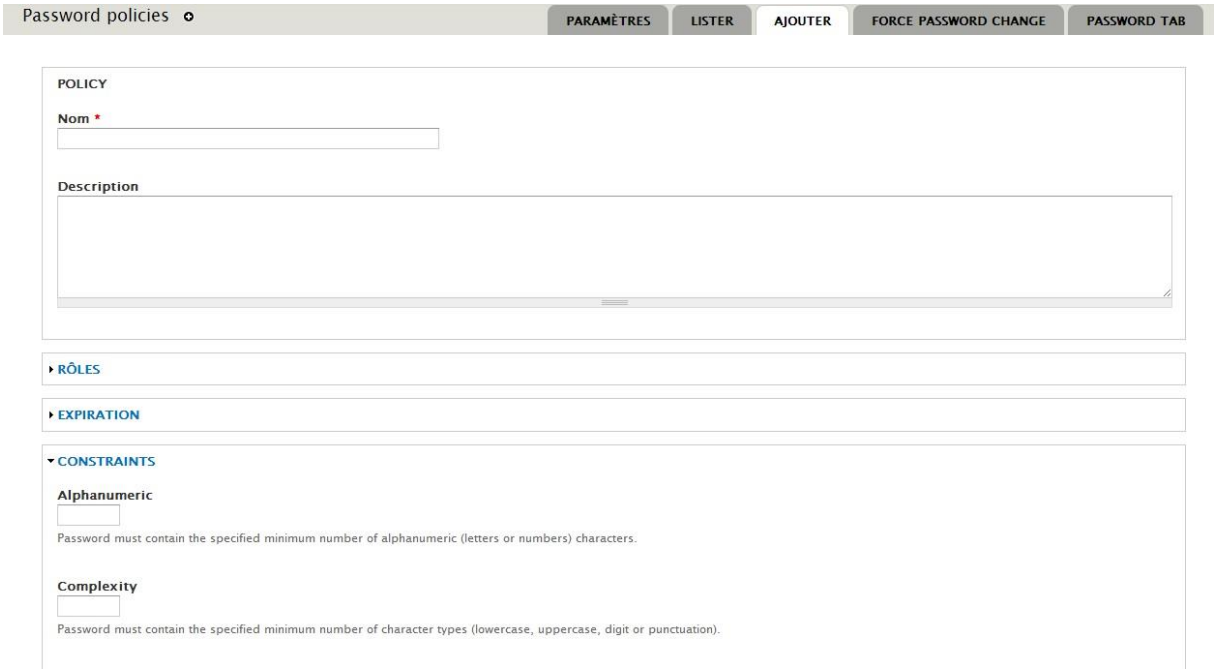
Par défaut, Drupal fournit un formulaire d'inscription et de connexion relativement basique : login, mot de passe et une adresse email.

La plus grande faiblesse des sites actuels sont souvent les mots de passe des utilisateurs. En effet, lors d'une inscription – peu importe le site – un mot de passe temporaire fort, est attribué mais il doit être changé par l'utilisateur. Si le mot de passe temporaire est #18Pç7$=)98zMl@ mais que l'utilisateur le change pour mettre toto123, autant dire que ce n'est pas ce qu'il y a de mieux. Or, à partir du moment où vous mettez en place un site collectant des données personnelles – même à titre ludique et personnel – vous êtes soumis à certaines obligations sinon légales, au moins morales vis-à-vis de vos utilisateurs.

La question est simple : comment être sûr(e) que les utilisateurs ne mettent pas des mots de passe trop évidents sans pour autant passer son temps à fouiller ? On utilise Password Policy qui permet d'inclure certains paramètres dans la création des mots de passe et qui a largement fait ses preuves.

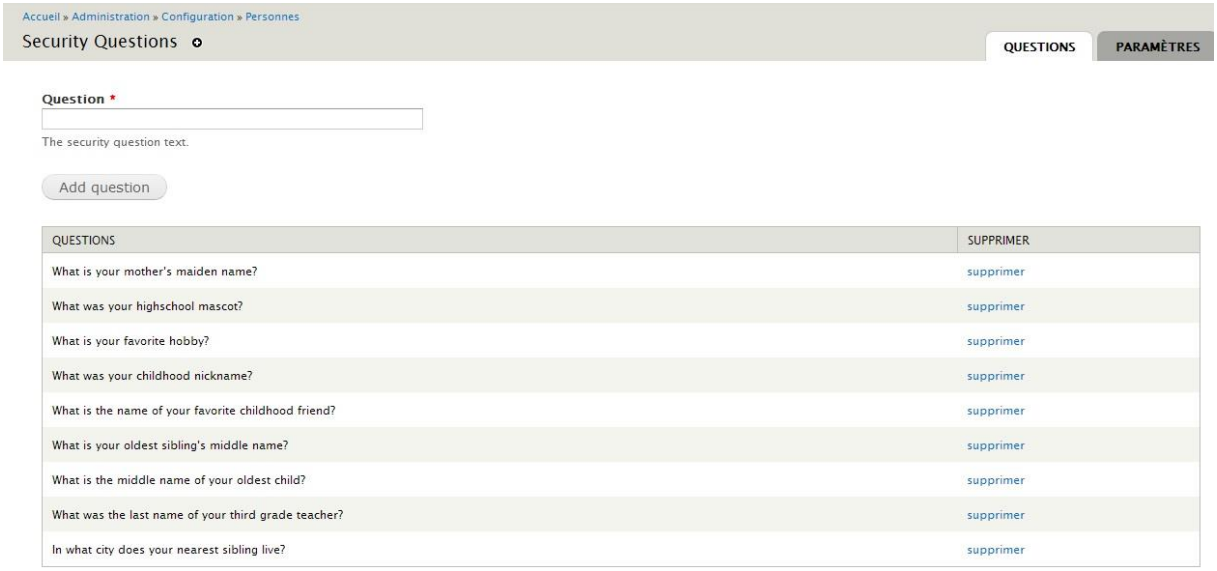
Une fois installé, le module est configurable dans la section Configuration, onglet Personnes.

Dans l'onglet ajouter, vous pouvez créer votre propre méthode de personnalisation de mot de passe en indiquant combien de chiffres, de lettres, de caractères spéciaux et quel niveau de complexité, les utilisateurs doivent prendre en compte pour créer leur mot de passe. Vous pouvez également indiquer un temps d'expiration ainsi que les utilisateurs auxquels cette politique va s'appliquer.

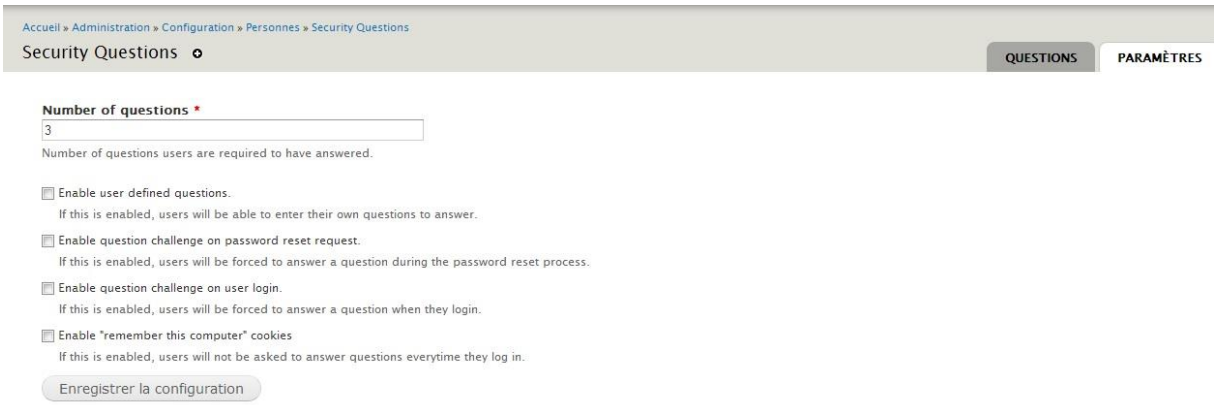


Une fois que l'on s'est assuré de la solidité des mots de passe, il reste un autre point sensible, que tous les administrateurs de forum connaissent très bien : les faux comptes utilisateurs destinés à spammer. Comment s'en débarrasser sans pour autant être obligé de valider – et donc de vérifier – tous les comptes un par un ? On peut recourir aux CAPTCHA mais la solution manque de finesse et les personnes présentant des déficiences ophtalmologiques risquent d'être découragées. On peut alors se tourner vers une solution plus amusante : Secure Questions. Il s'agit d'un module qui vous permet d'ajouter des questions à un formulaire d'inscription et qui ne le valide que lorsque les personnes y ont correctement répondu. On peut également s'en servir pour s'assurer que la personne qui aurait perdu son mot de passe est bien celle qu'elle prétend être.

Tout comme Password Policy, il est paramétrable dans la section Configuration, onglet Personnes.



Par défaut, on voit que certaines questions sont déjà insérées. Vous pouvez les supprimer et en ajouter d'autres, en tenant compte de la spécifié de votre site. Par exemple, si votre site traite d'informatique, vous pouvez demander combien de caractères sont présents dans un hash MD5 ou quelle commande sous Linux permet d'afficher l'intégralité d'un répertoire.



L'onglet paramètres vous permettra de spécifier les actions qui déclencheront les questions aux utilisateurs. Il convient d'émettre trois réserves concernant l'utilisation de ce module.

La première est la question des cookies. A partir du moment où vous avez recours à une solution statistiques telle que Google Analytics, vous avez forcément une collecte de cookies qui s'opère. Selon la nature de votre site et de votre activité et des informations que vous collectez, vous devrez faire une déclaration de fichiers auprès de la CNIL. Dans le cas contraire, vous risquez des sanctions pénales assez lourdes. Tous les sites n'ont pas à être déclarés auprès de la CNIL, vous trouverez sur leur page officielle un simulateur vous permettant de savoir si vous pouvez faire l'objet d'une dispense.

L'autre réserve concerne l'agacement potentiel de vos utilisateurs. Vous pouvez configurer des questions et paramétrer l'outil de telle façon qu'à chaque fois que quelqu'un se connecte, il devra répondre à une question. Mais cela peut rapidement agacer en engendrer une perte de trafic.

Enfin, dans la mesure où l'administrateur ne fournit que les questions et non les réponses qui doivent être données, n'importe quelle réponse est valable. Ainsi, si ce module peut être intéressant pour contrer les robots, il ne sera pas nécessairement d'une grande efficacité contre un véritable attaquant.

Les authentifications OAuth

Les authentifications OAuth sont les possibilités de se connecter à un site en utilisant ses identifiants Google, OpenID, (le défunt) MSN, Twitter ou encore Facebook. Le terme utilisé n'est pas le plus correct.

Par défaut, Drupal permet à des personnes qui ont des identifiants OpenID de les utiliser pour s'identifier. D'autres modules intégrant Google, le défunt MSN ou encore Twitter ont vu le jour afin de faciliter les accès. Pour autant, à titre purement personnel, je n'y suis pas favorable. A mon sens, chaque site sur lequel on se connecte doit avoir son propre jeu de login/mot de passe. Pour des raisons de commodité, on utilise souvent le même couple login/mot de passe partout mais ceci est une erreur. Le problème étant que l'attaquant qui a accès à un compte Facebook – par exemple – peut aussi avoir accès à d'autres sites. Autant fermer la porte à ce type de scénarios et laisser la possibilité aux utilisateurs de prendre leurs responsabilités.

La phase de pré-production

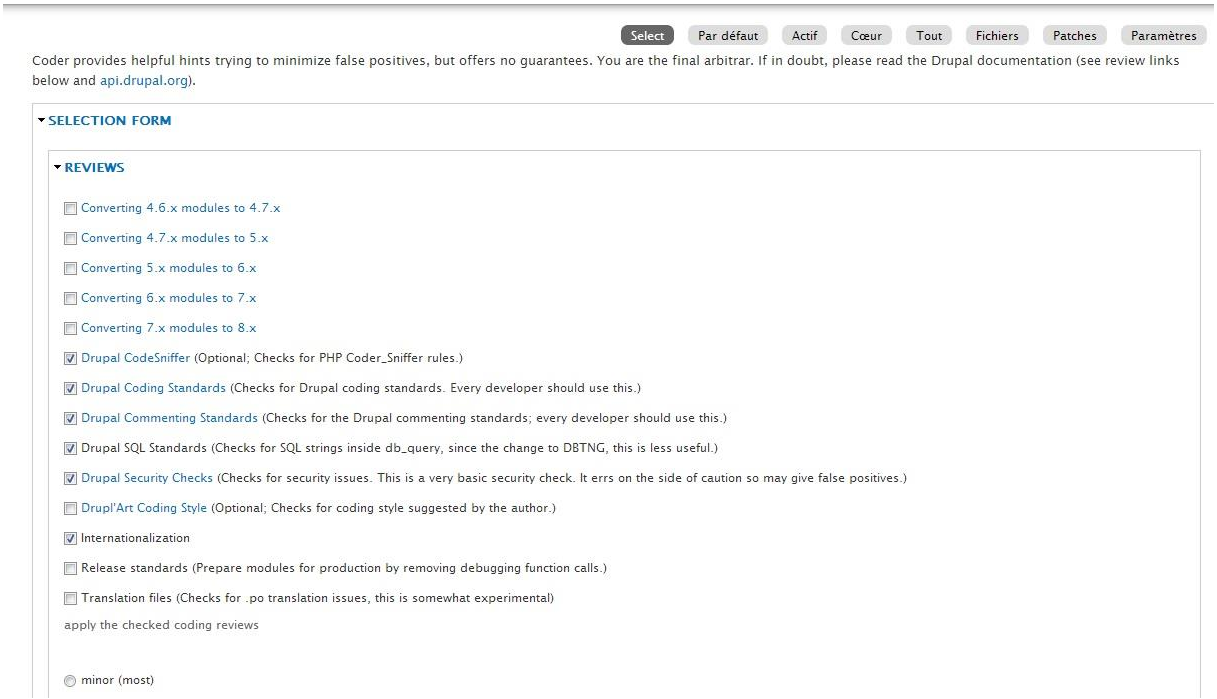
Dans cette partie, nous allons nous intéresser à la phase de pré-production. Idéalement, avant de propulser un site, on se doit de mettre sur pied une pré-prod qui va plus ou moins servir de brouillon et qui va nous permettre de tester, débugguer voire développer tout ce dont on aura besoin pour le futur site.

La vérification du code

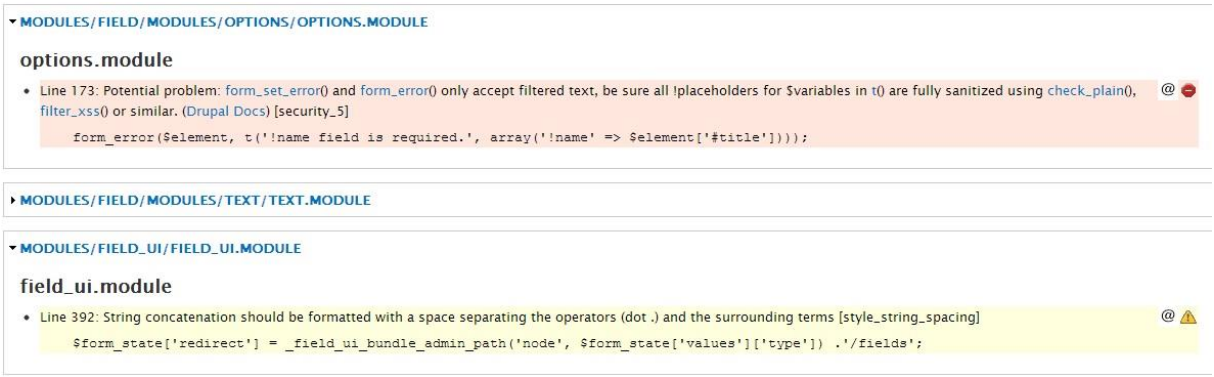
Parfois, on ne voit pas les erreurs qui peuvent exister dans un code. Code Sniffer s'occupe de cela. Il s'agit d'un module qui va scanner l'intégralité des modules installés et vérifier si le code est conforme aux standards. Par exemple, les else, elseif, les { et autres points-virgules.

Coder – anciennement Code Sniffer – va différencier les erreurs critiques, les erreurs «normales » ou mineures mais également ce qui peut être analysé comme des erreurs mais qu'il va ignorer car il les considère comme des faux-positifs. Rassurez-vous, il les listera quand même.

Après l'avoir installé, vous le trouverez dans Configuration, onglet développement.



Vous choisissez les paramètres et vous cliquez sur Run Reviews. Selon le nombre de fonctionnalités installées, le rapport peut mettre plus ou moins de temps à être généré. Rendez-vous ensuite dans l'onglet Tout afin de voir quelles sont les erreurs détectées.



Une fois le rapport généré, il sort une liste, en spécifiant le module et le fichier concerné, la ligne qu'il estime incorrecte, le niveau de sévérité ainsi qu'une explication et un renvoi vers la documentation Drupal qui explique les bonnes pratiques en la matière. En rouge sont listées les erreurs qualifiées de critiques et en jaune, les erreurs mineures.

Si vous avez envie de creuser le langage PHP et comprendre son fonctionnement, je vous recommande son apprentissage par le biais de la Code Academy qui propose exercices et tutoriels sur PHP mais aussi sur Javascript, JQuery, les fondamentaux Web mais aussi le Ruby et le Python.

On pourrait objecter que s'il s'agit de ce type de corrections – somme toute vu comme étant mineures par Coder – on peut procéder aux corrections en production. Ça serait une erreur car parfois, la simple correction d'un point-virgule dans le code peut générer une jolie erreur 500, ce qui n'est pas grave sur une plateforme de pré-production mais qui peut l'être sur une plateforme en production.

Autre point important : dans l'éventualité où vous souhaiteriez l'utiliser sur une plateforme en production, sur laquelle vous avez un système de rapport par email des activités de cette dernière, il conviendra de désactiver tous les systèmes de reporting par email car lors du scan, le watchdog peut s'emballer et littéralement noyer l'administrateur de mails.

Il n'y évidemment pas que PHP qu'il conviendra de revoir mais aussi le code HTML et pour ça, le W3C validator fait très bien le job.

Les tests

Autre module essentiel et qui est intégré dans le core de Drupal : test. Il s'agit d'un module qui va tester le bon fonctionnement des modules mais il doit être cantonné aux plateformes de pré-production et être désactivé lors de la mise en production du site. En effet, s'il peut être très utile dans la phase de test et de débug, il va avoir tendance à générer des erreurs et des ralentissements d'exécution quand il fonctionne en production. Les retours d'expérience de Drupal France sont unanimes sur ce point. Pour ma part, j'ai constaté quelques petits soucis lorsqu'il était activé, soucis qui ont complétement disparu lorsqu'il a été désactivé. A titre personnel, je l'ai carrément supprimé du core et ne le garde que pour les phases de pré-production.

La check-list

Comme son nom l'indique, nous allons établir une liste récapitulant les différents aspects que nous avons vu ensemble.

- Les formats du site sont-ils correctement définis ? □
- Ai-je limité l'accès à PHP ? □
- Ai-je bien configuré les droits d'administration des blocs et des modules ? □
- Ai-je mis en place une tâche Cron automatisée ? □
- Ai-je mis en place un système de rapport par email ? □
- Ai-je mis en place une solution de sauvegarde ? □
- Ai-je correctement configuré les accès aux fichiers ? □
- Les accès d'authentification sont sécurisés ? □
- Ai-je vérifié mon code ? □
- Mon site est-il à jour ? □
- Ai-je bien fait la documentation du site avec la liste complète des fonctionnalités ? □
- Ai-je isolé les modules et thèmes customisés pour les remettre en place en cas de mis à jour du core ? □

Les core Drupal dédiés à la sécurité

Lorsque vous travaillez sur Drupal, vous avez la possibilité de télécharger le core et des modules complémentaires ou d'installer directement un core adapté à vos besoins. Par exemple, le core Drupal Commons permet de mettre en place rapidement un site à vocation communautaire, avec le core de base et des modules complémentaires adaptés.

A ce jour, il existe deux cores Drupal dédiés à la sécurité. Le premier est Drupal Guardr, un core Drupal avec une combinaison de modules dédiés à la sécurité et le second est Drupal Hardened, qui poursuit le mêle objectif.

Si l'intention de départ est louable, ces deux cores ont exactement le même défaut : ils sont toujours en phase de développement. Par ailleurs, comme tous les cores customisés, ils sont relativement lourds et ralentissent considérablement le fonctionnement d'un site. Bien entendu, certains modules préinstallés ne sont pas forcément adaptés à vos besoins et les supprimer vous permettrez d'alléger votre structure.

Si vous souhaitez recourir à l'une de ces deux solutions, veillez à ce que l'ensemble soit à jour et que ce qui peut vous apparaître comme un gain de temps dans l'installation ne se solde pas par une maintenance constante et chronophage.

Annexes

Liste des modules

Les cores dédiés à la sécurité

Drupal Guardr	https://drupal.org/project/guardr
Drupal Hardened	https://drupal.org/project/hardened_drupal

Les modules complémentaires

IP Range Ban	https://drupal.org/project/ip_ranges
Bad Behavior	https://drupal.org/project/badbehavior
Honeypot	https://drupal.org/project/honeypot
Fail2Ban	https://drupal.org/project/fail2ban
Tiny-IDS	https://drupal.org/project/tinyids
Mail system	https://drupal.org/project/mailsystem
HTML Mail	https://drupal.org/project/htmlmail
Backup & Migrate	https://drupal.org/project/backup_migrate
Hacked	https://drupal.org/project/hacked
Security Reviews	https://drupal.org/project/security_review
Secure Pages	https://drupal.org/project/securepages
Secure Login	https://drupal.org/project/securelogin
Password Policy	https://drupal.org/project/password_policy
Code Sniffer/Coder	https://drupal.org/project/coder

Logiciels complémentaires à Drupal

IP NET Info	http://www.nirsoft.net/utils/ipnetinfo.html
Fail2Ban (pour serveur dédié)	http://www.fail2ban.org/wiki/index.php/Main_Page
NAXSI (pour serveur dédié)	https://code.google.com/p/naxsi/downloads/list

Ressources

Code Academy	http://www.codecademy.com/
W3C Validator	http://validator.w3.org/
Drupal Security	https://drupal.org/security
Drupal Watchodg	http://drupalwatchdog.com/
Les failles les plus courantes en 2013	http://blog.sedona.fr/2013/03/top-10-owasp-2013-les-failles-de-securite-web-les-plus-courantes/
OWASP	https://www.owasp.org/index.php/Top_10_2013-T10

Cracking Drupal – A drop in the bucket par Greg James Knaddison chez Wiley Publishing

Sécurité informatique et réseaux par Solange Ghernaouti-Hélie chez Dunod