Le livre de Bugzilla — version 2.18

Version française du livre *The Bugzilla Guide* (2005-01-14)

L'équipe Bugzilla

Adaptation française: Zahir Abdelouhab, Laurent Damour, Mathieu Vinel

Adaptation française de la version 2.16.4: Romain Conseil, Guillaume Huray, Mickaël Lagneaux, Guillaume Tanguy

Relecture: Yvon Benoist, Isabelle Hurbain, Emmanuel Seyman

Relecture de la version 2.16.4: Yvon Benoist, Guillaume Lelarge

Préparation de la publication de la v.f.: Jean-Philippe Guérard

Version : 2.18.fr.1.0

9 novembre 2005

Historique des versions		
Version 2.18.fr.1.0	2005-08-25	ZA, LD, MV, YB, IH, ES, JPG
Mise à jour de l'adaptation française par une équipe d'élèves-ingénieurs en 4e et 5e année (Architecture des Systèmes d'Information) de l'INSA de Rouen.		
Version 2.18	2005-01-14	ÉB
Version 2.16.4.fr.1.0	2004-08-25	RC, GH, ML, GT, YB, GL, JPG
Première adaptation française par une équipe d'élèves-ingénieurs en 4e et 5e année (Architecture des Systèmes d'Information) de l'INSA de Rouen.		
Version 2.16.4	2003-11-01	MPB, ÉB

Résumé

Ce livre est la documentation de Bugzilla, le système de suivi de bogues de mozilla.org. Bugzilla est un logiciel de qualité professionnelle qui permet à des centaines d'organismes dans le monde de suivre des millions d'anomalies.

La version la plus récente de ce livre est toujours disponible sur la page du livre Bugzilla.

Table des matières

Liste des illustrations

Liste des exemples

Chapitre 1. À propos de ce guide

Table des matières

1. Droits d'utilisation *(Copyright Information)*

Si vous avez des questions concernant ce document, ses droits d'utilisation ou sur la publication de ce document sous une forme non électronique, veuillez contacter l'équipe Bugzilla (N.D.T. : en anglais, SVP).

If you have any questions regarding this document, its copyright, or publishing this document in non-electronic form, please contact the Bugzilla Team.

2. Avertissement *(Disclaimer)*

Nous déclinons toute responsabilité quant au contenu de ce document. Vous utilisez les concepts, exemples et autres contenus du présent document à vos risques et périls. Ce document peut contenir des erreurs et des inexactitudes qui pourraient endommager votre système, provoquer une rupture dans votre couple, pousser votre patron à vous licencier, vos chats à faire pipi sur vos meubles et vêtements, voire déclencher une guerre thermonucléaire mondiale. Agissez avec prudence.

No liability for the contents of this document can be accepted. Follow the instructions herein at your own risk. This document may contain errors and inaccuracies that may damage your system, cause your partner to leave you, your boss to fire you, your cats to pee on your furniture and clothing, and global thermonuclear war. Proceed with caution.

Le fait de mentionner des marques ou des produits spécifiques ne constitue pas une recommandation, excepté pour ce qui est du terme « GNU/Linux ». Nous approuvons sans réserve l'utilisation de GNU/Linux. C'est un système d'exploitation extrêmement polyvalent, stable et robuste qui offre un environnement idéal de fonctionnement pour Bugzilla.

Naming of particular products or brands should not be seen as endorsements, with the exception of the term "GNU/Linux". We wholeheartedly endorse the use of GNU/Linux; it is an extremely versatile, stable, and robust operating system that offers an ideal operating environment for Bugzilla.

Bien que l'équipe de développement Bugzilla ait pris grand soin de corriger tous les bogues exploitables, il est bien évident que tout code contient des failles de sécurité. Le plus grand soin devra être pris quant à l'installation et l'utilisation de ce logiciel. Les membres de l'équipe de développement Bugzilla n'assumeront aucune responsabilité quant à l'usage que vous ferez de ce produit. Vous disposez du code source ; vous êtes responsable de le vérifier vous-même afin de vous assurer qu'il répond bien à vos critères en termes de sécurité.

Although the Bugzilla development team has taken great care to ensure that all exploitable bugs have been fixed, security holes surely exist in any piece of code. Great care should be taken both in the installation and usage of this software. The Bugzilla development team members assume no *liability for your use of Bugzilla. You have the source code, and are responsible for auditing it yourself to ensure your security needs are met.*

3. Nouvelles versions

Ceci est la version 2.18 du Guide Bugzilla. Elle est ainsi numérotée pour correspondre à la version de Bugzilla avec laquelle elle est distribuée.

La dernière version de ce guide est toujours disponible à http://www.bugzilla.org, on peut aussi passer par le serveur CVS en suivant les instructions disponibles sur la page CVS de Mozilla et en sortant l'arborescence `mozilla/webtools/bugzilla/docs/`. Cependant, il est conseillé de lire la version correspondante à celle de Bugzilla que vous employez.

Le guide Bugzilla, ou une partie de celui-ci, est aussi disponible dans les langues suivantes : Allemand. Français.

De plus, il y a des projets d'adaptation linguistique de modèles de Bugzilla dans les langues suivantes. La documentation disponible a peut-être été traduite : Biélorusse, Portugais Brésilien, Chinois, Français, Allemand, Coréen, Russe et Espagnol.

Si vous voulez vous porter volontaire pour traduire ce guide dans d'autres langues, veuillez entrer en contact avec Dave Miller.

4. Remerciements

Les personnes énumérées ci-dessous ont apporté d'énormes contributions à la création de ce guide, par leurs écrits, leur total investissement, de nombreuses séances d'aide par courriels ou IRC, et d'une façon générale leur engagement remarquable envers la communauté de Bugzilla :

Matthew P. Barnson <`mbarnson CHEZ sisna POINT com`>

> pour la tâche herculéenne qui a consisté à rassembler le guide Bugzilla et à le produire en version 2.14.

Terry Weissman <`terry CHEZ mozilla POINT org`>

> pour avoir été le premier à écrire Bugzilla et créer le README sur lequel la documentation d'installation UNIX est basée en grande partie.

Tara Hernandez <`tara CHEZ tequilarists POINT org`>

> pour avoir maintenu le développement de Bugzilla à un rythme soutenu après le départ de Terry de mozilla.org et pour s'occuper de landfill.

Dave Lawrence <`dkl CHEZ redhat POINT com`>

> pour avoir fourni un aperçu des principales différences avec le Bugzilla personnalisé de Red Hat.

Dawn Endico <`endico CHEZ mozilla POINT org`>

pour être un extraordinaire mordu d'informatique et avoir supporté les incessantes questions et pinaillages de Matthew sur irc.mozilla.org dans le canal #mozwebtools

Jacob Steenhagen <jake CHEZ bugzilla POINT org>

pour avoir pris la relève pendant la période de développement de la version 2.17.

Dave Miller <justdave CHEZ bugzilla POINT org>

pour la prise en charge de la direction du projet quand Tara s'est retiré et pour les efforts incessants fournis pour que la documentation soit aussi bonne que possible.

Nos remerciements vont également aux personnes suivantes pour leurs contributions significatives à cette documentation : Kevin Brannen, Vlad Dascalu, Ben FrantzDale, Eric Hanson, Zach Lipton, Gervase Markham, Andrew Pearson, Joe Robins, Spencer Smith, Ron Teitelbaum, Shane Travis, Martin Wulffeld.

Il faut également remercier les membres du groupe de nouvelles de netscape.public.mozilla.webtools. Sans vos discussions, perspicacité, suggestions, et correctifs, ceci n'aurait jamais pu exister.

5. Conventions de ce document

Ce document utilise les conventions suivantes :

Descriptions	Représentation
Attention	**Attention** Ne courez pas avec des ciseaux !
Conseil	**Astuce** Voulez-vous un bonbon à la menthe ?
Note	**Note** Cher Jean...
Information demandant une attention spéciale	**Avertissement** Lisez ceci sinon le chat va l'attra-per.
Nom de fichier ou de répertoire	`fichier`
Commande à taper	**commande**
Nom d'application	application
Invite de commande d'un utilisateur normal sous l'interpréteur de commandes bash	bash$
Invite de commande super utilisateur sous l'interpréteur de commandes bash	bash#
Invite de commande d'un utilisateur normal sous l'interpréteur de commandes tcsh	tcsh$
Variables d'environnement	`VARIABLE`
Terme trouvé dans le glossaire	*Bugzilla* [p 125]

Exemple de code	`<para>`
	`Début et fin de paragraphe`
	`</para>`

Cette documentation est maintenue au format DocBook 4.1.2 XML. Il est préférable de soumettre les changements en texte clair ou en XML diff, joints à un bogue référencé dans le composant Bugzilla Documentation.

Chapitre 2. Installer Bugzilla

Table des matières

1. Installation

Note

Si vous voulez juste *utiliser* Bugzilla, vous n'avez pas besoin de l'installer. Ce chapitre ne vous concerne pas. Demandez l'URL à votre administrateur de Bugzilla pour y accéder sur le web.

Le serveur Bugzilla est généralement installé sur Linux ou Solaris. Si vous êtes en train d'installer un autre système d'exploitation, consultez Section 4, « Notes d'installation sur un SE particulier » [p 25] la section 2.4 avant de commencer votre installation pour voir si il n'y a pas d'instructions particu-lières.

Comme alternative pour suivre ces instructions, vous pouvez essayer l'installation non officielle et non supportée de Arne Schimarcher, qui installe Bugzilla et tout le nécessaire sur des systèmes Linux ou Solaris.

Dans ce guide, on considère que vous avez un accès administratif à la machine de Bugzilla. Il n'est pas possible d'installer et d'exécuter Bugzilla tout seul sans accès administratif sauf dans le cas très peu probable où chaque élément obligatoire a déjà été installé.

Avertissement

L'installation peut rendre votre machine vulnérable pendant de courtes périodes de temps. Assurez vous qu'il y a un coupe-feu entre vous et Internet.

Il vous est fortement recommandé de faire une sauvegarde de votre système avant d'installer Bugzilla (et à des intervalles réguliers par la suite :-)

Dans les grandes lignes, l'installation procède comme suit :

1. Installation de Perl [p 11] (5.6.0 ou au dessus pour les plates-formes autres que Windows; 5.8.1 pour Windows)
2. Installation de MySQL [p 11] (3.23.41 ou au dessus)
3. Installation de un serveur Web [p 12]
4. Installation de Bugzilla [p 12]
5. Installation des modules de Perl [p 12]
6. Installation d'un Agent de Transfert de Mail [p 16] (Sendmail 8.7 ou au dessus, ou un ATM qui est compatible avec Sendmail avec au moins cette version)
7. Configuration de tout ce qui précède.

1.1. Perl

Test de la version installée : `perl -v`

Toute machine qui ne possède pas Perl est une machine bien malheureuse. Si vous ne l'avez pas et que votre système d'exploitation ne fournit pas de paquetages officiels, allez sur http://www.perl.com. Bien que Bugzilla fonctionne avec Perl 5.6.0, c'est une bonne idée d'utiliser la dernière version stable. Au moment où ces lignes sont écrites, c'est Perl 5.8.3.

1.2. MySQL

Test de la version installée : `mysql -V`

Si vous ne l'avez pas et que votre système d'exploitation ne fournit pas de paquetages officiels, allez sur http://www.mysql.com. Vous avez besoin de MySQL version 3.23.41 ou supérieure.

Note

De nombreuses versions binaires de MySQL stockent leurs fichiers de données dans le répertoire `/var`. Sur certains systèmes UNIX, ce répertoire se trouve dans la partition principale (root). Si celle-ci est trop petite, il peut ne pas y avoir assez de place pour contenir la base de données des bogues. Pour changer le répertoire de données, vous devez compiler MySQL vous-même à partir des sources, et le mettre comme option à `configure`.

Si vous installez à partir d'autre chose qu'un packaging/installation du système, comme un .rpm (Paquetage Redhat), .deb (Paquetage Debian), .exe (Executable Windows) ou .msi (Installateur Microsoft), assurez vous que le serveur MySQL soit lancé au démarrage de la machine.

1.3. Serveur Web

Test de la version installée : regardez la page de bienvenue par défaut à http://<votre-machine>/

Là, vous avez le choix, à peu près tous les serveurs web capables de faire fonctionner les scripts *CGI* [p 125] conviennent. Cependant, nous recommandons fortement d'utiliser le serveur web Apache (soit 1.3.x ou 2.x), et les instructions d'installation supposent généralement que vous l'employez. Si vous avez Bugzilla qui fonctionne en utilisant un autre serveur web, n'hésitez pas à partager votre expérience avec nous en utilisant la procédure de soumission de bogues dans Bugzilla Documentation.

Si vous n'avez pas Apache et que votre système d'exploitation ne fournit pas de paquetage officiel, allez sur http://httpd.apache.org/.

1.4. Bugzilla

Téléchargez une archive tar Bugzilla (ou jetez un œil sur CVS) et placez la dans un répertoire approprié, accessible par l'utilisateur du serveur web par défaut (probablement « apache » ou « www »). Le mieux est de la mettre soit directement dans l'espace web principal de votre ou peut être dans `/usr/local` avec un lien symbolique provenant de l'espace web.

Attention

La distribution Bugzilla par défaut n'est PAS conçue pour être placée dans un répertoire `cgi-bin`. Ceci est valable pour chaque répertoire configuré à l'aide de l'instruction du `ScriptAlias` d'Apache.

Une fois que tous les fichiers sont dans un répertoire accessible du web, faites en sorte que le répertoire soit accessible en écriture pour votre utilisateur du serveur web. Il s'agit là d une étape provisoire en attendant de lancer le script `checksetup.pl`, ce qui terminera votre installation.

1.5. Modules Perl

Le processus d'installation de Bugzilla s'appuie sur un script nommé `checksetup.pl`. vérifie en premier lieu si vous avez les bonnes versions de tous les modules Perl obligatoires. Cette section a pour but de mener à bien cette vérification. Lorsque que c'est le cas, *ne l'exécutez pas à nouveau*, mais passez à Section 2, « Configuration » [p 16] .

À ce stade, vous devez vous identifier comme administrateur (vous connecter en tant que root). Il vous faudra poursuivre la totalité de l'installation comme administrateur. Ensuite exécutez :

```
bash# ./checksetup.pl
```

`checksetup.pl` va imprimer une liste de modules de Perl obligatoires et optionnels, en même temps que les versions (si il y en a) installés sur votre machine. La liste des modules obligatoires est relativement longue; cependant, vous pouvez déjà en avoir plusieurs d'installés.

Il y a un méta-module nommé Bundle::Bugzilla, qui installe tous les autres modules avec une simple commande. Vous devriez l'utiliser si vous installez Perl 5.6.1 ou une version au dessus.

Le mode préféré pour installer des modules de Perl est via CPAN sur UNIX, ou PPM sur Windows (voir Section 4.1.2, « Perl Modules on Win32 » [p 25]). Ces instructions supposent que vous utilisez CPAN ; si pour une raison ou une autre vous devez installer les modules de Perl manuellement, consultez Annexe D, *Installation manuelle des modules Perl* [p 114] .

```
bash# perl -MCPAN -e 'install "<modulename>"'
```

Si vous utilisez Bundle::Bugzilla, invoquez la commande magique CPAN. Sinon, vous devez parcourir la liste des modules que `checksetup.pl` dit être obligatoires, dans l'ordre donné, en invoquant la commande pour chacun d'eux.

Astuce

Beaucoup d'utilisateurs se plaignent que les modules Perl ne s'installent pas chez eux. La plupart du temps, les messages d'erreur indiquent ne pas trouver un fichier dans « @INC ». À chaque fois ou presque, cette erreur se déclare soit parce que vous n'avez pas les droits suffisants pour compiler les modules Perl, soit parce que les bibliothèques de développement de Perl nécessaires ne sont pas installées sur votre système. Demandez à votre administrateur UNIX de vous accorder les droits d'accès suffisants. Si vous *êtes* l'administrateur UNIX, veuillez consulter le forum ou la liste de diffusion pour une aide plus poussée, ou faites appel à quelqu'un pour vous aider.

Voici une liste complète de modules et de leur version minimum. Quelques modules ont des notes d'installation spéciales; celles-ci sont indiquées juste après.

Modules Perl obligatoires :

1. AppConfig (1.52)
2. CGI (2.93)
3. Data::Dumper (n'importe)
4. Date::Format (2.21)
5. DBI (1.36)
6. DBD::mysql [p 14] (2.1010)

7. File::Spec (0.82)
8. File::Temp (n'importe)
9. Template [p 14] (2.08)
10. Text::Wrap (2001.0131)

Modules Perl facultatifs :

1. GD [p 14] (1.20) pour les graphiques de bogues
2. Chart::Base [p 15] (1.0) pour les graphiques de bogues
3. GD::Graph [p 15] (n'importe) pour les graphiques de bogues
4. GD::Text::Align [p 15] (n'importe) pour les graphiques de bogues
5. XML::Parser [p 15] (n'importe) pour l'interface XML
6. PatchReader [p 15] (0.9.4) pour une jolie vue en HTML des correctifs
7. MIME::Parser [p 15] (n'importe) pour l'interface facultative par courrier électronique

1.5.1. DBD::mysql

Au cours du processus d'installation, il vous sera posé quelques questions sur la cible que vous souhaitez pour la compilation et votre installation MySQL. Pour la plupart des questions, la réponse par défaut sera suffisante, mais quand le processus vous demandera si la cible souhaitée est le paquet MySQL ou mSQL, vous devrez choisir celle liée à MySQL. Plus tard, le programme vous demandera si vous voulez conserver une compatibilité inverse avec les paquets MySQL plus anciens. Vous devez répondre OUI alors que la réponse par défaut est NON.

Une machine hôte « localhost » sera suffisante. Un utilisateur de tests « test », avec un mot de passe vide, doit avoir les droits d'accès suffisants pour faire des essais sur la base de données de tests « test » que MySQL génère à l'installation.

1.5.2. Template Toolkit (2.08)

Lors de l'installation de Template Toolkit, une série de questions vous sera posée à propos des fonctionnalités à activer. Les options par défaut conviennent bien, à part qu'il est recommandé d'utiliser le très rapide « XS Stash » du Template Toolkit, pour réaliser de meilleures performances.

1.5.3. GD (1.20)

Le module GD n'est nécessaire que si vous voulez des rapports graphiques.

Note

Le module Perl GD nécessite d'autres bibliothèques qui peuvent ou non être installées sur votre système, telles que `libpng` et `libgd`. La liste complète des bibliothèques nécessaires se trouve dans le fichier README du module Perl GD. Si la compilation de GD échoue, c'est probablement parce qu'il manque une des bibliothèques demandées.

Astuce

La version du module GD dont vous avez besoin est très étroitement liée à la version de `libgd` installée sur votre système. Si vous avez une version 1.x de `libgd`, les versions 2.x du module GD ne fonctionneront pas chez vous.

1.5.4. Chart::Base (1.0)

Le module Chart::Base n'est nécessaire que si vous voulez des rapports graphiques. Notez que les versions antérieures à 0.99c utilisaient les GIF, mais ils ne sont plus supportés par les dernières versions de GD.

1.5.5. GD::Graph (n'importe)

Le module GD::Graph n'est nécessaire que si vous voulez des rapports graphiques.

1.5.6. GD::Text::Align (n'importe)

Le module GD::Text::Align n'est nécessaire que si vous voulez des rapports graphiques.

1.5.7. XML::Parser (n'importe)

Le module XML::Parser n'est nécessaire que si vous souhaitez importer les bogues XML en utilisant le script `importxml.pl`. Ceci est nécessaire pour utiliser la fonctionnalité « changement d'état d'un bogue » [« move bugs »] de Bugzilla ; vous aurez peut-être également besoin de l'utiliser pour un déplacement en provenance d'une autre base de données de bogues. XML::Parser a besoin que la bibliothèque `expat` soit déjà installée sur votre machine.

1.5.8. MIME::Parser (n'importe)

Le module MIME::Parser est nécessaire seulement si vous voulez utiliser l'interface de messagerie électronique située dans le répertoire `contrib`.

1.5.9. PatchReader (0.9.4)

Le module PatchReader est nécessaire seulement si vous voulez utiliser le « Patch Viewer », fonctionnalité de Bugzilla qui permet de montrer des correctifs de code dans votre navigateur WEB sous une forme plus lisible.

1.6. Mail Transfer Agent (MTA)

Bugzilla s'appuie sur la possibilité d'accès à un système de messagerie électronique pour son authentification utilisateur et pour d'autres tâches.

Sous Linux, tout MTA compatible avec Sendmail suffira. Sendmail, Postfix, qmail et Exim font partie des MTA courants. Sendmail est le MTA d'origine d'UNIX, mais les autres sont plus faciles à configurer, ce qui fait que beaucoup de personnes remplacent Sendmail par Postfix ou Exim. Le remplacement se fait sous forme d'échange standard, si bien que Bugzilla ne fera pas la différence entre eux.

Si vous utilisez Sendmail, la version 8.7 ou supérieure est nécessaire. Si vous utilisez un MTA compatible avec Sendmail, il doit être conforme au minimum à la version 8.7 de Sendmail.

Consultez les instructions d'installation détaillées dans le manuel correspondant au MTA que vous choisissez. Chacun de ces programmes aura son propre fichier de configuration où vous devrez configurer certains paramètres pour vous assurez que le message électronique est transmis correctement. Ils sont exécutés comme utilitaires, et vous devez vous assurer que le MTA est dans la liste de démarrage automatique des utilitaires de la machine.

Si un simple message électronique envoyé avec la ligne de commande **mail** fonctionne, alors Bugzilla devrait également être opérationnel.

2. Configuration

Avertissement

Les installations de MySQL et Bugzilla dont la configuration était médiocre ont permis à des pirates informatiques de s'introduire dans des systèmes par le passé. Veuillez prendre au sérieux les parties de ces instructions qui portent sur la sécurité, même pour les machines Bugzilla bien cachées derrière un coupe-feu. N'oubliez surtout pas de lire les conseils de sécurité importants donnés dans le Chapitre 4, *Sécurité de Bugzilla* [p 52] .

2.1. localconfig

Dès que vous exécutez `checksetup.pl` avec tous les bons modules installés, il affiche un message concernant un fichier nommé `localconfig` qu'il crée. Ce fichier contient les réglages par défaut pour un certain nombre de paramètres de Bugzilla.

Chargez ce fichier dans votre éditeur. La seule valeur que vous *devez* changer est $db_pass, mot de passe utilisateur que vous allez créer pour votre base de données. Tapez à cet endroit le mot de passe adéquate (pour simplifier, il ne devrait pas contenir le caractère « guillemet unique ») que vous avez choisi.

Les commentaires fournis dans le fichier `localconfig` donnent des informations sur les autres options. Si vous avez une installation de MySQL légèrement non standard, vous aurez la possibilité de changer un ou plusieurs des autres paramètres « $db_* ».

Si vous le souhaitez, vous pouvez changer le nom des priorités, les niveaux de gravité, les systèmes d'exploitation et les plates-formes pour votre installation. Cependant, vous pouvez toujours changer tout ça après que l'installation soit finie ; si vous re-exécutez `checksetup.pl`, les changements seront mis à jour.

2.2. MySQL

Attention

La configuration par défaut de MySQL est très vulnérable. Section 2, « MySQL » [p 53] donne des informations utiles pour améliorer la sécurité de votre installation.

2.2.1. Autoriser les fichiers joints volumineux

Par défaut, MySQL acceptera seulement les paquets ayants une taille maximum de 64 ko. Si vous voulez avoir des fichiers joints plus gros que ça, vous devez modifier votre `/etc/my.cnf` comme indiqué ci-dessous.

Si vous utilisez MySQL 4.0 ou plus récente, entrez :

```
[mysqld]
# Allow packets up to 1M
max_allowed_packet=1M
```

Si vous utilisez une version plus vieille de MySQL, entrez :

```
[mysqld]
# Allow packets up to 1M
set-variable = max_allowed_packet=1M
```

Il y a aussi un paramètre dans Bugzilla appelé 'maxattachmentsize' (par défaut = 1000 ko) qui contrôle la taille maximum autorisable des pièces jointes. Les pièces jointes plus volumineuses que l'une des deux valeurs 'max_allowed_packet' *ou* 'maxattachmentsize' ne seront pas acceptées par Bugzilla.

2.2.2. Autoriser les mots courts dans les index en texte intégral

Par défaut, les mots doivent avoir une longueur d'au moins quatre caractères pour être indexés dans les index en texte intégral de Bugzilla. En conséquence beaucoup de mots spécifiques à Bugzilla ne sont pas pris en compte, parmi lesquels « cc », « ftp » et « uri ».

MySQL peut être configuré pour indexer ces mots en réglant le paramètre ft_min_word_len à la taille minimum des mots à indexer. Ceci peut être fait en modifiant le `/etc/my.cnf` selon l'exemple ci-dessous :

```
[mysqld]
# Allow small words in full-text indexes
ft_min_word_len=2
```

La reconstruction des index peut être faite en s'appuyant sur la documentation trouvée à http://www.mysql.com/doc/en/Fulltext_Fine-tuning.html. (N.D.T. : la version française se trouve à http://www.mysql.com/doc/fr/Fulltext_Fine-tuning.html).

Note

Le paramètre Ft_min_word_len est supporté seulement dans MySQL v4 ou supérieure.

2.2.3. Permettre à la table des fichiers joints de passer à une taille supérieure à 4 Go

Par défaut, MySQL limitera la taille de la table à 4 Go. Cette limite existe même si le système de fichiers sous-jacent n’a pas une telle limite. Pour fixer une limite plus haute, suivez les instructions suivantes.

Exécutez le client en ligne de commande `MySQL` et entrez :

```
mysql> ALTER TABLE attachments AVG_ROW_LENGTH=1000000, MAX_ROWS=20000;
```

La commande ci-dessus fera passer à 20 Go. Mysql devra faire une copie temporaire de l’intégralité de votre table pour cela. Idéalement, vous devriez faire ça quand la table est encore petite.

2.2.4. Ajouter un utilisateur à MySQL

Vous devez ajouter un nouvel utilisateur que Bugzilla puisse utiliser. (Il n’est pas prudent que Bugzilla utilise le compte root de MySQL.) Les instructions suivantes supposent que les paramètres par défaut sont dans `localconfig`; si vous changez ces derniers, vous devez changer la commande MySQL de façon appropriée. Vous aurez besoin du mot de passe *`$db_pass`* vous avez mis dans `localconfig` dans Section 2.1, « localconfig » [p 16] .

Nous utilisons la commande SQL **GRANT** pour créer un utilisateur « bugs ». Cela restreint également les opérations de l’utilisateur « bugs » à celles agissant sur la base de données « bugs » et n’autorise le compte à se connecter que depuis « localhost ». Modifiez cela en fonction de votre configuration si vous devez vous connecter plus tard depuis une autre machine ou sous un autre utilisateur.

Exécutez le client en ligne de commande `mysql`.

Si vous utilisez MySQL 4.0 ou plus récent, entrez :

```
mysql> GRANT SELECT, INSERT,
       UPDATE, DELETE, INDEX, ALTER, CREATE, LOCK TABLES,
       CREATE TEMPORARY TABLES, DROP, REFERENCES ON bugs.*
       TO bugs@localhost IDENTIFIED BY '$db_pass';
mysql> FLUSH PRIVILEGES;
```

Si vous utilisez une version plus ancienne de MySQL, les permissions de `LOCK TABLES` et `CREATE TEMPORARY TABLES` seront indisponibles et doivent être supprimées de la liste des permissions. Dans ce cas, on peut utiliser la ligne de commande suivante :

```
mysql> GRANT SELECT, INSERT,
       UPDATE, DELETE, INDEX, ALTER, CREATE, DROP,
       REFERENCES ON bugs.* TO bugs@localhost IDENTIFIED BY
       '$db_pass';
mysql> FLUSH PRIVILEGES;
```

2.3. checksetup.pl

Ensuite, réexécutez `checksetup.pl`. Il reconfirme que tous les modules sont présents et fait remarquer que le fichier localconfig a changé, qu’il considère comme ayant été édité à votre satisfaction. Il compile les modèles UI [User Interface templates], se connecte à la base de données en utilisant l’utilisateur « bugs » que vous avez créé et le mot de passe que vous avez défini et il crée la base de données de « bugs » et les tables incluses.

Après cela, il demande des informations sur un compte administrateur. Bugzilla peut avoir plusieurs administrateurs — vous pouvez encore en créer d'autres plus tard — mais il en a besoin d'un pour commencer. Entrez l'adresse de courrier électronique d'un administrateur, son nom complet et un mot de passe Bugzilla approprié.

`checksetup.pl` a maintenant fini. Vous pouvez réexécuter `checksetup.pl` quand vous le souhaitez.

2.4. Web server

Configurez votre serveur web selon les instructions données dans la section appropriée. (Si cela peut influencer votre choix, l'équipe de Bugzilla recommande Apache.) Quel que soit le serveur web que vous utilisez, cependant, faites en sorte que les informations sensibles ne soient pas disponibles à distance en appliquant correctement les contrôles d'accès indiqués dans Section 3.1, « Désactivation des accès à distance pour les fichiers de configuration Bugzilla » [p 54] .

2.4.1. httpd™ d'Apache

Chargez `httpd.conf` dans votre editeur.

Décommentez (ou ajouter) la ligne suivante. Ceci autorise Apache à exécuter les fichiers .cgi files en dehors du répertoire `cgi-bin`.

```
  AddHandler cgi-script .cgi
```

Apache utilise les directives `<Directory>` pour peaufiner les droits. Ajoutez les deux lignes suivantes à une directive `<Directory>` qui s'applique soît au répertoire Bugzilla ou l'un de ses parents (i.e. la directive `<Directory /var/www/html>`). Ceci autorisent les fichiers `.htaccess` de Bugzilla à outrepasser les permissions globales. et permet aux fichiers .cgi de tourner dans le répertoire Bugzilla.

```
Options +ExecCGI +FollowSymLinks
AllowOverride Limit
```

Ajoutez `index.cgi` à la fin de la ligne `DirectoryIndex`.

`checksetup.pl` peut mettre des permissions plus restreintes sur les fichiers et les répertoires de Bugzilla si il sait en tant que quel groupe le serveur web est exécuté. Trouvez la ligne `Group` dans `httpd.conf` et placez la valeur que vous y trouvez dans la variable *`$webservergroup`* dans `localconfig`. Puis réexécutez `checksetup.pl`.

2.4.2. Internet Information Services™ de Microsoft

Si vous êtes en train d'exécuter Bugzilla sous Windows et que vous choisissez d'utiliser Internet Information Services™ ou Personal Web Server™ de Microsoft, vous devrez exécuter un certain nombre d'autres étapes de configuration comme expliqué ci-dessous. Vous aurez peut-être également besoin de consulter les articles suivants de la base de connaissance de Microsoft : 245225 « HOW TO: Configure and Test a PERL Script with IIS 4.0, 5.0, and 5.1 » (pour Internet Information Services™) (N.D.T. : « HOW TO : configurez et testez un script PERL avec IIS 4.0, 5.0 et 5.1 ») et 231998 « HOW TO: FP2000: How to Use Perl with Microsoft Personal Web Server on Windows 95/98 » (pour Personal Web Server™) (N.D.T. : « HOW TO : FP2000 : Perl utilisation avec serveur Web personnel Microsoft sous Windows 95/98 »).

Vous aurez besoin de créer un répertoire virtuel pour l'installation de Bugzilla. Mettez les fichiers de Bugzilla dans un répertoire dont le nom est *différent* du répertoire que vous voulez rendre accessible à l'utilisateur final. C'est-à-dire, si vous voulez que vos utilisateurs accèdent à l'installation de Bugzilla par « http://<lenomdevotredomaine>/Bugzilla », ne mettez *pas* les fichiers de Bugzilla dans un répertoire appelé « Bugzilla ». Mettez les plutôt dans un endroit différent, puis utilisez l'outil d'administration de IIS pour créer un répertoire virtuel appelé « Bugzilla » qui joue le rôle d'alias pour l'emplacement réel des fichiers. En créant ce fichier virtuel, assurez vous que vous avez ajouté les droits d'accès « Execute (comme les applications ISAPI ou CGI) ».

Vous devrez aussi dire à IIS comment s'y prendre avec les fichiers .cgi de Bugzilla. Utilisez de nouveau l'outil d'administration de IIS, ouvrez les propriétés pour le nouveau répertoire virtuel et choisissez l'option de configuration pour accéder aux scripts d'applications [Script Mapping]. Créez une entrée d'application .cgi à l'adresse :

```
<chemin complet vers perl.exe >\perl.exe -x<chemin complet vers Bugzilla> -wT "%s" %s
```

Par exemple :

```
c:\perl\bin\perl.exe -xc:\bugzilla -wT "%s" %s
```

Note

L'installation d'ActiveState peut avoir déjà créé une entrée pour les fichiers .pl qui est limitée à « GET,HEAD,POST ». Si c'est le cas, cette application doit être *supprimée* car les fichiers .pl de Bugzilla ne sont pas conçus pour être exécutés via un serveur web.

IIS devra également savoir que l'index.cgi doit être traité comme un document par défaut. Sur la page onglets de Documents des propriétés du répertoire virtuel, vous devez ajouter index.cgi comme type de document par défaut. Si vous le voulez, vous pouvez effacer les autres types de document par défaut de ce répertoire virtuel particulier, puisque Bugzilla n'utilise aucun d'eux.

De plus, et on ne le soulignera jamais assez, assurez-vous que les fichiers tels que `localconfig` et votre répertoire `data` sont sécurisés comme décrit dans Section 3.1, « Désactivation des accès à distance pour les fichiers de configuration Bugzilla » [p 54] .

2.4.3. Serveur AOL

Ben FrantzDale que l'utilisation du serveur AOL avec Bugzilla a donné d'excellents résultats. Il a fait part de son expérience et celle-ci nous permet de proposer ce qui suit.

Le serveur AOL devra être configuré pour pouvoir exécuter les scripts *CGI* [p 125] , veuillez consulter la documentation qui va avec votre serveur pour plus d'information sur la façon de procéder.

Parce que les serveurs AOL ne supportent pas les fichiers `.htaccess`, vous devrez créer un script *TCL* [p 128] . Vous devrez créer un `aolserver/modules/tcl/filter.tcl` (le nom du fichier sera sans importance) avec le contenu suivant (remplacez `/bugzilla/` par le chemin web de votre instance Bugzilla) :

```
ns_register_filter preauth GET /bugzilla/localconfig filter_deny
ns_register_filter preauth GET /bugzilla/localconfig~ filter_deny
ns_register_filter preauth GET /bugzilla/\#localconfig\# filter_deny
ns_register_filter preauth GET /bugzilla/*.pl filter_deny
ns_register_filter preauth GET /bugzilla/syncshadowdb filter_deny
```

```
ns_register_filter preauth GET /bugzilla/runtests.sh filter_deny
ns_register_filter preauth GET /bugzilla/data/* filter_deny
ns_register_filter preauth GET /bugzilla/template/* filter_deny

proc filter_deny { why } {
    ns_log Notice "filter_deny"
    return "filter_return"
}
```

Avertissement

Ceci ne fonctionne probablement pas pour tous les fichiers de sauvegarde d'éditeur possibles alors vous aurez peut-être envie d'ajouter quelques variations supplémentaires de `localconfig`. Pour plus d'informations, voyez le bogue 186383 ou Bugtraq ID 6501.

Note

Si vous utilisez le logiciel webdot depuis research.att.com (la configuration par défaut pour le paramètre `webdotbase`), vous devrez autoriser l'accès à `data/webdot/*.dot` pour la machine reasearch.att.com.

Si vous utilisez une installation locale de GraphViz, vous devrez autoriser tout le monde à accéder aux `*.png`, `*.gif`, `*.jpg` et `*.map` dans le répertoire `data/webdot`.

2.5. Bugzilla

Votre Bugzilla devrait maintenant fonctionner. Allez sur `http://<your-bugzilla-server>/` — vous devriez voir la page d'accueil de Bugzilla. Si ce n'est pas le cas, consultez la section Dépannage, Annexe B, *Résolution des problèmes* [p 109] .

Identifiez vous avec le compte administrateur que vous avez défini lors de la dernière exécution de `checksetup.pl`. Vous devriez parcourir les options sur la page « Edit Parameters » (le lien est en bas de page) et voir s'il n'y en pas certaines que vous aimeriez changer. Les options importantes sont documentées dans Section 1, « Configuration de Bugzilla » [p 33] ; vous devrez certainement modifier **maintainer** et **urlbase**; vous pouvez aussi modifier **cookiepath** ou **requirelogin**.

Cela pourrait être l'occasion de refaire un tour sur le fichier `localconfig` et de vous assurer que les noms des priorités, degrés de gravité [severities], plate-formes et systèmes d'exploitation sont ceux que vous souhaitez utiliser quand vous commencez à créer un bug. N'oubliez pas de réexécuter `checksetup.pl` si vous changez ce dernier.

Bugzilla a plusieurs options facultatives qui nécessitent des configurations supplémentaires. Vous pouvez vous documenter à ce sujet dans Section 3, « Configuration supplémentaire facultative » [p 21] .

3. Configuration supplémentaire facultative

Bugzilla a un nombre important d'options. Cette section décrit comment les configurer ou les activer.

3.1. Les graphiques de bogues

Si vous avez installé les modules Perl nécessaires, vous pouvez commencer à collecter des statistiques pour les superbes graphes Bugzilla.

```
bash# crontab -e
```

Ceci devrait afficher le fichier crontab dans votre éditeur. Ajoutez une entrée « cron » comme ceci pour exécuter `collectstats.pl` tous les jours à minuit 5 :

```
5 0 * * * cd <your-bugzilla-directory> ; ./collectstats.pl
```

Deux jours après, vous pourrez visualiser les graphiques de bogues depuis la page de rapport de bogue.

> **Note**
>
> Windows n'a pas de 'cron', mais il a un planificateur de tâches, qui fournit les mêmes fonctions. Il y a également des outils conçus par des tiers qui peuvent être utilisés pour exécuter cron, tels que nncron.

3.2. Diagrammes de dépendance

Tout comme les arborescences de dépendance en mode texte, Bugzilla permet également des aperçus graphiques, en utilisant un logiciel appelé « dot ». C'est le paramètre webdotbase qui en assure précisément le fonctionnement ; il peut prendre l'une de ces trois valeurs :

1. Le chemin d'accès complet à la commande 'dot' (qui fait partie de GraphViz) qui va générer les graphiques localement.
2. Un préfixe URL pointant vers une installation du logiciel webdot qui va générer les graphiques à distance.
3. Une valeur nulle qui désactive les graphiques de dépendance.

La façon la plus simple de le faire fonctionner est d'installer GraphViz. Dans ce cas, vous devez activer les cartes d'image côté serveur dans Apache. Autrement, vous pouvez installer un serveur webdot ou utiliser le serveur public webdot AT&T. C'est le choix par défaut pour l'option webdotbase mais il est souvent surchargé et lent. Notez que le serveur de AT&T ne fonctionnera pas si Bugzilla n'est accessible que par HARTS. *Note du rédacteur : Mais bon sang, qu'est ce que c'est que ce HARTS ? Google ne sait pas...*

3.3. Le planificateur de pleurnicherie

Les meilleurs bogues ne sont-ils pas aussi les plus énervants ? Pour vous aider à rendre ces bogues encore plus agaçants, vous pouvez activer le système automatique de plainte de Bugzilla pour vous plaindre des ingénieurs qui laissent leur bogues dans un état NEW ou REOPENED sans faire de triage.

Pour ce faire, ajoutez la commande suivante en tant qu'entrée crontab quotidienne, de la même manière que cela a été fait plus haut pour les diagrammes de bogue. Dans cet exemple, le planificateur s'active à minuit 55 :

```
55 0 * * * cd <your-bugzilla-directory> ; ./whineatnews.pl
```

Note

Windows n'a pas de 'cron', mais il a un planificateur de tâches, qui fournit les mêmes fonctions. Il y a également des outils conçus par des tiers qui peuvent être utilisés pour exécuter cron, tel que nncron.

3.4. Patch Viewer

Patch Viewer est le moteur qui permet l'affichage graphique des correctifs. Vous pouvez l'intégrer avec des copies des outils `cvs`, `lxr` et `bonsai` si vous les avez, en donnant l'emplacement de votre installation de ces outils dans `editparams.cgi`.

Patch Viewer pourra aussi éventuellement utiliser les utilitaires de ligne de commande `cvs`, `diff` et `interdiff` s'ils existent sur le système. On peut obtenir Interdiff sur http://cyberelk.net/tim/patchutils/. Si ces programmes ne sont pas dans le chemin du système, vous pouvez configurer leurs emplacements dans `localconfig`.

3.5. Authentification LDAP

L'authentification LDAP est un module pour l'architecture d'authentification de plugin de Bugzilla.

Le schéma d'authentification traditionnel de Bugzilla utilise les adresses de courrier électronique comme identifiant utilisateur primaire et un mot de passe pour identifier l'utilisateur. Tous les endroits de Bugzilla qui traitent les utilisateurs (par exemple pour assigner un bogue) utilisent l'adresse de courrier électronique. L'authentification LDAP se place au dessus de ce schéma au lieu de le remplacer. L'utilisateur se connecte au début avec un nom d'utilisateur et un mot de passe pour l'annuaire LDAP. L'adresse de courrier électronique est récupérée depuis LDAP et l'utilisateur est identifié par le système traditionnel d'une façon cohérente en utilisant son adresse de courrier électronique. Si un compte pour cette adresse de courrier électronique existe déjà dans votre système Bugzilla, il s'y connectera. Si aucun compte pour cette adresse de courrier électronique n'existe, un compte est créé au cours de la connexion. (Dans ce cas, Bugzilla essaie d'utiliser les attributs « displayName » ou « cn » pour déterminer le nom complet de l'utilisateur). Après l'authentification, toutes les tâches liées à cet utilisateur sont gérées par l'adresse de courrier électronique et non par le nom d'utilisateur LDAP. Vous continuez d'assigner des bogues par adresse de courrier électronique, de rechercher des utilisateurs par adresse de courrier électronique, etc.

Attention

Du fait que le compte Bugzilla ne se crée qu'à la première connexion d'un utilisateur, un utilisateur qui ne s'est pas encore connecté est inconnu de Bugzilla. Ceci signifie qu'ils ne peuvent pas être utilisés comme propriétaire d'un bogue ou contact QA (que ce soit par défaut ou autrement), ajoutés à une liste cc, ou toute autre opération de ce genre. Une solution de rechange possible est le script `bugzilla_ldapsync.rb` dans le répertoire *`contrib`* [p 125] . Une autre solution possible est de corriger le bogue 201069.

Paramètres requis pour utiliser l'authentification LDAP :

loginmethod

Ce paramètre doit être fixé à « LDAP » *seulement* si vous allez utiliser un répertoire LDAP pour l'authentification. Si vous mettez ce paramètre à « LDAP » mais que vous ne réglez pas les autres paramètres de la liste ci-dessous, vous ne pourrez pas vous reconnecter à Bugzilla une fois que vous serez déconnecté. Si cela vous arrive, vous devrez éditer manuellement `data/params` et mettre loginmethod à « DB ».

LDAPserver

Pour ce paramètre, vous devez indiquer le nom (et le port si vous le souhaitez) de votre serveur LDAP. Si le port n'est pas spécifié, il considère que c'est le port LPDA par défaut 389.

Ex. « ldap.company.com » ou « ldap.company.com:3268 »

LDAPbinddn [Optional]

Certains serveurs LDAP n'autoriseront pas une connexion anonyme à faire une recherche dans le répertoire. Si c'est le cas pour votre configuration, vous devrez régler le paramètre LDAPbinddn au compte de l'utilisateur que Bugzilla devra utiliser à la place de la connexion anonyme.

Ex. « cn=default,cn=user:password »

LDAPBaseDN

Pour le paramètre LDAPBaseDN vous devrez indiquer l'emplacement dans votre arborescence LDAP où vous aimeriez que se fasse la recherche des adresses électroniques. Vos identifiants UID devraient être uniques sous la base DN indiqués ici.

Ex. « ou=People,o=Company »

LDAPuidattribute

Pour le paramètre LDAPuidattribute vous devrez indiquer l'attribut qui contient l'UID unique de vos utilisateurs. La valeur récupérée de cet attribut sera utilisée quand ils essayeront de se connecter en tant qu'utilisateur pour confirmer leur mot de passe.

Ex. « uid »

LDAPmailattribute

Le paramètre LDAPmailattribute doit être le nom de l'attribut qui contient l'adresse électronique que vos utilisateurs entreront dans les cases d'identification de Bugzilla.

Ex. « mail »

3.6. Traitement des formats différents avec le type de MIME adéquat

Certaines pages de Bugzilla ont des formats différents, autres que le HTML ordinaire. En particulier, quelques pages peuvent produire leur contenu soit en XUL (un format spécial de Bugzilla, qui ressemble à un programme GUI) soit en RDF (un genre d'XML structuré qui peut être lu par de nombreux programmes).

Pour que vos utilisateurs voient ces pages correctement, Apache doit les envoyer avec le type MIME adéquat. Pour ce faire, ajoutez les lignes suivantes à la configuration d'Apache, soit dans la section `<VirtualHost>` pour votre Bugzilla, soit dans la section `<Directory>` pour votre Bugzilla :

```
AddType application/vnd.mozilla.xul+xml .xul
AddType text/xml .rdf
```

4. Notes d'installation sur un SE particulier

Le système d'exploitation sur lequel vous avez choisi d'installer Bugzilla peut avoir des effets sur de nombreux aspects de son installation. Selon le système, elle peut être plus facile ou être plus difficile. Cette section a pour but de vous aider à comprendre les difficultés rencontrées en l'exécutant sur un système d'exploitation spécifique ainsi que les utilitaires disponibles pour faciliter les choses.

Si vous avez quelque chose à ajouter ou des notes pour un système d'exploitation non traité ici, signalez le dans Bugzilla Documentation.

4.1. Microsoft Windows

Faire fonctionner Bugzilla sous Windows est plus difficile que de le faire fonctionner sous Unix. Pour cette raison, nous recommandons toujours de le faire sur un système basé sur Unix tel que GNU/Linux. Ceci dit, si vous voulez que Bugzilla fonctionne sous Windows, vous devrez faire les réglages suivants.

4.1.1. Win32 Perl

On peut obtenir Perl pour Windows sur ActiveState. Vous devriez pouvoir trouver un binaire compilé à http://aspn.activestate.com/ASPN/Downloads/ActivePerl/. Les instructions suivantes supposent que vous utilisez la version 5.8.1 d'ActiveState.

4.1.2. Perl Modules on Win32

Bugzilla sous Windows nécessite l'utilisation des mêmes modules Perl que ceux de Section 1.5, « Modules Perl » [p 12] . La principale différence est que Windows utilise *PPM* [p 127] au lieu de CPAN.

```
C:\perl> ppm install <module name>
```

La meilleure source pour les modules PPM de Windows nécessaires pour Bugzilla est probablement le serveur de test de Bugzilla (alias 'landfill') donc vous devriez ajouter l'ensemble des PPM de landfill comme suit :

```
ppm repository add landfill http://www.landfill.bugzilla.org/ppm/
```

Note

Sur landfill, les modules PPM sont regroupés en 'paquetages' qui peuvent avoir un nom légèrement différent de celui du module. Si vous récupérez ces modules à partir de là, vous devrez faire attention aux informations fournies lorsque vous exécutez **checksetup.pl** car il vous dira quel package vous devrez installer.

Astuce

Si vous êtes derrière un pare-feu d'entreprise, vous devrez informer l'utilitaire PPM ActiveState sur la façon de le contourner afin d'accéder aux fichiers contenant les PPM en configurant la variable d'environnement du système HTTP_proxy. Pour plus d'information sur la configuration de cette variable, voir la documentation de ActiveState.

4.1.3. Changements de code nécessaires à l'exécution sur win32

Bugzilla sur win32 est en général supporté du premier coup; il reste le problème lié aux courriels concernant les bogues. Pour qu'ils fonctionnent sous Win32 (jusqu'a la résolution du bogue 49893), le moyen le plus simple est d'avoir le module de Perl Net::SMTP installé et de remplacer ces lignes dans le fichier Bugzilla/Bugmail.pm :

```
open(SENDMAIL, "|/usr/lib/sendmail $sendmailparam -t -i") ||
  die "Can't open sendmail";

print SENDMAIL trim($msg) . "\n";
close SENDMAIL;
```

par

```
use Net::SMTP;
my $smtp_server = 'smtp.mycompany.com';  # changez ceci

# Utiliser die en cas d'erreur pour que le mail aille dans les 'mails non envoyés' et
# puisse être envoyé de la page sanitycheck.
my $smtp = Net::SMTP->new($smtp_server) ||
  die 'Cannot connect to server \'$smtp_server\'';

$smtp->mail('bugzilla-daemon@mycompany.com');  # changez ceci
$smtp->to($person);
$smtp->data();
$smtp->datasend($msg);
$smtp->dataend();
$smtp->quit;
```

N'oubliez pas de changer le nom de votre serveur SMTP et le domaine de l'adresse d'envoi de courriel (après le '@') dans les lignes de codes ci-dessus.

4.1.4. Traitement des pages Web

Comme c'est le cas sur les systèmes basés sur Unix, tous les serveurs web devraient supporter Bugzilla ; toutefois, si on lui demande son avis, l'équipe de Bugzilla recommande toujours Apache. Peu importe quel serveur web vous choisissez, surtout faites attention aux notes de sécurité de Section 3.1, « Désactivation des accès à distance pour les fichiers de configuration Bugzilla » [p 54] . On trouvera plus d'informations sur la configuration de serveurs web particuliers dans la section Section 2.4, « Web server » [p 19] .

Note

Si vous utilisez Apache sous Windows, vous pouvez mettre les instructions ScriptInterpreterSource dans votre configuration d'Apache pour éviter d'avoir à modifier la première ligne de chaque script afin qu'il contienne votre chemin vers perl au lieu de /usr/bin/perl.

4.2. Mac OS X™

Apple n'a pas inclus la bibliothèque GD dans MacOS X. Bugzilla en a besoin pour les graphes de bogues.

Vous pouvez l'installer en utilisant un programme appelé Fink, qui est proche de l'installeur CPAN mais il installe les utilitaires GNU courants. Fink est disponible à l'adresse http://sourceforge.net/projects/fink/.

Suivez les instructions pour configurer Fink. Une fois qu'il est installé, vous devez l'utiliser pour installer le paquetage `gd2`.

Fink va alors vous demander s'il doit résoudre un certain nombre de dépendances. Saisissez « y » et appuyez sur la touche Entrée pour installer toutes les dépendances et surveillez alors que tout se passe bien. Vous pourrez alors utiliser *CPAN* [p 125] pour installer le module Perl GD.

Note

Pour ne pas entrer en conflit avec les logiciels installés par défaut par Apple, Fink crée sa propre arborescence dans `/sw` et y place la plupart des logiciels qu'il installe. Cela signifie que vos bibliothèques et vos entêtes seront dans `/sw/lib` et `/sw/include` au lieu de `/usr/lib` et `/usr/include`. Quand le script de configuration du module Perl vous demande où est votre `libgd`, dites lui bien `/sw/lib`.

`Expat` est également disponible via Fink. Après avoir utilisé fink pour installer le paquetage expat, vous pourrez installer XML::Parser en utilisant CPAN. Il y a une restriction. À la différence des versions récentes du module GD, XML::Parser n'invite pas à indiquer l'emplacement des bibliothèques nécessaires. Quand vous utilisez CPAN, vous devrez utiliser l'enchaînement des commandes suivant :

```
# perl -MCPAN -e'look XML::Parser'          ❶ # perl Makefile.PL EXPATLIBPATH=/sw/lib EXPATINCPATH=/sw/include
# make; make test; make install             ❷ # exit                                          ❸
```

❶ [p 27] ❸ [p 27] La commande look téléchargera le module et créera un nouveau shell avec les fichiers extraits comme répertoire de travail courant. La commande exit vous renverra à votre shell d'origine.

❷ [p 27] Il vous faudra surveiller le résultat de ces commandes make, surtout « make test » dont les erreurs peuvent empêcher XML::Parser de fonctionner correctement avec Bugzilla.

4.3. Linux-Mandrake 8.0

Linux-Mandrake 8.0 comporte toutes les bibliothèques obligatoires et facultatives pour Bugzilla. La manière la plus simple de les installer consiste à utiliser l'utilitaire **urpmi**. Si vous suivez ces commandes, vous devriez avoir tout ce qu'il vous faut pour Bugzilla et **./checksetup.pl** ne devrait pas se plaindre de bibliothèques manquantes. Il se peut que vous ayez certaines de ces bibliothèques déjà installées.

```
bash# urpmi perl-mysql
bash# urpmi perl-chart
bash# urpmi perl-gd
bash# urpmi perl-MailTools          ❶ bash# urpmi apache-modules
```

❶ [p 28] pour l'intégration de la messagerie Bugzilla

5. Notes d'installation sous UNIX (non administrateur)

5.1. Introduction

Si vous exécutez un *NIX OS avec un compte limité (non administrateur), soit en raison d'un manque d'accès (les hôtes web, par exemple) ou pour des raisons de sécurité, cette partie vous indiquera comment installer Bugzilla sur une telle installation. Il est recommandé que vous lisiez d'abord la Section 1, « Installation » [p 10] pour avoir une idée des étapes d'installation nécessaires (ces notes renverront à des étapes de ce guide).

5.2. MySQL

Il se peut que MySQL soit installé en tant qu'administrateur. Si vous êtes en train d'installer un compte avec un hôte web, un compte MySQL devra être créé pour vous. À partir de là, vous pouvez créer le compte bugs ou utiliser le compte qui vous est fourni.

Avertissement

Vous pourriez avoir des problèmes en essayant de régler les permissions **GRANT** à la base de données. Si vous utilisez un hôte web, il y a de grandes chances que vous ayez une base de données séparée qui est déjà verrouillée (ou une grosse base de données sans accès ou avec des accès limités aux autres zones) mais vous pouvez demander à votre administrateur système comment les réglages de sécurité ont été fixés, et/ou qu'il exécute la commande **GRANT** pour vous.

En plus, vous ne pourrez sans doute pas changer le mot de passe administrateur de MySQL (pour des raisons évidentes) alors sautez cette étape.

5.2.1. Exécuter MySQL comme non administrateur

5.2.1.1. La méthode de configuration personnalisée

Créez un fichier .my.cnf dans votre répertoire home (/home/foo est utilisé dans cet exemple) comme suit...

```
[mysqld]
datadir=/home/foo/mymysql
socket=/home/foo/mymysql/thesock
port=8081

[mysql]
socket=/home/foo/mymysql/thesock
port=8081

[mysql.server]
```

```
user=mysql
basedir=/var/lib

[safe_mysqld]
err-log=/home/foo/mymysql/the.log
pid-file=/home/foo/mymysql/the.pid
```

5.2.1.2. La méthode de construction personnalisée

Vous pouvez installer MySQL en tant que non « root », si vous le souhaitez vraiment. Construisez le avec `/home/foo/mysql` comme PREFIX ou utilisez des exécutables pré installés, en spécifiant que vous voulez mettre tous les fichiers de données dans `/home/foo/mysql/data`. S'il y un autre serveur MySQL qui s'exécute sur le système dont vous n'êtes pas le propriétaire, utilisez l'option -P pour indiquer un port TCP qui n'est pas utilisé.

5.2.1.3. Démarrage du serveur

Après que votre programme mysqld soit construit et que les éventuels fichiers .my.cnf soient en place, vous devez initialiser la base de données (UNE FOIS).

```
bash$ mysql_install_db
```

Ensuite démarrez le démon avec

```
bash$ safe_mysql &
```

Après le premier démarrage de mysqld, connectez vous en tant qu'administrateur et accordez les droits **GRANT** aux autres utilisateurs. (Encore une fois, le compte administrateur MySQL n'a rien à voir avec le compte administrateur *NIX.)

Note

Vous devrez démarrer les démons vous-même. Vous pouvez soit demander à votre administrateur système de les ajouter au fichier de démarrage, soit ajouter une entrée crontab qui exécute un script qui fera une vérification de ces démons et les redémarrera si besoin est.

Avertissement

N'exécutez PAS de démons ou d'autres programmes sur un serveur avant de consulter d'abord l'administrateur système ! Les démons consomment des ressources système et en exécuter un peut être en contradiction avec les règles d'utilisation de la machine sur laquelle vous êtes !

5.3. Perl

Dans le cas très rare où vous n'auriez pas Perl sur la machine, vous devrez construire les sources vous-même. Avec les commandes suivantes, vous devriez pouvoir installer votre propre version de Perl sur votre système avec :

```
bash$ wget http://perl.com/CPAN/src/stable.tar.gz
bash$ tar zvxf stable.tar.gz
bash$ cd perl-5.8.1 (ou quelque soit la version de Perl qui est appelée)
bash$ sh Configure -de -Dprefix=/home/foo/perl
bash$ make && make test && make install
```

Une fois que Perl est installé dans un répertoire (probablement dans `~/perl/bin`), vous devrez changer les emplacements dans les scripts, ce qui est expliqué plus loin dans cette page.

5.4. Les modules Perl

Installer les modules de Perl en tant que non administrateur est probablement la partie la plus difficile du processus. Il y a deux méthodes différentes : un Perl complètement indépendant avec ses propres modules, ou des modules personnels utilisant la version actuelle de Perl (installée par l'administrateur). La méthode indépendante prend pas mal d'espace disque, mais est moins complexe, alors que la méthode mixte n'utilise pas plus d'espace que les modules eux-mêmes, mais demande plus de travail à installer.

5.4.1. La méthode indépendante

La méthode indépendante nécessite l'installation de votre propre version de Perl, comme on l'a expliqué dans la section précédente. Une fois installée, vous pouvez lancer l'interpréteur de commandes CPAN avec la commande suivante :

```
bash$ /home/foo/perl/bin/perl -MCPAN -e 'shell'
```

Puis :

```
cpan> install Bundle::Bugzilla
```

Avec cette méthode, l'installation de module se fera généralement avec beaucoup moins de difficultés, mais si vous avez le moindre problème, vous pouvez consulter la section suivante.

5.4.2. La méthode mixte

Tout d'abord, vous allez devoir configurer CPAN pour installer les modules dans votre répertoire personnel. Voici ce que dit la FAQ de CPAN à ce sujet :

> 5) Je ne suis pas administrateur, comment puis-je installer un module dans un répertoire personnel ?
>
> Voici une façon de faire qui vous conviendra certainement :
>
> ```
> o conf makepl_arg "LIB=~/myperl/lib \
>                 INSTALLMAN1DIR=~/myperl/man/man1 \
>                 INSTALLMAN3DIR=~/myperl/man/man3"
> install Sybase::Sybperl
> ```
>
> Vous pouvez rendre ce réglage permanent comme tous les paramètres « o conf » à l'aide de :
>
> ```
> o conf commit
> ```

Vous devrez ajouter ~/myperl/man à la variable d'environnement MANPATH ainsi qu'indiquer à vos programmes Perl de regarder dans `~/myperl/lib`, par exemple en incluant :

```
use lib "$ENV{HOME}/myperl/lib";
```

ou en réglant la variable d'environnement PERL5LIB.

Une autre chose que vous devriez garder à l'esprit est que le paramètre UNINST ne devrait jamais être activé si vous n'êtes pas root.

Il vous faudra donc créer un répertoire Perl dans votre répertoire personnel, ainsi que les répertoires `lib`, `man`, `man/man1`, et `man/man3` dans ce répertoire Perl. Réglez la variable MANPATH et la variable PERL5LIB afin que l'installation des modules se fasse sans difficulté (régler UNINST=0 dans vos options « make install » lors de la première configuration de CPAN est aussi une bonne idée).

Ensuite, allez dans l'interpréteur de commandes CPAN :

```
bash$ perl -MCPAN -e 'shell'
```

À partir de là, vous allez devoir entrer la commande « o conf » présentée plus haut et valider les changements. Puis vous pouvez lancer l'installation :

```
cpan> install Bundle::Bugzilla
```

L'essentiel du processus d'installation de module devrait se passer sans accrocs. Cependant, il se peut que vous ayez des problèmes avec Template. Pour commencer, il va falloir essayer d'installer Template avec les options XS Stash activée. Si ça ne marche pas, il se peut qu'on vous balance des messages d'erreur du compilateur C et qu'on vous renvoie à l'invite de l'interpréteur de commande CPAN. Dans ce cas, recommencez l'installation, et désactivez cette option. (En fait, répondez non à toutes les questions sur Template.) Il est également possible que le processus échoue à quelques tests. Si, au final, le taux de réussite aux tests est raisonnable (90+%), forcez l'installation à l'aide de la commande suivante :

```
cpan> force install Template
```

Si vous le souhaitez, vous pouvez aussi installer les autres modules optionnels :

```
cpan> install GD
cpan> install Chart::Base
cpan> install MIME::Parser
```

5.5. Serveur HTTP

Idéalement, il faut également l'installer en tant que root et l'exécuter sous un compte serveur web particulier. Tant que le serveur Web permettra l'exécution des fichiers *.cgi hors d'un répertoire cgi-bin, ainsi qu'un moyen de refuser l'accès à certains fichiers (comme un fichier .htaccess), vous devriez bien vous en sortir.

5.5.1. Lancer Apache en tant qu'utilisateur non root

Vous pouvez lancer Apache en tant qu'utilisateur non root, mais le port attribué devra être supérieur à 1024. Si vous entrez **httpd -V**, vous obtiendrez une liste des variables qu'utilise votre copie système de httpd. L'une d'entre elles, à savoir HTTPD_ROOT, vous indique l'endroit où l'installation cherche ses informations de configuration.

À partir de ce point, vous pouvez copier les fichiers de configuration dans votre répertoire personnel pour commencer leur modification. Lorsque vous les éditez et que vous utilisez l'option -d pour outrepasser le HTTPD_ROOT compilé dans votre serveur web, vous prenez le contrôle de votre propre serveur Web personnalisé.

Note

Vous devrez démarrer les démons vous-même. Vous pouvez soit demander à votre administrateur système de les ajouter au fichier de démarrage, soit ajouter une entrée crontab qui exécute un script qui fera une vérification de ces démons et les redémarrera si besoin est.

Avertissement

N'exécutez PAS de démons ou d'autres programmes sur un serveur avant de consulter d'abord l'administrateur système ! Les démons consomment des ressources système et en exécuter un peut être en contradiction avec les règles d'utilisation de la machine sur laquelle vous êtes !

5.6. Bugzilla

Si vous deviez installer des modules Perl en tant qu'utilisateur non root (Section 5.4, « Les modules Perl » [p 30]) ou dans des répertoires non standards, vous devrez modifier les scripts, en spécifiant l'emplacement correct des modules Perl :

```
perl -pi -e 's@use strict\;@use strict\; use lib \"/home/foo/perl/lib\"\;@' \
    *cgi *pl Bug.pm processmail syncshadowdb
```

Modifiez `/home/foo/perl/lib` dans votre répertoire de bibliotèques Perl personnel. Vous pouvez vraisemblablement sauter cette étape si vous utilisez la méthode indépendante d'installation du module Perl.

Lorsque vous lancerez **./checksetup.pl** pour créer le fichier `localconfig`, il fera une liste des modules Perl qu'il trouvera. S'il en manque un, revenez en arrière et revérifiez l'installation du module depuis l'interpréteur de commandes CPAN, puis supprimez le fichier `localconfig` et réessayez.

Avertissement

L'option dans `localconfig` avec laquelle vous risquez d'avoir des problèmes est le groupe du serveur Web. Si vous ne parvenez pas à remonter jusqu'à `index.cgi` (vous obtenez une erreur du type Accés Interdit, vous devrez peut-être être assouplir vos droits d'accès et effacer le groupe du serveur Web. Bien entendu, cela peut représenter un risque pour la sécurité. Il est vrai qu'un interpréteur de commandes correctement sécurisé et/ou un accès limité aux comptes munis d'interpréteurs de commandes diminuent le risque au niveau sécurité, mais c'est à vous d'assumer.

Chapitre 3. Administrer Bugzilla

Table des matières

1. Configuration de Bugzilla

Pour configurer Bugzilla, il faut changer différents paramètres accessibles à partir du lien « Edit parameters » au bas de la page. Voici quelques-uns des paramètres importants de cette page. Il faut parcourir cette liste et les remplir correctement après l'installation de Bugzilla.

1. **maintainer** : Le paramètre « maintainer » est l'adresse électronique de la personne chargée de maintenir cette installation Bugzilla. Il n'est pas nécessaire que l'adresse soit celle d'un compte Bugzilla valide.

2. **urlbase** : Ce paramètre indique à votre installation Bugzilla le nom de domaine entier et le chemin du serveur web.

   Par exemple, si votre page de requête Bugzilla est `http://www.foo.com/bugzilla/query.cgi`, fixez votre paramètre « urlbase » à `http://www.foo.com/bugzilla/`.

3. **makeproductgroups** : permet de créer automatiquement ou pas des groupes quand des produits nouveaux sont créés.

4. **useentrygroupdefault** : Les produits de Bugzilla peuvent avoir un groupe associé, si bien que certains utilisateurs ne voient des bogues que dans certains produits. Quand ce paramètre a pour valeur « on », cela peut provoquer le fait que les commandes du groupe initial sur les produits fraîchement créés placent immédiatement tous les bogues fraîchement générés dans le groupe

ayant le même nom que le produit. Après la création initiale d'un produit, les commandes du groupe peuvent être réglées plus précisément sans interférence avec le mécanisme.

5. **shadowdb** : Un problème intéressant se pose quand Bugzilla atteint un niveau élevé d'activité en continu. MySQL supporte uniquement le verrouillage en écriture au niveau des tables. Cela signifie que, si quelqu'un doit apporter une modification à un bogue, il devra verrouiller la table tout entière jusqu'à la fin de l'opération. Le verrouillage en écriture bloque également la lecture jusqu'à ce que l'écriture soit terminée. Notez que des versions plus récentes de mysql acceptent le verrouillage des lignes en utilisant différents types de tables. Ces types sont plus lents que le type standard, et Bugzilla ne bénéficie pas encore des fonctionnalités comme les transactions qui justifieraient une diminution de la vitesse. L'équipe Bugzilla espère bien avoir des informations sur toute expérience relative au verrouillage des lignes et à Bugzilla.

   Le paramètre « shadowdb » a été conçu pour contourner cette limitation. Alors qu'un seul utilisateur est autorisé à écrire dans la table à un moment donné, la lecture peut continuer sans difficulté sur une copie fantôme en lecture seule de la base de donnée. Bien que votre base de données sera deux fois plus volumineuse, une base de donnée fantôme peut apporter une énorme amélioration des performances quand elle est implémentée sur des bases de données Bugzilla à gros trafics.

   À titre indicatif, sur du matériel relativement ancien, mozilla.org n'a pas eu besoin de « shadowdb » avant d'arriver à environ 40000 utilisateurs de Bugzilla avec plusieurs centaines de modifications et de commentaires sur les bogues de Bugzilla par jour.

   La valeur du paramètre définit le nom de la base de données fantôme contenant les bogues. Vous aurez besoin de configurer les paramètres de l'hôte et du port depuis la page paramètres, et de configurer une copie sur votre serveur de base de données de manière à ce que les mises à jour atteignent le miroir en lecture seul. Consultez la documentation de votre base de données pour plus d'informations.

6. **shutdownhtml** : Si vous devez éteindre Bugzilla pour des tâches d'administration, tapez un texte descriptif (en langage HTML imbriqué, si vous voulez) dans ce champ. Chaque personne qui essayera d'utiliser Bugzilla (y compris les administrateurs) recevra une page affichant ce texte. Les utilisateurs ne peuvent ni se connecter ni se déconnecter tant que shutdownhtml est activé.

   Note

   Bien que les possibilités habituelles de connexion soient désactivées tant que 'shutdownhtml' est activé, des protections sont mises en place pour protéger l'administrateur malchanceux qui perd sa connexion à Bugzilla. Si cela vous arrive, allez directement à `editparams.cgi` (en tapant le lien manuellement si nécessaire). Vous serez alors invité à vous connecter, et là (mais nulle part ailleurs) votre nom d'utilisateur/mot de passe seront acceptés.

7. **passwordmail** : Chaque fois qu'un utilisateur crée un compte, le contenu de ce paramètre (avec des adaptations) ainsi que son mot de passe lui est envoyé.

   Ajoutez le texte que vous souhaitez dans le champ du paramètre « passwordmail ». Par exemple, beaucoup de personnes utilisent ce champ pour donner de rapides indications sur la façon d'utiliser Bugzilla sur leur site.

8. **movebugs** : Cette option est une fonctionnalité non documentée permettant de déplacer les bogues entre des installations Bugzilla distinctes. Vous aurez besoin de comprendre le code source pour utiliser cette fonctionnalité. Veuillez consulter `movebugs.pl` dans votre arborescence de fichiers Bugzilla pour davantage d'informations, comme il se doit.

9. **useqacontact** : Permet de définir une adresse électronique pour chaque composant, en plus de celle du propriétaire par défaut, à laquelle seront envoyées des copies des nouveaux bogues.

10. **usestatuswhiteboard** : Ce paramètre indique si vous souhaitez un champ réinscriptible de forme libre associé à chaque bogue. L'avantage du « Status Whiteboard » est qu'il peut être supprimé ou modifié aisément, et qu'il fournit un champ interrogeable facilement pour l'indexation des bogues qui ont des traits en commun.

11. **whinedays** : Fixez ce paramètre au nombre de jours que vous souhaitez laisser les bogues à l'état « NEW » ou « REOPENED » avant de signaler aux personnes concernées qu'elles ont des bogues non traités. Si vous ne pensez pas utiliser cette fonctionnalité, il suffit tout simplement de ne pas installer la fonction cron de rappel (« whining cron ») comme cela est décrit dans les instructions d'installation, ou de fixer la valeur à « 0 » (aucun rappel).

12. **commenton*** : Tous ces champs vous permettent de signaler quelles modifications ne nécessitent pas de commentaires et celles qui doivent être commentées par leur auteur. Souvent, les administrateurs autorisent les utilisateurs à s'ajouter eux-mêmes à la liste des copies, à accepter des bogues, ou à changer le « Status Whiteboard » sans ajouter de commentaires sur les raisons du changement, et exigent cependant que la plupart des autres changements soient accompagnés d'une explication.

    Remplissez les options « commenton » conformément à la politique de votre site. Au minimum, il est bon de demander des commentaires quand les utilisateurs résolvent, réassignent ou rouvrent des bogues.

    Note

    Il est généralement préférable de demander le commentaire du développeur quand il résout un bogue. Il n'y pas grand chose de plus agaçant pour les utilisateurs de base de données de bogues que de rencontrer un bogue marqué comme « fixed » par un développeur sans aucun commentaire sur la nature de la réparation (ou même pour indiquer qu'il a vraiment été réparé !).

13. **supportwatchers** : Activer cette option permet aux utilisateurs de demander à recevoir des copies de tous les messages électroniques concernant un bogue particulier d'un autre utilisateur. Observer un utilisateur avec des permissions de groupe différentes ne permet pas pour autant de 'contourner' le système; les messages copiés restent toujours soumis aux permissions de groupe normales sur un bogue, et les « observateurs » seront seulement en copie sur des courriels provenant de bogues qu'ils seraient normalement autorisés à voir.

14. **noresolveonopenblockers** : Cette option évitera aux utilisateurs de résoudre des bogues considérés comme RESOLVED [FIXED] s'ils ont des dépendances non résolues. Seule la résolution FIXED est affectée. Les utilisateurs seront toujours en mesure de passer les bogues à une résolution autre que CORRIGES s'ils ont des bogues de dépendances non résolues.

2. Administration des utilisateurs

2.1. Créer l'utilisateur par défaut

Quand vous lancez pour la première fois checksetup.pl après l'installation de Bugzilla, on vous demandera le nom de l'administrateur (adresse électronique) et le mot de passe de ce « super utilisateur ». Si, pour une raison quelconque, vous supprimez ce compte, relancez checksetup.pl et on vous redemandera le nom d'utilisateur et le mot de passe de l'administrateur.

> **Astuce**
>
> Si vous souhaitez ajouter plus d'administrateurs, ajouter les au groupe « admin » et, éventuellement, éditez les groupes tweakparams, editusers, creategroups, editcomponents et editkeywords pour ajouter le groupe entier d'administrateurs à ces groupes.

2.2. Gérer les autres utilisateurs

2.2.1. Créer de nouveaux utilisateurs

Vos utilisateurs peuvent créer leurs propres comptes utilisateur en cliquant sur le lien « Nouveau compte » situé en bas de chaque page (en supposant qu'ils n'aient pas déjà ouvert une autre session sous un autre nom). Toutefois, si vous souhaitez créer les comptes utilisateur à l'avance, voici comment faire.

1. Après avoir ouvert votre session, cliquez sur le lien « Users » situé en bas de la page de requête, puis sur le lien « Add a new user ».

2. Remplissez le formulaire. Cette page est toute simple. Une fois que cela est fait, cliquez sur « Submit ».

   > **Note**
   >
   > En ajoutant un utilisateur de cette façon, *aucun* message électronique ne lui sera envoyé pour l'informer de son nom d'utilisateur et de son mot de passe. Bien que cela soit utile pour créer des comptes factices (par exemple, des observateurs qui transmettent et reçoivent des messages électroniques à un autre système, ou des adresses électroniques qui font partie d'une liste de diffusion), il est, en général, préférable de fermer sa session et d'utiliser le bouton « New account » pour créer des utilisateurs, car tous les champs obligatoires seront remplis à l'avance et l'utilisateur sera également informé du nom de son compte et de son mot de passe.

2.2.2. Modifier les utilisateurs

Pour voir un utilisateur particulier, recherchez-le en tapant son nom d'utilisateur dans la case prévue sur la page « Edit Users ». Pour voir tous les utilisateurs, laissez la case vide.

Vous pouvez effectuer des recherches de différentes façons grâce à la liste déroulante située à droite de la zone de texte. Vous pouvez rechercher à partir d'une sous-chaîne (par défaut sans tenir compte de la casse), d'une expression rationnelle ou d'une expression rationnelle *inverse*, qui trouve tous les utilisateurs qui ne correspondent pas à l'expression rationnelle (Veuillez vous reporter au manuel issu de la commande **man regexp** pour des informations sur la syntaxe des expressions rationnelles).

Une fois que vous avez trouvé votre utilisateur, vous pouvez changer les champs suivants :

- *Login Name* : en général, c'est l'adresse électronique entière de l'utilisateur. Toutefois, si vous utilisez le paramètre « emailsuffix », ce champ peut contenir uniquement le nom d'utilisateur. On peut noter que les utilisateurs peuvent désormais changer leur nom d'utilisateur eux-mêmes (en le remplaçant par une adresse électronique valide).

- *Real Name* : c'est le nom réel de l'utilisateur. On peut noter que Bugzilla n'exige pas cette information pour créer un compte.

- *Password* : vous pouvez changer le mot de passe utilisateur ici. Les utilisateurs peuvent demander un nouveau mot de passe automatiquement, vous ne devriez donc pas avoir à le faire souvent. Si vous voulez désactiver un compte, reportez-vous à la description de « Disable Text » ci-dessous.

- *Disable Text* : si vous entrez quelque chose dans cette case, ne serait-ce qu'un espace, vous empêchez l'utilisateur d'ouvrir sa session, ou d'apporter des modifications à des bogues via l'interface Internet. Le code HTML que vous tapez dans cette case est affiché à l'utilisateur quand il essaye d'effectuer une de ces actions. Il est conseillé d'y expliquer pourquoi le compte a été désactivé.

  Les utilisateurs ayant un compte désactivé continueront de recevoir des courriels de Bugzilla; en outre, ils seront incapables de se connecter eux-mêmes pour modifier leur propres préférences et l'arrêter. Si vous voulez qu'un compte (désactivé ou activé) arrête de recevoir des courriels, ajoutez le nom du compte (un compte par ligne) au fichier `data/nomail`.

  Note

  Même les utilisateurs dont le compte a été désactivé peuvent quand même transmettre des bogues via le portail de messagerie électronique, si il en existe une. Pour la sécurité des installations de Bugzilla, le portail de messagerie électronique ne doit *pas* être activé.

  Avertissement

  Ne désactivez pas tous les comptes administrateurs !

- *<groupname>* : si vous avez créé des groupes, comme « securitysensitive », des cases apparaîtront ici pour vous permettre d'ajouter ou de supprimer des utilisateurs à ces groupes.

- *canconfirm* : ce champ n'est utilisé que si vous avez autorisé l'état « unconfirmed ». Si vous activez ce champ pour un utilisateur, celui-ci pourra passer des bogues de l'état « Unconfirmed » à un état « Confirmed » (par exemple : l'état « New »).

- *creategroups* : cette option permet à un utilisateur de créer et détruire des groupes dans Bugzilla.

- *editbugs* : les utilisateurs peuvent seulement éditer les bogues dont ils sont responsables ou qu'ils ont signalés, sauf si ce champ est activé. Même si l'option n'est pas sélectionnée, les utilisateurs peuvent tout de même ajouter des commentaires à des bogues.

- *editcomponents* : cette option permet à un utilisateur de créer de nouveaux produits et composants, ainsi que de modifier et supprimer ceux auxquels aucun bogue n'est associé. Si un produit ou un composant a des bogues qui lui sont associés, ces bogues doivent être affectés à un autre produit ou composant pour que Bugzilla permette sa suppression.

- *editkeywords* : si vous utilisez la fonctionnalité de mot-clé de Bugzilla, en activant cette option vous permettrez à un utilisateur de créer et détruire des mots-clé. Comme toujours, les mots-clé pour des bogues existants qui contiennent le mot-clé que l'utilisateur veut détruire doivent être modifiés pour que Bugzilla autorise sa suppression.

- *editusers* : cette option permet à un utilisateur de faire ce que vous faîtes en ce moment : éditer d'autres utilisateurs. Cela permettra à ceux qui ont le droit de le faire de supprimer les droits administrateur d'autres utilisateurs ou de se les accorder. À activer avec précaution.

- *tweakparams* : cette option permet à un utilisateur de changer les paramètres de Bugzilla (en utilisant `editparams.cgi`).

- *<productname>* : ceci permet à un administrateur de spécifier les produits au sein desquels un utilisateur pourra voir des bogues. L'utilisateur devra quand même avoir le droit « editbugs » pour éditer les bogues dans ces produits.

3. Produits

Les *produits* [p ??] constituent la catégorie la plus importante dans Bugzilla, et représentent quasiment les produits réels embarqués. Par exemple, si votre entreprise conçoit des jeux vidéo, vous devriez avoir un produit par jeu, peut-être un produit « commun » pour les technologies utilisées dans de nombreux jeux, et peut-être quelques produits spéciaux (Site Internet, Administration, ...)

Une grande partie des réglages de Bugzilla est paramétrable pour un produit donné. Le nombre de « votes » disponibles par utilisateur est fixé pour un produit donné, tout comme le nombre de votes nécessaire pour faire passer un bogue automatiquement de l'état UNCONFIRMED à l'état NEW.

Pour créer un nouveau produit :

1. Sélectionnez « Produits » en bas de page

2. Sélectionnez le lien « Add » en bas à droite

3. Entrez le nom du produit et une description. Il est possible de remplir le champ Description en HTML.

Ne vous souciez pas des options « Closed for bug entry », « Maximum Votes per person », « Maximum votes a person can put on a single bug », « Number of votes a bug in this Product needs to automatically get out of the UNCOMFIRMED state », et « Version » pour le moment. Nous allons les aborder dans quelques instants.

4. Composants

Les composants sont des sous-sections des produits. Par exemple, le jeu vidéo que vous développez peut avoir un composant IHM, un composant « API », un composant « Sound System » et un composant « Plugins », chacun étant surveillé par un programmeur différent. Il est souvent pratique de séparer les composants dans Bugzilla suivant la répartition naturelle des responsabilités au sein du produit ou de votre entreprise.

Chaque composant a un propriétaire et un contact QA (si vous l'activez dans les paramètres). Il est préférable que le propriétaire d'un composant soit la première personne à réparer les bogues dans ce composant. Il est conseillé que ce soit le contact QA qui s'assurera que ces bogues sont complètement réparés. Le propriétaire, le contact QA et celui qui a signalé le bogue recevront un message électronique quand de nouveaux bogues seront créés dans ce composant et quand ces bogues seront modifiés. Les champs Default Owner et QA contact indique seulement les *affectations par défaut*; celles-ci peuvent être modifiées lors de la soumission d'un bogue, ou à n'importe quel moment de la vie du bogue.

Pour créer un nouveau composant :

1. Sélectionnez le lien « Edit components » dans la page « Edit product ».
2. Sélectionnez le lien « Add » en bas à droite.
3. Remplissez le champ « Component », une courte « Description », les champs « Initial Owner » et « Initial QA Contact » (s'ils sont activés). Il est possible de remplir les champs Component et Description en HTML. Vous devez remplir le champ « Initial Owner » avec un nom d'utilisateur déjà présent dans la base de données.

5. Versions

Les versions sont les révisions d'un produit, comme « Flinders 3.1 », « Flinders 95 », et « Flinders 2000 ». La version n'est pas un champ à plusieurs choix; en général, on choisit la version la plus récente dont on sait qu'elle contient le bogue.

Pour créer et éditer des versions :

1. Sélectionnez le lien « Edit Versions » à partir de la page « Edit product ».
2. Vous pourrez remarquer que le produit a déjà par défaut la version « undefined ». Cliquez sur le lien « Add » en bas à droite.
3. Entrez le nom de la version. Ce champ ne peut contenir que du texte. Ensuite, cliquez sur le bouton « Add ».

6. Cibles Jalon

Les cibles jalons sont des « cibles » qui représentent la version pour laquelle vous prévoyez de réparer un bogue. Par exemple, si vous prévoyez de réparer un bogue pour la version 3.0, vous lui affecterez le jalon 3.0.

Note

Les options des cibles jalons n'apparaissent pour un produit que si vous activez le paramètre « usetargetmilestone » dans la page « Edit Parameters ».

Pour créer de nouveaux jalons, définir les jalons par défaut et fixer l'URL des jalons :

1. Sélectionnez « Edit milestones » dans la page « Edit product ».
2. Sélectionnez « Add » dans le coin inférieur droit.
3. Entrez le nom du jalon dans le champ « Milestone ». En option, vous pouvez fixer le « sortkey », qui est un nombre positif ou négatif (-255 à 255) qui définit où ce point clé apparaîtra dans la liste. En effet, les jalons ne se présentent pas souvent dans l'ordre alphanumérique. Par exemple, on peut trouver « Future » après « Release 1.2 ». Sélectionnez « Add ».
4. Dans la page Edit product, vous pouvez entrer l'URL d'une page qui donne des informations sur vos jalons et ce qu'ils signifient.

7. Fanions

Les fanions permettent d'attribuer un statut spécifique à un bogue ou une pièce jointe, aussi bien « + » que « - ». La signification de ces symboles dépend du texte du fanion lui-même, mais selon le contexte ils peuvent signifier passe/échoue, accepter/refuser, approuvé/refusé, ou même un simple oui/non. Si votre site autorise les fanions de requêtes, les utilisateurs peuvent attribuer au fanion la valeur « ? » de manière à en faire une requête auprès d'un autre utilisateur pour pouvoir regarder le bogue/pièce jointe, et donner au fanion son statut correct.

7.1. Exemple simple

Un développeur peut avoir envie de demander à sa hiérarchie : « Devrions nous corriger ce bogue avant de sortir la version 2.0 ? ». Ils peuvent avoir envie de le faire pour de *nombreux* bogues, et donc ce serait bien de rationaliser le procédé...

Avec Bugzilla, cela marcherait ainsi :

1. L'administrateur de Bugzilla crée un type de fanion appelé « blocking2.0 » qui révèle tous les bogues de votre produit.

   Sur l'écran « Trouver bug », ce fanion affiche le texte « blocking2.0 » avec une zone déroulante à côté. La zone déroulante contient 4 valeurs : un espace vide, « ? », « - », et « + ».
2. Le développeur attribue au fanion la valeur « ? ».
3. Le directeur voit le fanion `blocking2.0` à la valeur « ? » value.
4. Si le directeur pense que le dispositif devrait être inclus dans le produit avant la sortie de la version 2.0, il attribue au fanion la valeur « + ». Sinon il lui attribue « - ».
5. Maintenant, chaque utilisateur de Bugzilla qui voit le bogue sait si le bogue a besoin ou pas d'être corrigé avant la sortie de la version 2.0.

7.2. À propos des fanions

7.2.1. Valeurs

Les fanions peuvent prendre 3 valeurs :

?
: Un utilisateur demande qu'un statut soit donné (considérez le comme « une question est posée »).

–
: Le statut donné est négatif (on a répondu « non » à la question).

+
: Le statut donné est positif (on a répondu « oui » à la question).

En fait, il existe une quatrième valeur possible du fanion : « aucun statut » ; celle-ci est représentée par un espace vide. Cela veut juste dire que personne n'a exprimé d'opinion (ou demandé à quelqu'un d'autre d'exprimer une opinion) à propos de ce bogue ou de la requête par pièce jointe.

7.3. Utilisation des requêtes par fanions

Si un fanion a été déclaré comme étant fanion « de requête », les utilisateurs sont autorisés à attribuer au fanion le statut « ? ». Ce statut indique que quelqu'un (alias « le demandeur ») demande à quelqu'un d'autre d'attribuer « + » ou « - ».

Si un fanion a été défini comme « fanion de requête particulière », une zone de texte apparaîtra à côté du fanion dans laquelle le demandeur peut entrer un nom d'utilisateur Bugzilla. La personne mentionnée (alias « le demandé ») recevra un courriel l'avertissant de la requête et renvoyant celle-ci vers le bogue ou la pièce jointe en question.

Si un fanion n'a *pas* été défini « fanion de requête particulière », aucune zone de ce genre n'apparaîtra. Une requête pour marquer ce fanion ne peut pas être faite à une personne en particulier, mais doit être demandée « à la cantonade ». Un demandeur peut « demander à la cantonade » pour n'importe quel fanion simplement en laissant la zone de texte vide.

7.4. Deux types de fanions

Les fanions peuvent se mettre à deux endroits : dans une pièce jointe, ou dans un bogue.

7.4.1. Fanions de pièce jointe

Les fanions de pièce jointe sont utilisés pour poser une question sur une pièce jointe particulière à un bogue.

Plusieurs installations Bugzilla s'en servent pour demander à un développeur « d'examiner » [« review »] le code d'un autre développeur avant de le valider. Ils joignent le code à un rapport de bogue, puis lui attribue un fanion nommé « review » à `review?boss@domain.com`. boss@domain.com est alors averti par courriel qu'il doit vérifier la pièce jointe et l'approuver ou la refuser.

Pour un utilisateur Bugzilla, les fanions de pièce jointe apparaissent à deux endroits :

1. Sur la liste des pièces jointes sur l'écran « montrer un bogue » [« Show bug »], on peut voir l'état actuel de tous les fanions affectés de ?, +, ou -. On peut voir qui a demandé le fanion (le demandeur), et qui a été sollicité (le demandé).

2. Quand vous « modifiez » une pièce jointe, on peut voir les fanions qui n'ont pas encore de valeur, ainsi que ceux qui en ont déjà une. C'est sur l'écran « éditer une pièce jointe » que l'on met ou que l'on enlève ?, -, +.

7.4.2. Fanions de bogue

Les fanions de bogue sont utilisés pour attribuer un statut au bogue lui-même. On peut voir les fanions de bogue sur l'écran « Show Bug » (`editbug.cgi`).

Seuls les utilisateurs ayant le droit d'éditer le bogue peuvent attribuer des valeurs aux fanions qui sont sur des bogues. Cela inclut le propriétaire, celui qui signale les bogues, et tout utilisateur ayant la permission `editbugs`.

7.5. Administration des fanions

Si vous avez le privilège « editcomponents », vous aurez « Edit: ... | Flags | ... » en bas de page. En cliquant sur ce lien vous pourrez accéder à la page « Administrer des types de fanions ». Ici, vous pouvez choisir si vous voulez créer (ou éditer) un fanion de bogue, ou un fanion de pièce jointe.

Peu importe celui que vous choisissez, l'interface est la même, donc nous ne l'aborderons qu'une fois.

7.5.1. Créer un fanion

Quand vous cliquez sur le lien « Create a Flag Type for... », il vous sera présenté un formulaire. Voici la signification des champs du formulaire :

7.5.1.1. Nom

C'est le nom du fanion. Il s'affichera pour les utilisateurs Bugzilla qui sont en train de regarder ou de configurer un fanion. Le nom peut contenir n'importe quel caractère Unicode valide.

7.5.1.2. Description

Donne plus de précisions sur le fanion. Pour le moment, cela n'a pas l'air d'être très utile; idéalement, ce serait bien que ce soit affiché comme aide. Ce champ peut être aussi long que vous le désirez, et peut contenir autant de caractères que vous voulez.

7.5.1.3. Catégorie

Le comportement par défaut pour un fanion fraîchement créé est d'apparaître sur les produits et tous les composants, c'est pourquoi « __Any__:__Any__ » est déjà présent dans le champ « Inclusions ». Si ce n'est pas ce comportement que vous souhaitez, vous pouvez également configurer des exclusions (pour les produits sur lesquels vous ne voulez pas que le fanion apparaisse), ou bien vous devez enlever « __Any__:__Any__ » du champ Inclusion et définir les produits/composants spécialement pour ce fanion.

Pour créer une Inclusion, sélectionnez un produit à partir du menu déroulant du haut. Vous pourrez également sélectionner un composant donné à partir de ce menu déroulant. (La configuration « __Any__ » pour les produits se traduit par « tous les produits de ce Bugzilla ». Choisir « __Any__ » dans le champ Composants veut dire « tous les composants du produit sélectionné ».) Une fois les sélections faites, appuyer sur « Include », et l'appariement de votre produit/composant sera visible dans le champ « Inclusions » sur la droite.

Pour créer une Exclusion, la procédure est la même; choisissez un produit à partir du menu déroulant du haut, choisissez un composant donné si vous en voulez un, et appuyez sur « Exclude ». L'appariement de votre produit/composant sera visible dans le champ « Exclusions » sur la droite.

Ce fanion *devra* et *peut* être configuré pour n'importe quel produit/composant apparaissant dans le champ « Inclusions » (ou qui dépend du « __Any__ » approprié). Ce fanion n'apparaîtra (et donc ne pourra être configuré) sur *aucun* des produits apparaissant dans le champ « Exclusions ». *IMPORTANT : les Exclusions prévalent sur les Inclusions.*

Vous pouvez choisir un produit sans sélectionner un composant particulier, mais choisir un composant sans produit est contraire à la règle, de même que choisir un composant qui n'appartient pas au produit évoqué. Si vous le faites, vous obtiendrez une erreur, comme on l'indique dans ce texte (2.18rc3)... même si tous vos produits possèdent un composant du même nom.

Exemple : disons que vous avez un produit appelé « Avion » qui contient des centaines de composants. Vous voulez avoir la possibilité de demander si un problème sera corrigé dans le prochain modèle d'avion que vous sortirez. Nous appellerons ce fanion « corrigerDansSuivant ». Cependant, il y a un composant dans « Jet », appelé « Pilote ». Il ne vous sert à rien de sortir un nouveau pilote, donc vous ne voulez pas que le fanion soit visible pour ce composant. Donc, vous incluez « Jet :__Any__ » et vous excluez « Jet :Pilote ».

7.5.1.4. Clé de tri

Les fanions sont normalement visibles dans l'ordre alphabétique. Si vous voulez les afficher dans un ordre différent vous pouvez utiliser ce champ pour configurer l'ordre sur chaque fanion. Les fanions ayant une clé de tri inférieur apparaîtront avant les fanions à clés de tris supérieur. Les fanions ayant le même critère seront rangés alphabétiquement mais ils seront toujours après les fanions à clés de tri inférieur et avant ceux à clés supérieur.

Exemple : j'ai AFanion (critère de tri 100), BFanion (critère de tri 10), CFanion (critère de tri 10), et DFanion (critère de tri 1). Ceux-ci s'affichent dans cet ordre : DFanion, CFanion, BFanion, AFanion.

7.5.1.5. Actif

Il vous arrivera de vouloir conserver les informations de l'ancien fanion dans la base de données Bugzilla, tout en empêchant les utilisateurs d'initialiser de nouveaux fanions de ce genre. Pour ce faire, dé-sélectionnez « active ». Les fanions désactivés seront toujours visibles dans l'interface s'ils sont ?, +, ou -, mais ils peuvent seulement être effacés (non initialisés), et ne peuvent pas avoir de nouvelles valeurs. Une fois que le fanion désactivé est supprimé, il disparaîtra complètement du bogue/pièce jointe, et ne pourra plus être reconfiguré.

7.5.1.6. Fanions de requête

Les nouveaux fanions sont, par défaut, des fanions « de requête », ce qui signifie qu'ils offrent aux utilisateurs l'option « ? », ainsi que « + » et « - ». Pour supprimer l'option ? , dé-sélectionnez « requestable ».

7.5.1.7. Liste CC

Si vous voulez que certains utilisateurs soient avertis à chaque fois qu'un fanion est initialisé à ?,-,+, ou non initialisé, ajoutez les ici. Il s'agit d'une liste d'adresses électroniques, séparées par des virgules, qu'il n'est pas nécessaire de restreindre aux noms d'utilisateurs Bugzilla.

7.5.1.8. Fanions de requêtes spécifiques

Par défaut cette case est cochée pour les nouveaux fanions, ce qui signifie que chaque utilisateur peut faire des requêtes de fanions destinées à des personnes en particulier. Décocher cette boîte supprimera le champ de texte à côté du fanion; s'il demeure fanion de requête, les demandes ne peuvent être faites qu' « à la cantonnade ». Le supprimer après que des demandes spécifiques aient été faites ne supprimera pas ces demandes; ces données vont rester dans la base de données (bien que cela n'apparaîtra plus aux utilisateurs).

7.5.1.9. Multipliable

Tout fanion initialisé sur « Multipliable » (par défaut pour les nouveaux fanions) peut être configuré plus d'une fois. Après avoir été initialisé une fois, un fanion non initialisé du même type apparaîtra en dessous de lui avec « addl. » (abréviation de « additional ») sous son nom. Il n'y a pas de limite au nombre de fois qu'un fanion Multipliable peut être initialisé sur le même bogue/pièce jointe.

7.5.2. Supprimer un fanion

Sur le boîte de dialogue « Administrer les types de fanions », vous verrez une liste de fanions de bogues et une liste de fanions de pièces jointes.

Pour supprimer un fanion, cliquez sur le lien « Delete » à côté de la description du fanion.

> **Avertissement**
>
> Une fois que vous avez supprimé un fanion, il n'est *plus* dans votre Bugzilla. Toutes les données de ce fanion seront supprimées. Il disparaîtra de tous les endroits où il était installé, et vous ne pourrez pas récupérer les données. Si vous voulez garder les données d'un fanion, mais ne voulez pas que n'importe qui configure de nouveaux fanions ou modifie les fanions courants, dé-sélectionnez « active » dans le formulaire d'édition des fanions.

7.5.3. Éditer un fanion

Pour éditer les propriétés d'un fanion, il suffit de cliquer sur le lien « Edit » à côté de la description du fanion. Cela vous ramènera au formulaire décrit dans la partie « Créer un fanion ».

8. Le vote

Le vote permet aux utilisateurs de disposer d'un certain nombre de voix qu'ils peuvent affecter à des bogues, pour indiquer qu'ils souhaitent que ces bogues soient réparés. Cela permet aux développeurs d'évaluer les besoins utilisateurs pour une amélioration ou une réparation particulière. En permettant que les bogues ayant un certain nombre de voix puissent passer automatiquement de « UNCONFIRMED » à « NEW », les utilisateurs du système de bogues peuvent attirer l'attention sur les bogues de priorité élevée de façon à ce qu'ils ne restent pas trop longtemps à en attente de triage.

Pour modifier les paramètres de vote :

1. Allez jusqu'à la page « Edit product » du produit que vous voulez modifier.
2. *Nombre de voix par personne* : en mettant ce champ à 0, on désactive le vote.
3. *Nombre de voix maximum qu'une personne peut donner à un seul bogue* : à l'évidence, cela doit être un nombre inférieur au nombre inscrit dans le champ « Nombre de voix par personne ». Ne mettez pas ce champ à 0 si « Nombre de voix par personne » n'est pas à 0; cela n'a pas de sens.
4. *Nombre de voix nécessaires pour qu'un bogue de ce produit sorte automatiquement de l'état UNCONFRIMED.* : En mettant ce champ à « 0 », on désactive le passage automatique des bogues de UNCONFIRMED à NEW.
5. Une fois que vous avez ajusté les valeurs selon vos préférences, cliquez sur « Update ».

9. Les mots d'esprit

Les mots d'esprit sont des messages texte courts qui peuvent être configurés pour apparaître à côté des résultats d'une recherche. Une installation Bugzilla peut avoir ses propres mots d'esprit bien spécifiques. Quel que soit le moment où le mot d'esprit a besoin d'être affiché, une sélection aléatoire est faite sur l'ensemble des mots d'esprits déjà existants.

Les mots d'esprit sont gérés par le paramètre *enablequips*. Il a plusieurs valeurs possibles : actif, approuvé, gelé ou éteint. Pour permettre l'approbation de mots d'esprit, il vous faut initialiser ce paramètre à « approved ». De cette manière, les utilisateurs sont libres de proposer des mots d'esprits en plus mais un administrateur doit les approuver explicitement avant qu'ils puissent être en fait utilisés.

Pour voir l'interface utilisateur pour les mots d'esprits, il suffit de cliquer sur l'un d'entre eux quand il est affiché avec les résultats de la recherche. On peut aussi le voir directement dans le navigateur en allant sur le lien quips.cgi (préfixé de l'adresse WEB habituelle de l'installation Bugzilla). Dès que l'interface des mots d'esprit s'affiche, il suffit de cliquer sur « voir et modifier la liste complète des mots d'esprit » afin de voir la page d'administration. Une page avec tous les mots d'esprits disponibles dans la base de données s'affichera.

À côté de chaque mot d'esprit, il y a une case à cocher, dans la colonne « approved ». Les mots d'esprits qui ont leur case cochée sont déjà approuvés et apparaîtront après chaque résultat de recherche. Ceux dont la case est non cochée sont toujours présents dans la base de données mais n'apparaîtront pas sur les pages de résultats de recherche. Pour les mots d'esprit proposés par les utilisateurs, cette case est, au départ, non cochée.

Il y a également un lien de suppression à côté de chaque mot d'esprit, lequel peut être utilisé pour supprimer définitivement un mot d'esprit.

10. Groupes et sécurité des groupes

Les groupes permettent à l'administrateur d'isoler des bogues ou produits qui ne devraient être vus que par certaines personnes. L'association entre les produits et les groupes est gérée à partir de la page d'édition du produit sous « éditer les commandes des groupes ».

Si le paramètre makeproductgroups est activé, un nouveau groupe sera automatiquement créé pour chaque nouveau produit. Il est principalement disponible pour la compatibilité descendante avec des sites plus anciens.

Notez que les permissions de groupes sont telles que vous devez être membre de *tous* les groupes où se trouve un bogue, quelle qu'en soit la raison, pour voir ce bogue. De même, vous devez être membre de *tous* les groupes d'entrée d'un produit pour ajouter des bogues à un produit et vous devez être membre de *tous* les groupes « canedit » (« peut modifier ») d'un produit pour faire *toute* modification de bogues dans ce produit.

Note

Par défaut, les bogues peuvent également être vus par le propriétaire du bogue, le rapporteur et par toutes les personnes de la liste CC, quel que soit leur statut par rapport à leur accès normal au bogue. La visibilité pour le rapporteur et pour la liste CC peut être annulée (pour un bogue à la fois) en ressortant ce bogue, puis en allant à la section qui commence par « Utilisateurs dans les rôles sélectionnés ci-dessous... » et en décochant la case à côté de « rapporteur » ou « Liste CC » (ou les deux).

10.1. Création des groupes

Pour créer des groupes :

1. Sélectionnez le lien « groups » dans le bas de page.

2. Lisez attentivement les instructions sur l'écran « Edit Groups », puis sélectionnez le lien « Add Groups ».

3. Remplissez les champs « Group », « Description », et « User RegExp ». « User RegExp » vous permet de placer automatiquement tous les utilisateurs qui remplissent les conditions de l'expression rationnelle dans le nouveau groupe. Quand vous avez fini, cliquez sur « Add ».

   Les utilisateurs dont les adresses électroniques correspondent à l'expression rationnelle seront automatiquement membres du groupe tant que leurs adresses électroniques continueront de correspondre à l'expression rationnelle.

Note

C’est une modification du 2.16 où l’expression rationnelle avait pour résultat l’adhésion permanente à un groupe pour un utilisateur. Pour supprimer un utilisateur d’un groupe dans lequel il était à cause d’une expression rationnelle, dans la version 2.16 ou précédemment, l’utilisateur doit être supprimé explicitement du groupe.

Avertissement

Si vous spécifiez un domaine dans le regexp, assurez vous que vous avez bien terminé l’expression rationnelle avec un « $ ». Sinon quand vous autoriserez l’accès à `@monentreprise\.com`, vous autoriserez l’accès à `mauvaisepersonne@monentreprise.com.cracker.net`. Il vous faut utiliser `@monentreprise\.com$` comme expression rationnelle.

4. Si vous décidez d’utiliser ce groupe pour contrôler directement l’accès aux bogues, cochez la case « use for bugs » . Les groupes non utilisés pour des bogues restent utiles car les autres groupes peuvent inclure le groupe entier.

5. Après avoir ajouté votre nouveau groupe, éditez le nouveau groupe. Sur la page d’édition, vous pouvez spécifier d’autres groupes qui devraient être inclus dans ce groupe et quels groupes devraient être autorisés à ajouter ou à supprimer des utilisateurs à partir de ce groupe.

10.2. Affecter des utilisateurs aux groupes

Les utilisateurs peuvent devenir membres d’un groupe de plusieurs façons.

1. L’utilisateur peut être explicitement placé dans le groupe en éditant le propre profil de l’utilisateur.

2. Le groupe peut contenir un autre groupe auquel l’ utilisateur appartient.

3. L’adresse électronique de l’utilisateur peut correspondre à une expression rationnelle que le groupe définit pour attribuer automatiquement le statut du membre.

10.3. Affecter les commandes de groupe aux produits

Sur la page d’édition du produit, il y a une page pour éditer le « Group Controls » d’un produit. Cela vous permet de paramétrer le lien qui existe entre un groupe et un produit. Les groupes peuvent être applicables, par défaut et obligatoires ainsi qu’être utilisés pour contrôler qui a le droit d’entrer un rapport de bogue ou pour limiter à un groupe particulier le droit de modifier les rapports de bogue.

Pour chaque groupe, il est possible de spécifier si être membre de ce groupe...

1. est nécessaire pour entrer un bogue (Entry).

2. est sans objet pour ce produit (NA), est une restriction que les membres de ce groupe peuvent ajouter à ce produit (Shown), est une restriction par défaut lorsque les membres de ce groupe ajoutent un bogue à ce produit (Default), ou est une restriction obligatoire à placer sur les bogues de ce produit (Mandatory).

3. est sans objet pour ce produit pour les non membres (NA), est une restriction que les non membres de ce groupe peuvent ajouter à ce produit (Shown), est une restriction par défaut lorsque les non membres de ce groupe ajoutent un bogue à ce produit (Default), ou est une restriction obligatoire à placer sur les bogues de ce produit lorsqu'ils sont entrés par des non membres (Mandatory).

4. est nécessaire pour faire *toute* modification sur les bogues de ce produit *y compris des commentaires*.

Ces paramètres sont souvent indiqués dans cet ordre. Par exemple, pour un produit qui demande, pour entrer un bogue, qu'un utilisateur soit membre du groupe « foo » et demande ensuite que le bogue reste en permanence restreint au groupe « foo » et enfin que seuls les membres du groupe « foo » puissent modifier ce bogue, bien que celui-ci puisse être visible pour d'autres, le paramétrage du groupe serait résumé ainsi :

```
foo: ENTRY, MANDATORY/MANDATORY, CANEDIT
```

10.4. Applications courantes des commandes de groupe

10.4.1. Accès utilisateur général avec le groupe Securité

Pour laisser tout utilisateur soumettre des bogues dans chaque produit (A, B, C...) et laisser tout utilisateur soumettre ces bogues dans un groupe de sécurité...

```
Produit A...
security: SHOWN/SHOWN
Produit B...
security: SHOWN/SHOWN
Produit C...
security: SHOWN/SHOWN
```

10.4.2. Accès utilisateur général avec un produit Security

Pour laisser tout utilisateur soumettre des bogues dans un produit de sécurité tout en gardant ces bogues invisibles à toute personne étrangère au groupe securityworkers à moins qu'un membre du groupe securityworkers enlève cette restriction...

```
Product Security...
securityworkers: DEFAULT/MANDATORY
```

10.4.3. Isolation du produit avec un groupe commun

Pour laisser les utilisateurs du produit A accéder aux bogues du produit A, les utilisateurs du produit B à accéder au produit B et l'équipe du support à accéder aux deux, 3 groupes sont nécessaires

1. Support : contient les membres de l'équipe de support.

2. AccessA : contient les utilisateurs du produit A et le groupe Support.

3. AccessB : contient les utilisateurs du produit B et le groupe Support.

Une fois ces 3 groupes définis les commandes de groupe des produits peuvent être fixées à...

```
Product A...
AccessA: ENTRY, MANDATORY/MANDATORY
Product B...
AccessB: ENTRY, MANDATORY/MANDATORY
```

Si on le souhaitait, le groupe Support pourrait être autorisé à rendre les bogues inaccessibles aux utilisateurs et pourrait être autorisé à publier les bogues concernant tous les utilisateurs dans un produit courant qui est en lecture seule pour toute personne extérieure au groupe de soutien. Cette configuration pourrait être...

```
Product A...
AccessA: ENTRY, MANDATORY/MANDATORY
Support: SHOWN/NA
Product B...
AccessB: ENTRY, MANDATORY/MANDATORY
Support: SHOWN/NA
Product Common...
Support: ENTRY, DEFAULT/MANDATORY, CANEDIT
```

11. Mise à niveau aux nouvelles versions

Avertissement

Mettre à niveau est une opération à sens unique. Il est conseillé de faire une copie de sauvegarde de votre base de données et répertoire courant Bugzilla avant de tenter une mise à niveau. Si vous souhaitez repasser à la version précédente pour une raison quelconque, vous devrez restaurer à partir de ces copies de sauvegarde.

Mettre à niveau Bugzilla est une chose que nous voulons tous faire de temps en temps, que ce soit pour récupérer de nouvelles fonctionnalités ou les dernières failles de sécurité. La facilité à télécharger dépend de quelques facteurs :

- Si la nouvelle version est une révision ou version intermédiaire
- Le nombre de modifications en local (s'il y en a) qui ont été faites

Il existe trois manières différentes de mettre à jour votre installation.

1. En utilisant un CVS (Exemple 3.1, « Mise à niveau par le CVS » [p 50])
2. En téléchargeant une nouvelle archive (Exemple 3.2, « Mettre à niveau avec le tarball » [p 51])
3. En appliquant les programmes de correction appropriés (Exemple 3.3, « Mise à niveau avec les correctifs » [p 51])

Chacune de ces options a ses propres avantages et inconvénients; celle qui vous correspond dépend du temps écoulé depuis la dernière fois que vous avez fait une installation et/ou de votre configuration réseau.

Les révisions sont normalement officialisées seulement lorsque des points faibles au niveau sécurité sont en cause; ils se distinguent par une augmentation du troisième chiffre. Par exemple, lorsque 2.16.6 est sorti, c'était une amélioration de la 2.16.5.

Les versions intermédiaires sont généralement émises lorsque l'équipe de développement Bugzilla estime que suffisamment de progrès a été fait globalement. Elles sont souvent obtenus par une période de stabilisation et de ***release candidates***, par contre, l'utilisation des versions de développement et de ***release candidates*** échappe à la portée de ce document. Les versions intermédiaires se distinguent par une augmentation dans le deuxième numéro, celui de version mineure. Par exemple, 2.18.0 est une version intermédiaire plus récente que 2.16.5.

Les exemples des chapitres suivants sont rédigés comme si l'utilisateur passait à la version 2.18.1, mais les procédures sont les mêmes que l'on passe à une nouvelle version intermédiaire ou que l'on essaie simplement de récupérer une nouvelle version bugfix. Par contre, le risque d'avoir des problèmes est plus grand lorsque 'on passe à une version intermédiaire, surtout si des modifications locales ont été effectuées.

Dans ces exemples, l'installation Bugzilla de l'utilisateur se trouve à `/var/www/html/bugzilla`. Si ce n'est pas le même emplacement pour votre installation Bugzilla, vous n'avez qu'à remplacer par les chemins qui conviennent là où c'est nécessaire.

Exemple 3.1. Mise à niveau par le CVS

Chaque version de Bugzilla, que ce soit une version intermédiaire ou un bugfix, est marquée dans le CVS. Ainsi, chaque tarball qui a été distribué depuis la version 2.12 a été créé de telle sorte qu'il peut être utilisé avec le CVS une fois dépaqueté. Une telle manière de procéder exige cependant que vous puissiez accéder à cvs-mirror.mozilla.org sur le port 2401.

Astuce

Si vous en avez la possibilité, la mise à jour par le CVS est probablement la méthode la moins pénible, particulièrement si vous avez beaucoup de modifications locales.

```
bash$ cd /var/www/html/bugzilla
bash$ cvs login
Logging in to :pserver:anonymous@cvs-mirror.mozilla.org:2401/cvsroot
CVS password: anonymous
bash$ cvs -q update -r BUGZILLA-2_18_1 -dP
P checksetup.pl
P collectstats.pl
P globals.pl
P docs/rel_notes.txt
P template/en/default/list/quips.html.tmpl
```

Attention

Si une ligne en sortie de **cvs update** commence avec un `C`, cela indique un fichier avec des modifications locales que le CVS a été incapable d'intégrer. Vous avez besoin de résoudre ces conflits manuellement avant que Bugzilla (ou au moins la partie utilisant ce fichier) devienne utilisable.

Note

vous aurez besoin d'exécuter **./checksetup.pl** avant que votre mise à niveau de Bugzilla soit achevée.

Exemple 3.2. Mettre à niveau avec le tarball

Si vous être dans l'incapacité ou n'avez pas envie d'utiliser le CVS, une autre option toujours possible est d'obtenir la dernière archive tar. Ceci est la plus difficiles des options à utiliser, surtout si vous avez des modifications locales.

```
bash$ cd /var/www/html
bash$ wget ftp://ftp.mozilla.org/pub/mozilla.org/webtools/bugzilla-2.18.1.tar.gz
Affichage omis
bash$ tar xzvf bugzilla-2.18.1.tar.gz
bugzilla-2.18.1/
bugzilla-2.18.1/.cvsignore
bugzilla-2.18.1/1x1.gif
Affichage tronqué
bash$ cd bugzilla-2.18.1
bash$ cp ../bugzilla/localconfig* .
bash$ cp -r ../bugzilla/data .
bash$ cd ..
bash$ mv bugzilla bugzilla.old
bash$ mv bugzilla-2.18.1 bugzilla
bash$ cd bugzilla
bash$ ./checksetup.pl
Affichage omis
```

Avertissement

Si les commandes **cp** terminent toutes les deux par un point, ce qui est un détail important, cela indique à la console que le répertoire de destination est le répertoire de travail courant. De la même façon, le point au début de la commande **./checksetup.pl** est important et ne doit pas être omis.

Note

Vous devrez rétablir à la main toute modification locale que vous avez faite.

Exemple 3.3. Mise à niveau avec les correctifs

L'équipe de développement Bugzilla rend généralement disponible un correctif pour aller d'une révision donnée à la nouvelle. Vous pouvez aussi lire les release notes et utiliser les correctifs attachés aux bogues décrits mais il est plus simple de prendre le correctif publié car les correctifs sont quelques fois modifiés avant d'être intégré. Il est aussi théoriquement possible de parcourir la liste des bogues corrigés et de choisir les correctifs d'une révision qu'on veut appliquer mais ceci n'est pas recommandé non plus car on obtient un Bugzilla qui ne correspond a aucune version officielle. Ceci rend les mises à niveau futures plus compliqués.

```
bash$ cd /var/www/html/bugzilla
bash$ wget ftp://ftp.mozilla.org/pub/mozilla.org/webtools/bugzilla-2.18.0-to-2.18.1.diff.gz
Affichage omis
bash$ gunzip bugzilla-2.18.0-to-2.18.1.diff.gz
bash$ patch -p1 < bugzilla-2.18.0-to-2.18.1.diff
patching file checksetup.pl
patching file collectstats.pl
patching file globals.pl
```

Attention

N'oubliez pas que mettre à jour à partir d'un fichier correctif ne modifie pas les entrées de votre répertoire `CVS`. Cela peut rendre plus difficile une future mise à niveau par le CVS (Exemple 3.1, « Mise à niveau par le CVS » [p 50]).

Chapitre 4. Sécurité de Bugzilla

Table des matières

Bien que certains éléments de ce chapitre soient relatifs au système d'exploitation pour lequel Bugzilla fonctionne ou aux logiciels nécessaires à l'exécution de Bugzilla, tout est en rapport avec la protection de vos données. Ceci ne prétend pas être un guide complet pour sécuriser Linux, Apache, MySQL ou tout autre logiciel mentionné. Rien ne remplace une administration dynamique et la surveillance d'une machine. L'élément essentiel pour une bonne sécurité, vous la connaissez parfaitement : *c vous*.

Bien qu'en général les programmeurs s'efforcent d'écrire du code sécurisé, les accidents sont toujours possibles, et ils se produisent. La meilleure approche pour la sécurité est de toujours supposer que le programme sur lequel vous êtes en train de travailler n'est pas sécurisé à 100% et de restreindre son accès aux autres parties de votre machine autant que possible.

1. Le système d'exploitation

1.1. Ports TCP/IP

La norme TCP/IP établit plus de 65000 ports pour le trafic entrant et sortant. Parmi ceux-là, Bugzilla en a besoin précisément d'un pour fonctionner (selon les configurations et les options, on pourra en avoir besoin de 2, voire 3). Il faut vérifier votre serveur et vous assurer qu'il n'est pas à l'écoute sur des ports dont vous n'avez pas besoin. Il est également fortement recommandé que le serveur sur lequel se trouve Bugzilla, ainsi que toute autre machine que vous administrez, soit placé derrière quelque chose du genre coupe-feu.

1.2. Comptes utilisateur du système

Beaucoup de *démons* [p 126] tels que le `httpd` d'Apache ou le `mysqld` de MySQL s'exécutent aussi bien en tant que « root » qu'en tant que « nobody ». C'est encore pire sur les machines Windows où la majorité des *services* [p 127] sont exécutés en tant que « SYSTEME ». Bien que l'exécution en tant que « root » ou « SYSTEM » présente des problèmes de sécurité évidents, les problèmes que présente l'exécution de tout en tant que « nobody » peuvent ne pas être si évidents. En gros, si vous exécutez tous les démons en tant que « nobody » et que l'un d'entre eux se trouve compromis, il peut compromettre tous les autres démons exécutés en tant que « nobody »sur votre machine. Pour cette raison, il est conseillé de créer un compte utilisateur pour chaque démon.

Note

Il vous faudra configurer l'option `webservergroup` dans `localconfig` pour le groupe dans lequel votre serveur Web s'exécute. Cela permettra à `./checksetup.pl` de configurer les permissions des fichiers sur les systèmes Unix de manière à ce que rien ne soit en écriture libre.

1.3. Environnement d'exécution fermé

Si votre système le supporte, vous souhaitez peut être exécuter Bugzilla à l'intérieur d'un environnement d'exécution fermé `chroot`. Cette option fournit une sécurité sans précédent en interdisant à tout ce qui s'exécute dans cet environnement d'exécution fermé d'accéder à toute information lui étant extérieure. Si vous souhaitez utiliser cette option, veuillez consulter la documentation livrée avec votre système.

2. MySQL

2.1. Les comptes Système MySQL

Comme mentionné dans Section 1.2, « Comptes utilisateur du système » [p 53] , le démon MySQL devrait s'exécuter comme utilisateur unique non privilégié. Ne manquez pas de consulter les instructions de la documentation MySQL ou de la documentation livrée avec votre système.

2.2. Le super utilisateur et l'utilisateur anonyme de MySQL

Par défaut, MySQL est fourni avec un utilisateur « root » dont le mot de passe est vide et avec un utilisateur « anonymous », dont le mot de passe est également vide. Afin de protéger vos données, il faut demander un mot de passe à l'utilisateur « root » et l'utilisateur anonyme doit être désactivé.

Exemple 4.1. Affecter un mot de passe à l'utilisateur « root » de MySQL

```
bash$ mysql mysql
mysql> UPDATE user SET password = password('new_password') WHERE user = 'root';
mysql> FLUSH PRIVILEGES;
```

Exemple 4.2. Désactiver l'utilisateur « anonymous » de MySQL

```
bash$ mysql -u root -p mysql           ❶ Enter Password: new_password
mysql> DELETE FROM user WHERE user = '';
mysql> FLUSH PRIVILEGES;
```

❶ [p 54] Cette commande suppose que vous avez déjà effectué Exemple 4.1, « Affecter un mot de passe à l'utilisateur « root » de MySQL » [p 53] .

2.3. Accès au réseau

Si MySQL et votre serveur Web fonctionnent tous les deux sur la même machine et que vous n'avez pas d'autres raisons d'accéder à MySQL à distance, vous devriez désactiver l'accès au réseau. Cela, ainsi que ce qui est proposé dans Section 1.1, « Ports TCP/IP » [p 52] , vous permettra de protéger votre système de toutes vulnérabilités à distance de MySQL.

Exemple 4.3. Désactiver la gestion du réseau dans MySQL

Tapez simplement ce qui suit dans `/etc/my.conf`:

```
[myslqd]
# Prevent network access to MySQL.
skip-networking
```

3. Serveur Web

3.1. Désactivation des accès à distance pour les fichiers de configuration Bugzilla

Il y a beaucoup de fichiers placés dans la zone du répertoire Bugzilla qui ne doivent pas être accessibles depuis le web. À cause de la présentation actuelle de Bugzilla, la liste de ceux qui doivent être accessibles ou pas est plutôt compliquée. Une nouvelle méthode d'installation est actuellement en cours de réalisation; elle devrait résoudre cela en autorisant les fichiers qui ne doivent pas être accessibles depuis le web à être placés dans un répertoire à l'extérieur de la racine internet. Voir le bogue 44659 pour plus d'informations.

Astuce

À l'état brut, Bugzilla a la capacité de créer des fichiers *.htaccess* [p 123] qui imposent ces règles. Les instructions pour activer ces directives dans Apache sont disponibles dans Section 2.4.1, « httpd™ d'Apache » [p 19] .

- Dans le répertoire principal de Bugzilla, il faut :
  - bloquer : `*.pl`, `*localconfig*`, `runtests.sh`
  - Mais autoriser : `localconfig.js`, `localconfig.rdf`

- Dans `data`:
    - bloquer tout
    - mais autoriser : `duplicates.rdf`
- Dans `data/webdot`:
    - si vous utilisez un serveur distant webdot :
        - bloquer tout
        - mais autoriser `*.dot` seulement pour le serveur webdot distant
    - autrement, si vous utilisez un Graph Viz local :
        - bloquer tout
        - mais autoriser : `*.png`, `*.gif`, `*.jpg`, `*.map`
    - et si vous n'utilisez aucun fichier dot :
        - bloquer tout
- Dans `Bugzilla`:
    - bloquer tout
- Dans `template`:
    - bloquer tout

Surtout faites bien un test pour voir si les données qui ne doivent pas être accessibles à distance sont correctement bloquées. Le fichier localconfig qui contient votre mot de passe de base de données est particulièrement intéressant ici. Sachez également que beaucoup d'éditeurs créent des fichiers temporaires et de restauration dans le répertoire de travail et que ceux-ci ne doivent également pas être accessibles. Pour plus d'informations, voyez le bogue 186383 ou le Bugtraq ID 6501. Pour le test, précisez simplement à votre navigateur web où est le fichier; par exemple, pour tester une installation mozilla.org, on essaiera d'accéder à http://bugzilla.mozilla.org/localconfig. Vous devriez obtenir une erreur « 403 Forbidden ».

Astuce

N'oubliez pas de consulter, dans Section 2.4, « Web server » [p 19] , les instructions spécifiques au serveur web que vous utilisez.

3.2. Utilisation de `mod_throttle` pour éviter un déni de service

Note

Cette partie ne concerne que les gens qui ont choisi un serveur Apache. Il est peut-être possible de faire des choses similaires avec d'autres serveurs web. Consultez la documentation livrée avec votre serveur web pour voir si c'est le cas.

Il est possible pour un utilisateur, par erreur ou volontairement, d'accéder plusieurs fois de suite à la base de données, ce qui peut engendrer des vitesses d'accès très faibles pour les autres utilisateurs (en fait, c'est une attaque par *DOS* [p 124]). Si votre installation Bugzilla rencontre ce problème, vous pouvez installer le module Apache `mod_throttle` qui est capable de limiter le nombre de connexions par adresse IP. Vous pouvez télécharger ce module à http://www.snert.com/Software/mod_throttle/. Suivez les instructions pour l'insérer dans votre installation d'Apache. La commande qu'il vous faut est **ThrottleClientIP**. Veuillez lire la documentation du module pour plus d'informations.

4. Bugzilla

4.1. Empêcher les utilisateurs d'introduire du Javascript malveillant

Il est possible pour un utilisateur Bugzilla de profiter des ambiguïtés de codage du jeu de caractères pour injecter du HTML dans les commentaires Bugzilla. Celui-ci pourrait contenir des scripts malveillants. En raison de considérations liées à l'internationalisation, nous ne sommes pas en mesure d'intégrer par défaut les modifications de code proposées par les services d'assistance du CERT sur cette question. Si votre installation est destinée uniquement à un public anglophone, vous éviterez ce problème en effectuant la modification de Exemple 4.4, « Obliger Bugzilla à indiquer le codage utilisé (charset) » [p 56] .

Exemple 4.4. Obliger Bugzilla à indiquer le codage utilisé (charset)

Trouvez la ligne suivante dans `Bugzilla/CGI.pm` :

```
$self->charset('');
```

et remplacez la par :

```
$self->charset('ISO-8859-1');
```

Chapitre 5. Personnalisation de Bugzilla

Table des matières

1. Personnalisation des modèles

Les administrateurs peuvent configurer l'aspect et la convivialité de Bugzilla sans avoir à éditer des fichiers Perl ou à être confrontés au cauchemar des conflits massifs de fusion quand, plus tard, ils mettront à niveau avec une version plus récente.

L'utilisation de modèles rend également possible, pour la première fois, les versions de Bugzilla adaptées selon la région. Il est possible de faire en sorte que la langue de l'interface utilisateur de Bugzilla soit déterminée par le navigateur de l'utilisateur. Vous trouverez davantage d'informations dans Section 1.6, « Configurer Bugzilla pour détecter la langue de l'utilisateur » [p 61] .

1.1. Structure du répertoire de modèles

La structure de répertoires des modèles est composée d'un répertoire de haut niveau nommé `template`, qui contient un répertoire pour chaque région installée. Le niveau suivant définit la langue utilisée dans les modèles. Bugzilla est livré avec des modèles en anglais, le nom du répertoire est donc `en`, et nous parlerons de `template/en` dans toute la documentation. En dessous de ça, template/en est un répertoire `default` qui contient tous les modèles standard fournis avec Bugzilla.

> **Avertissement**
>
> Un répertoire `data/templates` existe aussi ; c'est là où Template Toolkit met les versions compilées à partir des répertoires par défaut ou personnalisés. N'éditez *pas* directement les fichiers dans ce répertoire, ou alors toutes vos modifications seront perdues la prochaine fois que Template Toolkit recompilera les modèles.

1.2. Choix d'une méthode de personnalisation

Si vous voulez éditer des modèles Bugzilla, la première choix à faire est de décider comment vous allez vous y prendre. Il y a deux possibilités et celle que vous utiliserez dépend principalement de la portée de vos modifications, et de la méthode que vous prévoyez d'utiliser pour mettre à niveau Bugzilla.

La première méthode pour personnaliser les modèles consiste à éditer directement ceux qui sont dans `template/en/default`. C'est probablement la meilleure méthode pour de petites modifications si vous comptez utiliser la méthode de mise à niveau par CVS, parce qu'alors, si vous exécutez un **cvs update**, toute correction dans un modèle s'intègrera automagiquement dans les versions que vous aurez mises à jour.

Note

Si vous utilisez cette méthode, et si des conflits CVS surviennent pendant une mise à jour, les modèles en conflit (et éventuellement d'autres parties de votre installation) ne fonctionneront pas tant qu'ils ne seront pas résolus.

L'autre méthode consiste à copier les modèles à modifier dans une arborescence de répertoires miroir sous `template/en/custom`. Les modèles de cette arborescence de répertoire prennent le pas sur tous les modèles de même nom et situés au même endroit dans le répertoire `default`.

Note

Le répertoire `custom` n'existe pas initialement et doit être créé si vous voulez l'utiliser.

C'est la technique que vous devez utiliser si vous utilisez la méthode de mise à niveau par réécriture, parce qu'autrement vos modifications seront perdues. Cette méthode est peut-être également la meilleure si vous utilisez la méthode de mise à niveau par CVS et que vous comptez faire des modifications majeures, car il est garanti que le contenu de ce répertoire ne sera pas touché au cours d'une mise à niveau, et vous pouvez alors décider soit de continuer en utilisant votre propre modèle, soit de faire l'effort d'intégrer à la main vos modifications dans les nouvelles versions.

Avec cette méthode, votre installation peut être perdue en cas de modifications incompatibles faites à l'interface des modèles. Si de telles modifications sont faites, elles seront détaillées dans la notice de la version, à condition que vous utilisiez une version stable de Bugzilla. Si vous utilisez du code instable, vous devrez vous occuper de ça vous-même, bien que, autant que possible, les modifications seront citées, avant qu'elles ne s'appliquent, dans la partie « À éviter » de la notice de la précédente version stable.

Note

Quelle que soit la méthode que vous avez choisie, il est recommandé d'exécuter **./checksetup.pl** après la création ou l'édition de tout modèle dans le répertoire `template/en/default`, et après l'édition de tout modèle dans le répertoire `custom`.

Avertissement

Il est *indispensable* d'exécuter **./checksetup.pl** après la création d'un nouveau modèle, dans le répertoire `custom`. Si vous ne le faites pas, cela entraînera un message d'erreur incompréhensible.

1.3. Méthode d'édition de modèles

Note

Si vous apportez des modifications aux modèles que vous souhaitez voir inclus dans le Bugzilla standard, il faut lire les parties correspondantes du guide des développeurs.

La syntaxe du langage de Template Toolkit dépasse le cadre de ce guide. Elle est assez facile à comprendre en regardant les modèles actuels ; vous pouvez également lire le manuel, disponible sur la page d'accueil de Template Toolkit.

Cependant, une chose à laquelle vous devriez apporter un soin particulier est la nécessité de filtrer correctement, par rapport au langage HTML, les données qui ont été transmises dans le modèle. Cela signifie que si les données sont susceptibles de contenir des caractères HTML spéciaux comme <, et que les données ne sont pas sensées être du HTML, elles doivent être converties sous la forme d'entité, c'est-à-dire < on utilise le filtre « html » de Template Toolkit pour faire cela. Si vous ne le faîtes pas, vous pouvez exposer votre installation à des attaques par scripts à l'intérieur du site.

Notez également que Bugzilla inclut de lui-même quelques filtres, qui ne sont pas dans le Template Toolkit standard. En particulier, le filtre « url_quote » peut convertir les caractères qui sont illégaux ou qui ont un sens particulier dans les URL, comme &, en une forme encodée, c'est-à-dire %26. En fait, celui-ci encode la plupart des caractères (mais pas les plus fréquents comme les lettres et les nombres et cætera), y compris les caractères HTML spéciaux, de sorte que, par la suite, vous n'avez jamais besoin de filtrer par rapport au langage HTML.

Editer des modèles est une bonne façon de faire les « champs personnalisés du pauvre ». Par exemple, si vous n'utilisez pas le Tableau Blanc mais que vous souhaitez disposer d'une boîte d'entrée de texte de forme libre pour le « Identifiant de Création », vous pouvez vous contenter d'éditer les modèles pour changer le nom des champs. Ca s'appellera toujours status_whiteboard en interne, mais vos utilisateurs n'ont pas besoin de le savoir.

1.4. Formats et type des modèles

Certains CGI sont capables d'utiliser plus d'un modèle. Par exemple, `buglist.cgi` peut générer des listes de bogues en RDF ou en 2 formes différentes de HTML (complexe et simple). Le mécanisme qui fournit ce dispositif est extensible.

Bugzilla peut supporter différents types de sorties, qui peuvent eux aussi avoir des formats multiples. Pour demander un certain type, vous pouvez joindre le &ctype=<contenttype> (tel que rdf ou html) au lien `<cginame>.cgi`. Si vous souhaitez rechercher un certain format, vous pouvez utiliser le &format=<format> (comme simple ou complex) dans l'URL.

Pour savoir si un CGI supporte plusieurs formats de sortie et plusieurs types, effectuez un grep sur le CGI pour « GetFormat ». S'il n'y en a pas, l'ajout de support de plusieurs formats/types n'est pas trop dur ; regardez comment c'est fait dans d'autres CGI, i.e. config.cgi.

Pour faire un nouveau modèle de format pour un CGI qui supporte ça, ouvrez un modèle existant pour ce CGI et prenez note du commentaire INTERFACE (s'il est présent). Ce commentaire définit quelles variables sont passées dans ce modèle. S'il n'y en a pas, j'ai bien peur que vous deviez lire le modèle et le code pour voir ce que vous avez comme informations.

Vous pouvez écrire votre modèle dans n'importe quel style de balisage ou de texte qui convient.

Il vous faut maintenant choisir le type de contenu dans lequel vous voulez qu'il soit utilisé. Les types de contenu sont définis dans le fichier `Bugzilla/Constants.pm` dans la constante `contenttypes`. Si votre type de contenu ne s'y trouve pas, ajoutez le. Mémorisez l'étiquette de trois ou quatre lettres affectée à votre type de contenu. Cette étiquette fera partie du nom de fichier du modèle.

Note

Après l'ajout ou la modification d'un type de contenu, il convient d'éditer `Bugzilla/Constants.pm` afin de répercuter les modifications. En outre, le fichier doit être conservé à jour après une mise à jour si des types de contenu ont été personnalisés dans le passé.

Enregistrez le modèle sous `<stubname>-<formatname>.<contenttypetag>.tmpl`. Testez le modèle en appelant le CGI par `<cginame>.cgi?format=<formatname>&ctype=<type>`.

1.5. Modèles particuliers

Il y a quelques modèles qui pourraient vous intéresser tout particulièrement pour personnaliser votre installation.

index.html.tmpl : c'est la page d'accueil de Bugzilla.

global/header.html.tmpl : définit l'en-tête qui apparaît sur toutes les pages de Bugzilla. L'en-tête inclut la bannière, qui est ce qui apparaît aux utilisateurs et qui est probablement ce que vous voulez éditer plutôt que l'en-tête. Cependant, l'en-tête inclut également la section HTML HEAD, vous pourriez donc par exemple ajouter une feuille de style ou une balise META en éditant l'en-tête.

global/banner.html.tmpl : contient la « bannière », la partie de l'en-tête qui apparaît en haut de toutes les pages de Bugzilla. La bannière par défaut est assez sommaire, vous voudrez donc probablement la personnaliser pour donner à votre installation son propre aspect et sa propre convivialité. Il est recommandé de conserver le numéro de version Bugzilla sous une forme ou une autre pour permettre de déterminer la version que vous utilisez, et que les utilisateurs sachent quelle documentation ils doivent lire.

global/footer.html.tmpl : définit le pied de page qui apparaît sur toutes les pages de Bugzilla. En l'éditant, vous avez un autre moyen de donner rapidement son propre aspect et sa propre convivialité à votre installation Bugzilla.

global/variables.none.tmpl : définit une liste de termes qui peuvent être modifiés afin « d'estampiller » l'instance Bugzilla. De cette manière, les termes tels que « bugs » peuvent être remplacés par « problèmes » dans toute l'installation Bugzilla. Le nom « Bugzilla » et d'autres mots peuvent être tout aussi bien personnalisés.

list/table.html.tmpl : ce modèle gère l'apparence des listes de bogues créées par Bugzilla. En éditant ce modèle, on peut modifier la largeur et le titre des colonnes une par une, la taille maximale de l'affichage pour chaque entrée, et le comportement de l'enveloppe des entrées longues. Pour les longues listes de bogues, Bugzilla insère un « break » tous les 100 bogues par défaut ; ce comportement est également géré par ce modèle, et c'est là que cette valeur pourra être modifié.

bug/create/user-message.html.tmpl : message qui apparaît près du haut de la page de soumission de bogues. En le modifiant, vous pouvez dire à vos utilisateurs comment soumettre des bogues.

bug/process/midair.html.tmpl : ceci est le modèle utilisé si deux personnes modifient un bug simultanément. La deuxième personne a soumettre une modification verra apparaître cette page indiquant la modification apportée par la première personne et demandant s'ils veulent écraser cette modification ou revenir sur le bogue. Le titre et l'en-tête par défaut de cette page est « Collision en plein vol détectée ! » Si vous travaillez dans l’industrie aero-nautique ou un autre milieu ou ceci peut géner quelqu’un (oui, on nous a rapporté des histoires vraies à ce sujet), vous voudrez peut-être changer cette phrase.

bug/create/create.html.tmpl et **bug/create/comment.txt.tmpl** : vous n’avez peut-être pas envie de vous casser la tête à créer des zones personnalisées dans Bugzilla, mais vous voulez être sûr que chaque rapport de bogue contienne un nombre d’informations importantes pour lesquelles il n’y a pas de zone spéciale. Le système d’entrée de bogues a été conçu sous une forme extensible pour vous permettre d’ajouter arbitrairement des gadgets HTML, tels que listes déroulantes et zones de texte, à la page d’entrée de bogues et de faire apparaître leurs valeurs formatées dans la description initiale. Un champ caché qui indique le format doit être ajouté à l’intérieur du formulaire afin de rendre le modèle fonctionnel. Sa valeur doit être le suffixe de nom du fichier du modèle. Par exemple, si le fichier s’appelle `create-cust.html.tmpl`, alors

```
<input type="hidden" name="format" value="cust">
```

doit être utilisé à l’intérieur du formulaire.

Un exemple en est le formulaire de soumission de bogues de mozilla.org. Le code pour cela est livré avec la distribution Bugzilla comme exemple à copier par vous. Vous le trouverez dans les fichiers `create-guided.html.tmpl` et `comment-guided.html.tmpl`.

Donc pour utiliser ce dispositif, créez un modèle personnalisé pour `enter_bug.cgi`. Le modèle par défaut, sur lequel vous pouvez vous baser, est `custom/bug/create/create.html.tmpl`. Nommez le `create-<formatname>.html.tmpl` et ajoutez y des gadgets pour chaque information que vous aimeriez voir collectée : un numéro de création, l’ensemble des étapes pour le reproduire.

Puis, créez un modèle comme `custom/bug/create/comment.txt.tmpl`, et nommez le `comment-<formatname>.txt.tmpl`. Ce modèle doit renvoyer aux champs du formulaire que vous avez créé en utilisant la syntaxe `[% form.<fieldname> %]`. Quand un rapport de bogue est soumis, le commentaire initial joint au rapport de bogue sera formaté d’après la disposition de ce modèle.

Par exemple, si votre modèle enter_bug possédait un champ

```
<input type="text" name="buildid" size="30">
```

et que votre comment.txt.tmpl avait

```
BuildID : [% form.buildid %]
```

alors

```
BuildID : 20020303
```

apparaîtrait dans le commentaire initial.

1.6. Configurer Bugzilla pour détecter la langue de l’utilisateur

Bugzilla prend en compte l’en-tête HTTP Accept: de l’utilisateur. Vous pouvez installer des modèles dans d’autres langues, et Bugzilla retiendra le plus approprié d’après l’ordre de priorités que vous avez défini. Beaucoup de modèles de langue peuvent être récupérés à partir de http://www.bugzilla.org/download.html#localizations. Les instructions pour intégrer de nouvelles langues sont aussi disponibles à cet endroit.

Après avoir décompressé les localisations (ou créé les vôtres) dans le répertoire `BUGZILLA_ROOT/template`, vous devez mettre à jour le paramètre `languages` pour contenir toutes les adaptations que vous aimeriez laisser. Vous pouvez également, si vous le souhaitez, modifier le paramètre `defaultlanguage` en remplaçant « en » par autre chose si vous ne voulez pas l'anglais comme langue par défaut.

2. Crochets de modèles

Avertissement

Les crochets Template ont besoin du Template Toolkit version 2.12 ou plus, ou l'application d'un correctif. Voir le bogue 239112 pour plus d'informations.

Les crochets Template permettent aux extensions de Bugzilla d'insérer du code dans les modèles Bugzilla standard sans modifier les fichiers du modèle eux-mêmes. Le mécanisme de crochets définit une API cohérente pour étendre les modèles standards de manière à séparer proprement le code standard du code d'extension. Les crochets réduisent les conflits de fusion et facilitent l'écriture d'extensions qui fonctionnent pour de multiples versions de Bugzilla, ce qui facilite la mise à jour d'une installation Bugzilla avec des extensions installées.

Un crochet de modèle est juste un emplacement nommé dans un fichier de modèle standard où les fichiers du modèle d'extension pour ce crochet sont traités. Chaque crochet a un répertoire correspondant dans l'arborescence de Bugzilla. Accrocher un modèle d'extension à un crochet est aussi simple que de mettre le fichier d'extension dans le répertoire du crochet. Les crochets eux-mêmes peuvent être ajoutés dans un modèle standard sur demande des auteurs de l'extension.

Pour utiliser des crochets afin d'étendre un modèle Bugzilla, assurez vous d'abord qu'il y a un crochet à l'emplacement approprié dans le modèle que vous voulez étendre. Les crochets apparaissent dans les modèles standard Bugzilla en tant que simple directive dans le format `[% Hook.process("nom") %]`, où `nom` est (dans ce modèle) l'unique nom du crochet.

Si vous n'êtes pas sûr quel modèle vous voulez étendre ou que vous voulez juste passer en revue les crochets disponibles, vous pouvez soit utiliser votre outil de recherche multi-fichiers préféré (i.é. **grep**) pour chercher parmi les modèles standards les occurrences de `Hook.process`, soit passer en revue l'arborescence Bugzilla dans `BUGZILLA_ROOT/template/en/extension/hook/`, qui contient un répertoire pour chaque crochets à l'emplacement suivant :

```
BUGZILLA_ROOT/template/en/extension/hook/CHEMIN_VERS_MODÈLE_STANDARD/NOM_MODÈLE_STANDARD/NOM_CROCHET/
```

S'il n y a pas de crochets à l'emplacement approprié dans le modèle Bugzilla que vous voulez étendre, faites un rapport de bogue pour en demander un, en spécifiant : un lien vers les informations à propos de votre extension, s'il y en a. S'il n'y a pas de crochets à l'endroit approprié dans le modèle Bugzilla que vous souhaitez étendre, rapportez un bug pour en demander un, en précisant :

le modèle pour lequel vous demandez un crochet

où vous aimeriez que le crochet soit placé dans le modèle (numéro de ligne/position pour la dernière version de modèle dans le CVS ou description du but du crochet) ;

le but de ce crochet ;

un lien vers les informations à propos de votre extension, s'il y en a.

Les responsables de l'examen des bogues de Bugzilla examineront sans délai chaque demande de crochet, nommeront le crochet et l'ajouteront au modèle, vérifieront les nouvelles versions du modèle dans le CVS, et créeront le répertoire correspondant dans `BUGZILLA_ROOT/template/en/extension/hook/`.

Vous pouvez en option joindre un correctif au bogue qui implémente le crochet et l'enregistrer vous-même après avoir reçu l'accord d'un vérificateur Bugzilla. Les développeurs peuvent suggérer des modifications de l'emplacement d'un crochet en se basant sur l'analyse de vos besoins ou pour que le crochet puisse satisfaire les besoins d'extensions multiples mais la procédure permettant d'approuver et d'enregistrer les crochets n'est pas aussi rigoureuse que la procédure pour les modifications générales de Bugzilla, et toute extension, qu'elle soit sortie ou toujours en développement, peut avoir des crochets ajoutés pour répondre à leurs besoins.

Après vous être assuré que le crochet dont vous avez besoin existe (ou après l'avoir récupéré si ce n'est pas le cas), ajoutez votre modèle d'extension au répertoire de l'arborescence correspondant au crochet.

Oui, c'est tout ! Maintenant, quand le modèle standard contenant le crochet sera traité, votre modèle d'extension sera traité à l'endroit où le crochet apparaît.

Par exemple, disons que vous avez une extension nommée Projman qui ajoute les possibilités de gestion de projet à Bugzilla. Projman a une interface d'administration `edit-projects.cgi` et vous voulez ajouter un lien vers elle dans la barre de navigation en bas de toutes les pages Bugzilla pour ceux parmi les utilisateurs qui sont autorisés à administrer les projets.

La barre de navigation est générée par le fichier `useful-links.html.tmpl` du modèle, qui est placé dans le sous-répertoire `global/` sur le chemin `BUGZILLA_ROOT/template/en/default/` d'un modèle Bugzilla standard. Si vous regardez dans `useful-links.html.tmpl`, vous trouverez le crochet suivant à la fin de la liste des liens d'administration Bugzilla standard :

```
...
    [% ', <a href="editkeywords.cgi">keywords</a>'
                                              IF user.groups.editkeywords %]
    [% Hook.process("edit") %]
...
```

Le répertoire correspondant pour ce crochet est `BUGZILLA_ROOT/template/en/extension/hook/global/useful-links.html.tmpl/edit/`.

Vous mettez un modèle nommé `projman-edit-projects.html.tmpl` dans ce répertoire avec le contenu suivant :

```
...[% ', <a href="edit-projects.cgi">projects</a>' IF user.groups.projman_admins %]
```

Voilà ! Le lien apparaît maintenant après les autres liens d'administration dans la barre de navigation pour les utilisateurs du groupe `projman_admins`.

Astuces :

- Vous aurez peut être envie de préfixer vos noms de modèle d'extension avec le nom de votre extension, par exemple `projman-foo.html.tmpl` pour qu'ils n'entrent pas en conflit avec les noms des modèles installés par d'autres extensions.

- Si votre extension englobent complètement de nouveaux modèles en plus des extensions des modèles standards, il faut qu'elle installe ces nouveaux modèles dans un sous-répertoire du répertoire `BUGZILLA_ROOT/template/en/extension/` spécifique aux extensions. Le répertoire `extension/`, tout comme les répertoires `default/` and `custom/`, fait partie du chemin de recherche de modèle, et donc en mettant des modèles à cet endroit là, le processeur de modèles pourra les trouver.

  Le processeur de modèles cherche les modèles d'abord dans le répertoire `custom/` (modèles ajoutés par une installation spécifique), puis dans le répertoire `extension/` (modèles ajoutés par des extensions), et finalement dans le répertoire `default/` (les modèles Bugzilla standard). Donc les modèles d'extension peuvent prendre le pas sur des modèles standard mais les modèles à installation particulière prennent le pas sur tous les deux.

  Notez que l'annulation de modèles standard vous donne un grand pouvoir mais rend également la mise à jour d'une installation plus difficile. Comme avec un modèle personnalisé, nous vous recommandons d'utiliser cette fonctionnalité avec parcimonie et seulement quand c'est absolument nécessaire.

- Les personnalisateurs d'installation peuvent aussi profiter des crochets quand ils ajoutent du code à un modèle Bugzilla. Pour ce faire, créez des répertoires dans `BUGZILLA_ROOT/template/en/custom/hook/` équivalents aux répertoires dans `BUGZILLA_ROOT/template/en/extension/hook/` pour les crochets que vous voulez utiliser puis placez vos modèles de personnalisation dans ces répertoires.

  Evidemment cette méthode de personnalisation de Bugzilla vous laisse seulement ajouter du code à vos modèles standard ; vous ne pouvez pas modifier le code existant. Cependant, pour ces personnalisations qui ne font qu'ajouter du code, cette méthode peut réduire les conflits quand la fusion est modifiée, rendant la mise à jour de votre installation Bugzilla personnalisée plus facile.

3. Personnalisation : Qui peut faire quoi ?

Avertissement

Cette fonctionnalité devrait être considérée comme expérimentale ; le code Bugzilla que vous allez changer n'est pas stable et est susceptible de changer ou d'être déplacé suivant les versions. Il faut savoir que si vous effectuez des modifications comme celles indiquées ci-dessous, il se peut que vous ayez à les refaire ou à les porter en cas de changement interne dans une nouvelle version de Bugzilla et que vous passiez à cette version.

Les entreprises ont souvent des règles dictant quels employés, ou quelles catégories d'employés, sont autorisés à changer certaines parties dans le système de bogues. Par exemple, seul le Contact Assurance Qualité assigné au bogue devrait avoir la permission de donner à un bogue le statut VERIFY. Bugzilla a été conçu pour vous faciliter l'écriture de vos propres règles personnalisées afin de définir qui a la permission d'effectuer telle ou telle sorte de changement de valeur.

Pour une flexibilité maximale, ce type de personnalisation implique l'édition du code Perl de Bugzilla. Ceci donne à l'administrateur tout contrôle sur qui exactement est autorisé à faire quoi. La fonction appropriée s'appelle `CheckCanChangeField()` et se situe dans `process_bug.cgi` dans votre répertoire de Bugzilla. Si vous ouvrez ce fichier et que vous cherchez la chaîne « sub CheckCanChangeField », vous la trouverez.

Cette fonction est accompagnée d'un commentaire détaillé afin de vous permettre de comprendre exactement comment elle fonctionne, et vous donner une idée sur la manière d'y effectuer des changements. Certaines sections sont marquées comme ne devant pas être changées : il s'agit de la partie « tuyauterie » qui fait marcher tout le reste. Entre ces sections, vous pourrez trouver des petits bouts de code comme :

```
# Permettre au propriétaire de tout changer.
if ($ownerid eq $whoid) {
    return 1;
}
```

Ce que fait ce bout de code est assez évident.

Alors, comment s'y prend-on pour changer cette fonction ? Et bien, des changements simples peuvent êtres effectués juste en supprimant des morceaux ; par exemple, si vous vouliez empêcher tout utilisateur de rajouter un commentaire à un bogue, supprimez simplement les lignes marquées « Autoriser toute personne à changer les commentaires ». Si vous ne voulez pas que le rapporteur du bogue ait des droits particuliers sur les bogues qu'elle a soumis, supprimez simplement toute la section qui le concerne.

Les personnalisations plus complexes ne sont pas beaucoup plus difficiles. En gros, vous ajoutez un test au bon endroit dans la fonction, c'est-à-dire après que toutes les variables que vous utilisez aient été initialisées. Ainsi, ne touchez pas à la variable $ownerid avant d'avoir obtenu $ownerid depuis la base de données. Vous pouvez soit ajouter un test positif, qui renvoie 1 (permission accordée) si certaines conditions sont vraies ou un test négatif, qui retourne 0 (permission refusée). Par exemple :

```
if ($field eq "qacontact") {
    if (Bugzilla->user->groups("quality_assurance")) {
        return 1;
    }
    else {
        return 0;
    }
}
```

Cela fait que seuls les utilisateurs du groupe « quality_assurance » peuvent changer le champ Contact QA d'un bug.

Un exemple plus bizarre :

```
if (($field eq "priority") &&
    (Bugzilla->user->email =~ /.*\@example\.com$/))
{
    if ($oldvalue eq "P1") {
        return 1;
    }
    else {
        return 0;
    }
}
```

Cela fait que si l'utilisateur essaie de changer le champ priorité et que son adresse électronique est @example.com, il n'y parviendra qu'à la condition que la précédente valeur du champ ait été « P1 ». Pas très utile, mais démonstratif.

Avertissement

Si vous modifiez `process_bug.cgi` de quelque manière que ce soit, ne changez pas le code délimité par les blocs DO_NOT_CHANGE. Ceci pourrait compromettre la sécurité ou causer l'arrêt complet du fonctionnement de votre installation.

Pour une liste des noms de champs possibles, consultez la liste appelée `@::logcolumns` dans le fichier `data/versioncache`. Si vous avez besoin d'aide pour écrire des règles personnalisées pour votre entreprise, demandez dans le groupe de discussion.

4. Modifier votre système en fonctionnement

Bugzilla optimise les opérations de consultation de la base de données en stockant toute information relativement statique dans le fichier `versioncache`, situé dans le sous-répertoire `data/` dans votre répertoire d'installation.

Si vous effectuez un changement dans les données structurelles de votre base de données (dans la table des versions par exemple), ou dans les « constantes » encodées dans `defparams.pl`, vous devrez supprimer du répertoire de données le contenu mis en cache (en effectuant un **rm data/versioncache**) sinon vos changements n'apparaîtront pas.

Comme `versioncache` est regénéré automatiquement dès lors qu'il existe depuis plus d'une heure, Bugzilla finira par remarquer vos changements de lui-même, mais généralement on préfère qu'il en tienne compte immédiatement, afin que vous puissiez faire des tests.

5. Introduction à la base de données MySQL de Bugzilla

Ces informations sont tirées de mon expérience personnelle. J'ai dû apprendre comment Bugzilla organise la base de données à cause des pinaillages d'utilisateurs demandant de minuscules changements dans le choix des termes plutôt que d'attendre que les gens se rééduquent par eux-même ou comprennent comment utiliser nos procédures. Ça craint mais vous y passerez peut-être aussi, donc apprenez comment fonctionne le schéma pour pouvoir vous débrouiller quand ça arrivera.

Donc, vous en êtes là avec votre installation toute neuve de Bugzilla. Vous avez installé MySQL, Apache tourne bien, Perl DBI et DBD communiquent parfaitement avec la base. Peut-être avez-vous même rentré quelques bogues de test pour vérifier que le courrier électronique fonctionne bien ; les gens semblent être avertis des nouveaux bogues ou modifications et vous pouvez créer ou éditer des bogues comme bon vous semble. Vous avez peut-être rencontré des problèmes lors de la configuration d'une passerelle de connexion destinée à permettre aux gens de soumettre des bogues à votre base de données par courrier électronique, vous avez demandé à des personnes de tester votre installation et reçu des retours enthousiastes de la part de vos bêta-testeurs.

Quelle est la suite au programme ? Esquisser une stratégie de formation de votre équipe de développement et leur demander de se précipiter sur le nouvel outil auquel vous avez consacré des heures de travail.

Votre première session de formation débute très bien ! Votre auditoire captivé semble émerveillé par l'efficacité intrinsèque de cette chose qui s'appelle « Bugzilla ». Vous êtes complètement absorbé par la description des caractéristiques pratiques : comment les gens peuvent sauvegarder leurs requêtes favorites dans la base de données, les mettre en tête ou en pied de leur page, personnaliser leur mise en page, générer des rapports, suivre le statut avec une efficacité plus grande que jamais, atteindre le sommet d'un gratte-ciel d'un seul bond, sauver Jane des griffes de la Mort personnifiée !

Mais la Mort personnifiée prend la parole : une petite voix, venue d'un coin sombre de la salle de conférence. La voix sort de l'ombre : « J'ai une remarque au sujet de l'utilisation du mot `verified`. »

La salle, jusque là remplie d'un joyeux babil, plonge dans un silence religieux tandis que la Mort personnifiée (plus connue sous le nom de Vice-Président du Département Génie Logiciel) poursuit. « Vous voyez, cela fait deux ans que nous utilisons le mot "verified" pour indiquer le fait qu'un développeur ou un ingénieur qualité a confirmé qu'un bogue était valide. Je ne veux pas perdre deux ans de formation à un nouveau logiciel. Vous devez remplacer le statut `verified` d'un bogue par "approved" dès que possible. Pour éviter toute confusion, bien sûr. »

Oh non ! La peur vous frappe de plein fouet alors que vous balbultiez « oui, oui, je pense que ça ne sera pas un problème, » vous examinez les changements avec la Mort personnifiée et continuez à baragouiner, « non, ça ne représente pas une grosse modification. Je veux dire, nous avons le code source, n'est-ce pas ? Vous savez bien, "Utiliser le code source tu dois, Luke" et tout ça... aucun problème, » pendant tout ce temps vous frémissez intérieurement comme une méduse échouée qui fait des gargouillis, des glougloutis et qui rôtit sur le sable brûlant d'une dune jamaïquaine...

Ainsi commence votre aventure au cœur de Bugzilla. Vous êtes condamné à faire connaissance avec les champs enum() fixes, les colonnes de varchar et les définitions des tinyint. Prêt pour l'aventure !

5.1. Les fondamentaux de la base de données de Bugzilla

Si, comme je l'étais, vous êtes totalement ignorant des mécanismes internes de MySQL à ce stade, et si vous n'étiez pas sous les ordres du Vice-Président, vous vous moqueriez de la différence qui existe entre un champ « bigint » et un « tinyint » dans MySQL. Je vous recommande de consulter la documentation MySQL en ligne. Vous trouverez ci-dessous les rudiments nécessaires pour comprendre la base de données de Bugzilla. Consultez la documentation MySQL ci-dessus pour plus d'informations.

1. Pour vous connecter à votre base de données :

   `bash#` **mysql** *-u root*

Si cela fonctionnne sans demander de mot de passe, *honte sur vous* ! Vous auriez dû mettre en place un verrouillage de sécurité comme cela était expliqué dans les instructions d'installation. Vous pouvez trouver plus d'informations concernant le verrouillage de votre base de données dans la FAQ de Bugzilla de ce répertoire (dans la partie « Sécurité »), ou des généralités plus solides sur la sécurité dans la documentation consultable de MySQL.

2. Vous devez maintenant voir une invite de commande qui ressemble à celle-ci :

   `mysql>`

   À l'invite, si « bugs » est le nom que vous avez choisi dans le fichier `localconfig` pour votre base de données Bugzilla, tapez :

   `mysql` **use bugs;**

5.1.1. Les tables de la base de données de Bugzilla

Imaginez votre base de données MySQL comme une série de feuilles de calcul d'un tableur, et vous ne serez pas loin de la vérité. Si vous tapez cette commande :

`mysql>` **show tables from bugs;**

vous pourrez voir le nom de toutes les « feuilles de calcul » (tables) dans votre base de données.

À partir de la commande tapée ci-dessus, vous devriez obtenir une réponse qui ressemble à celle-ci :

```
+-------------------+
| Tables in bugs    |
+-------------------+
| attachments       |
| bugs              |
| bugs_activity     |
| cc                |
| components        |
| dependencies      |
| fielddefs         |
| groups            |
| keyworddefs       |
| keywords          |
| logincookies      |
| longdescs         |
| milestones        |
| namedqueries      |
| products          |
| profiles          |
| profiles_activity |
| tokens            |
| versions          |
| votes             |
| watch             |
+-------------------+
```

Voici un aperçu de ce que fait chaque table. La plupart des colonnes de chaque table possèdent un nom descriptif qui permet de comprendre relativement rapidement leur rôle.

attachments

cette table contient tous les fichiers joints aux bogues. Cette table tend à être la plus grande mais généralement aussi celle qui comporte le moins d'entrées car les fichiers joints sont (relativement) gros.

bugs

c'est le cœur de votre système. La table bugs contient la plupart des informations concernant les bogues à l'exception des informations stockées dans les autres tables.

bugs_activity

conserve les informations concernant les changements apportés aux bogues — un fichier d'historique.

cc

cette petite table conserve simplement toutes les informations de copie carbone (CC) des bogues dont le champ CC est renseigné. Notez que, comme la plupart des autres tables dans Bugzilla, il n'est pas fait référence aux utilisateurs par le biais de leur nom d'utilisateur, mais par leur seul identifiant utilisateur, stocké dans la table profiles comme clé primaire.

components

stocke les programmes et les composants (ou les produits et les composants dans le nouveau langage de Bugzilla) de Bugzilla. Curieusement, le champ « program » (product) est le nom complet du produit et non un autre identifiant unique, comme bug_id et user_id le sont ailleurs dans la base de données.

dependencies

contient les données sur ces supers arbres de dépendance.

fielddefs

table pratique qui définit les autres tables. Par exemple lorsque vous validez un formulaire qui change la valeur de « AssignedTo », cette table permet le transfert de l'information vers le champ actuel dont le nom est « assigned_to » pour une entrée de MySQL.

groups

définit les masques binaires pour les groupes. Un masque binaire est un nombre qui peut identifier de manière unique les appartenances à un groupe. Par exemple, disons que le groupe qui a le droit d'apporter quelques petites modifications aux paramètres est affecté d'une valeur « 1 », le groupe qui a la droit d'éditer les utilisateurs est affecté d'une valeur « 2 » et le groupe qui peut créer de nouveaux groupes est affecté d'un masque binaire de « 4 ». Simplement en combinant les masques binaires des groupes (tout comme la commande chmod d'UNIX,) vous pouvez identifier un utilisateur qui a le droit de modifier légèrement les paramètres et de créer des groupes mais pas d'éditer les utilisateurs, en lui affectant un masque binaire de « 5 », ou bien un utilisateur ayant le droit d'éditer des utilisateurs et de créer des groupes, mais pas de modifier les paramètres, en lui donnant un masque binaire de « 6 ». Simple, hein ?

Si vous n’avez pas compris, essayer en tapant ceci à l’invite de commande de mysql :

```
mysql> select * from groups;
```

Vous verrez la liste, c’est plus compréhensible de cette manière.

keyworddefs

définition des mots-clés utilisés.

keywords

à l’inverse de ce que vous pensez, cette table dit quels mots-clés sont associés avec quels identifiants de bogues.

logincookies

cette table stocke tous les cookies de connexion qui vous sont assignés pour chaque machine à partir de laquelle vous vous êtes connecté à Bugzilla. Curieusement, elle ne gère pas le nettoyage ; je retrouve des cookies que je n’ai pas utilisés depuis des mois. Cependant, comme Bugzilla ne donne pas de date d’expiration aux cookies, cela se comprend.

longdescs

la substance de Bugzilla : voici où sont stockés tous les commentaires des utilisateurs ! Vous disposez uniquement de 2^24 octets par commentaire (il s’agit d’un champ de type mediumtext), donc soyez concis ; cela représente seulement l’espace que l’Ancien Testament prendrait (16 mégaoctets, non compressé). Tout commentaire reçoit comme clé le bug_id du bogue auquel il est lié de telle manière que l’ordre soit nécessairement chronologique car les commentaires sont retournés dans l’ordre où ils ont été reçus.

milestones

c’est une bonne chose que les jalons soient associés à un produit particulier dans cette table mais Bugzilla ne supporte pas encore les jalons qui diffèrent en fonction des produits via l’interface de configuration standard.

namedqueries

c’est ici que chacun stocke ses « requêtes personnelles ». Caractéristique très sympa ; avec cela, plus besoin de se casser la tête à mettre un signet sur chaque requète sympa que vous élaborez.

products

indique les produits dont vous disposez, si les nouveaux ajouts de bogues sont permis pour ce produit, sur quel jalon vous travaillez avec ce produit, les votes, etc. Ce sera bien lorsque la table components contiendra les mêmes attributs, de telle façon que l’on puisse bloquer l’ajout de nouveaux bogues pour un composant particulier sans bloquer un produit entier...

profiles

ah, alors vous vous demandiez où vos précieuses informations d'utilisateur étaient gardées ? C'est ici ! Avec les mots de passe en clair pour tout le monde ! (mais chuut... n'en parlez pas à vos utilisateurs !)

profiles_activity

vous avez besoin de savoir qui a fait quoi à quel profil utilisateur ? C'est ici que vous le saurez, il s'agit d'un historique assez complet.

versions

informations sur la version de chaque produit.

votes

qui a voté pour quoi et quand.

watch

qui (relativement au userid) consulte les bogues de qui (relativement à leur userid).

5.1.1.1. Les détails

Ah, alors vous vous demandez que faire des informations ci-dessus ? À l'invite de commande vous pouvez visualiser les informations sur n'importe quelle colonne d'une table grâce à cette commande (où « table » est le nom de la table que vous souhaitez voir) :

```
mysql> show columns from table;
```

Vous pouvez aussi voir toutes les données contenues dans une table avec cette commande :

```
mysql> select * from table;
```

Note

il est fortement déconseillé de le faire sur, par exemple, la table « bugs » si vous avez 50 000 bogues. Vous pourriez rester planté là un bon moment avant de faire un ctrl-c ou à attendre que les 50 000 bogues s'affichent à l'écran.

Vous pouvez limiter un peu l'affichage ci-dessus avec cette commande où « colonne » est le nom de la colonne à laquelle vous voulez restreindre l'affichage d'information :

```
mysql> select * from table where (column = "some info");
```

— ou l'inverse de cela

```
mysql> select * from table where (column != "some info");
```

Reprenons l'exemple de l'introduction et supposons que vous ayez besoin de changer le mot « verified » par « approved » dans le champ de résolution. Nous savons depuis la partie précédente que la résolution doit se trouver dans la table « bugs ». Notez que nous devrons changer un peu le code perl en plus de la modification de la base de données mais je ne vais pas aborder cela dans ce document. Vérifions que l'information est bien stockée dans la table « bugs » :

```
mysql> show columns from bugs
(résultat bien trop long, on l'a raccourci)
| bug_status| enum('UNCONFIRMED','NEW','ASSIGNED','REOPENED','RESOLVED','VERIFIED','CLOSED')||MUL | UNCONFIRMED||
```

Désolé de cette longue ligne. Nous voyons à partir du résultat que la colonne « bug status » est de type « enum field », ce qui est une particularité de MySQL où un champ de type chaîne de caractères ne peut prendre que certaines valeur en entrée. Bien que je trouve cela vraiment sympa, il ne s'agit pas de SQL standard. Néanmoins, nous avons besoin d'ajouter « APPROVED » dans les valeurs possibles du champ « enum » en modifiant la table « bugs ».

```
mysql> ALTER table bugs CHANGE bug_status bug_status
    -> enum("UNCONFIRMED", "NEW", "ASSIGNED", "REOPENED", "RESOLVED",
    -> "VERIFIED", "APPROVED", "CLOSED") not null;
```

(notez que vous pouvez entrer trois lignes ou plus ; tout ce que vous placerez avant le point virgule sera compris comme une seule expression)

Maintenant si vous faites cela :

```
mysql> show columns from bugs;
```

vous verrez que le champ bug_status dispose d'un « APPROVED » supplémentaire dans « enum » ! Une autre chose sympa serait que ce changement soit aussi propagé jusqu'à votre page de requête ; vous pouvez effectuer une requête au moyen du nouveau statut. Mais comment cela se propage-t-il dans la réalité des choses ?

Il semble que vous deviez retourner chercher les instances du mot « verified » dans le code perl de Bugzilla ; partout où vous trouvez « verified », remplacez le par « approved » et voilà, ça roule (assurez-vous que la recherche n'est pas sensible à la casse). Bien que vous puissiez effectuer des requêtes grâce au champ « enum », vous ne pouvez donner le statut « APPROVED » à quoi que ce soit avant d'avoir réalisé les changements dans le code perl. Notez que ce changement que j'ai mentionné peut aussi être réalisé en éditant checksetup.pl, qui automatise un bon nombre de choses. Mais vous avez besoin de connaître ce truc aussi, pas vrai ?

6. Intégrer Bugzilla avec des outils tiers

6.1. Bonsai

Bonsai est un outil orienté web pour gérer CVS, le Concurrent Versioning System [p 72] . En utilisant Bonsai, les administrateurs peuvent contrôler les états ouverts/fermés des arbres, interroger une base de données relationnelle rapide pour modifier, relier, et commenter des informations, et voir les changements effectués depuis la dernière fois que l'arbre a été fermé. Bonsai s'intègre également avec Tinderbox, the Mozilla automated build management system [p 73] .

6.2. CVS

Pour l'instant, l'intégration de CVS se fait mieux en utilisant l'entrée messagerie de Bugzilla.

Suivez les instructions de ce guide pour permettre l'intégration des courriels dans Bugzilla. Faites en sorte que vos scripts d'inscription envoient un courriel à l'entrée messagerie de Bugzilla avec en objet « [Bug XXXX] », et vous pouvez faire ajouter des commentaires d'inscriptions CVS à votre bogue Bugzilla. Si vous voulez que le bogue se ferme automatiquement, vous devrez modifier le script `contrib/bugzilla_email_append.pl`.

Il y a également un projet CVSZilla, basé sur un code de Bugzilla qui date, pour intégrer CVS et Bugzilla grâce aux capacités de CVS à envoyer des courriels. Allez voir à l'adresse : http://home-pages.kcbbs.gen.nz/~tonyg/.

Un autre système capable d'intégrer CVS à Bugzilla est Scmbug. Ce système fournit une intégration générique de la gestion de la configuration du code source avec suivi des bogues. Jetez-y un œil à l'adresse : http://freshmeat.net/projects/scmbug/.

6.3. Perforce SCM

Vous pouvez trouver la page de projet pour l'intégration de Bugzilla et de Teamtrack Perforce (p4dti) à l'adresse : http://www.ravenbrook.com/project/p4dti/. « p4dti » est maintenant un produit officiellement supporté de Perforce, et vous pouvez trouver la page p4dti « Perforce Public Depot » à l'adresse http://public.perforce.com/public/perforce/p4dti/index.html.

L'intégration de Perforce avec Bugzilla, une fois que les correctifs sont installés, se fait sans difficulté. Les informations de réplication de Perforce apparaîtront au-dessous des commentaires de chaque bogue. Veillez à avoir un ensemble assorti de correctifs pour la version de Bugzilla que vous êtes en train d'installer. p4dti est conçu pour être compatible avec les dispositifs de suivi des défauts multiples, et pour mettre à jour sa propre documentation dans ce but. Veuillez consulter les pages en lien ci-dessus pour de plus amples informations.

6.4. Subversion

Subversion est un système libre de suivi des révisions, conçu pour palier à différentes limites de CVS. L'intégration de Subversion dans Bugzilla est possible en utilisant Scmbug, système qui propose une intégration générique de la gestion de la configuration du code source avec suivi des bogues. Scmbug est disponible à l'adresse : http://freshmeat.net/projects/scmbug/.

6.5. Tinderbox/Tinderbox2

Tinderbox est un système de construction en continu pouvant s'intégrer à Bugzilla — consultez http://www.mozilla.org/projects/tinderbox pour plus d'information sur Tinderbox et http://tinderbox.mozilla.org/showbuilds.cgi pour le voir à l'œuvre.

Chapitre 6. L'utilisation de Bugzilla

Table des matières

1. Introduction

Cette section contient des informations pour les utilisateurs de Bugzilla. Il existe des installations de test de Bugzilla, appelée landfill, avec laquelle vous pouvez jouer tant que vous voulez (si elle est disponible). Par contre, les installations de Bugzilla là-bas n'ont pas toutes les fonctionnalités de Bugzilla activés et les installations différentes utilisent des versions différentes, ce qui fait que certaines choses ne marcheront pas tout a fait tel que le décrit ce document.

2. Créez un compte Bugzilla

Si vous voulez utiliser Bugzilla, vous devez commencer par créer un compte. Consulter l'administrateur responsable de votre installation de Bugzilla pour l'URL que vous devez utiliser pour y accéder. Si vous voulez tester Bugzilla, utilisez cette URL : http://landfill.bugzilla.org/bugzilla-2.18-branch/.

1. Cliquer sur le lien « Ouvrir un nouveau compte Bugzilla », entrer votre adresse de messagerie électronique et, optionnellement, votre nom dans les espaces prévues à cette effet, puis cliquez sur « Créer le compte ».

2. Quelques instants plus tard, vous allez recevoir un courriel à l'adresse donnée, qui contient votre nom d'utilisateur (en général, c'est l'adresse de messagerie elle-même) et un mot de passe. Ce mot de passe est généré de manière aléatoire mais peut être changé vers quelque chose de plus facile à retenir.

3. Cliquer sur le lien « Log In » dans le pied de page en bas de la page dans votre navigateur, entrer votre adresse de messagerie et mot de passe dans les espaces prévues à cette effet et cliquer sur « Login ».

Vous êtes maintenant connecté. Bugzilla utilise des cookies pour se rappeler que vous êtes connectés donc, a moins que vous ayez désactivé les cookies ou que votre adresse IP change, vous ne devriez plus avoir à vous connecter a nouveau.

3. Anatomie d'un bogue

Le noyau de Bugzilla est l'écran qui affiche un bogue particulier. C'est le lieu idéal pour expliquer quelques concepts de Bugzilla. Le bogue 1 de landfill est un bon exemple. Notez que les étiquettes de la plupart des champs sont des hyperliens ; en cliquant dessus vous ferez apparaître une aide contextuelle sur ce champ particulier. Les zones marquées d'un * peuvent ne pas être présentes sur chaque installation de Bugzilla.

1. *Produit et Composant* : les bogues sont divisés en produits et composants, un produit étant composé d'un ou plusieurs composants. Par exemple, le produit « Bugzilla » de bugzilla.mozilla.org est composé de plusieurs composants :

   Administration : administration d'une installation de Bugzilla.

   Bugzilla-General : tout ce qui ne va pas dans les autres composants, ou couvre plusieurs composants.

   Creating/Changing Bugs : créer, modifier, et visualiser les bogues.

   Documentation : documentation de Bugzilla, y compris le guide Bugzilla.

   Email : tout ce qui est en rapport avec un courriel envoyé par Bugzilla.

   Installation : procédé d'installation de Bugzilla.

   Query/Buglist : fonctions de recherche de bogues et visualisation des listes des bogues.

   Reporting/Charting : obtenir des comptes-rendus de Bugzilla.

   User Accounts : administration du compte utilisateur selon le point de vue de l'utilisateur. Requêtes sauvegardées, création de comptes, changement de mot de passe, authentification, etc.

   User Interface : problèmes généraux en relation avec l'interface utilisateur (et pas les fonctionnalités) comprenant les problèmes d'ordre esthétique, les modèles HTML, etc.

2. *Statut et Résolution :* définit avec précision quel est l'état du bogue : depuis l'état où il n'est même pas confirmé comme bogue, jusqu'à celui où il a été corrigé et où la correction est confirmée par l'Assurance Qualité. Les différents statuts possibles pour les rubriques Statut et Résolution concernant votre installation devraient être documentées dans l'aide contextuelle relative à ces éléments.

3. *Assigné À :* personne étant chargée de corriger le bogue.

4. **URL :* une éventuelle URL associée au bogue.

5. *Sommaire :* problème résumé en une phrase.

6. **Tableau Blanc des États :* (appelé également Tableau Blanc) zone de texte libre pour ajouter de courtes notes ou des étiquettes à un bogue.

7. **Mots Clés :* l'administrateur peut définir des mots-clés que vous pouvez employer pour étiqueter et classer les bogues par catégorie ; par exemple le projet Mozilla a des mots-clés comme plantage et régression.

8. *Platforme et SE :* indiquent l'environnement de travail dans lequel le bogue a été détecté.

9. *Version :* la zone « version » est habituellement utilisée pour les versions d'un produit qui sont parues et sert à indiquer quelle version d'un composant est touchée par le problème dont le compte rendu de bogue fait état.

10. *Priorité :* le responsable de bogues emploie ce champ pour fixer la priorité de ses bogues. Il faut éviter de le changer sur les bogues des autres.

11. *Sévérité :* indique la gravité du problème : depuis bloquant (« application inutilisable ») à insignifiant (« problème esthétique mineur »). Vous pouvez également employer cette zone pour indiquer si un bogue est une demande d'amélioration.

12. **Cible :* (aussi intitulé Cible Jalon) version future pour laquelle le bogue devra être corrigé. Par exemple, les jalons du projet de Bugzilla pour de futures versions de Bugzilla sont 2.18, 2.20, 3.0, et cætera. Les jalons ne sont pas limitées aux nombres, par conséquent vous pouvez utiliser n'importes quelles chaînes de caractères textuelles, telles que des dates.

13. *Rapporteur :* personne qui a signalé l'anomalie.

14. *Liste CC :* liste de personnes à prévenir par courriel quand le bogue change.

15. *Pièces Jointes :* vous pouvez joindre des fichiers (par exemple des cas qui constituent des précédents ou des correctifs) aux bogues. S'il y a des pièces jointes, elles sont listées dans cette section.

16. **Dépendances :* si ce bogue ne peut pas être corrigé tant que d'autres bogues ne sont pas eux-mêmes corrigés (dépend de), ou si ce bogue annule la correction d'autres bogues déjà corrigés (bloque), leurs numéros sont enregistrés ici.

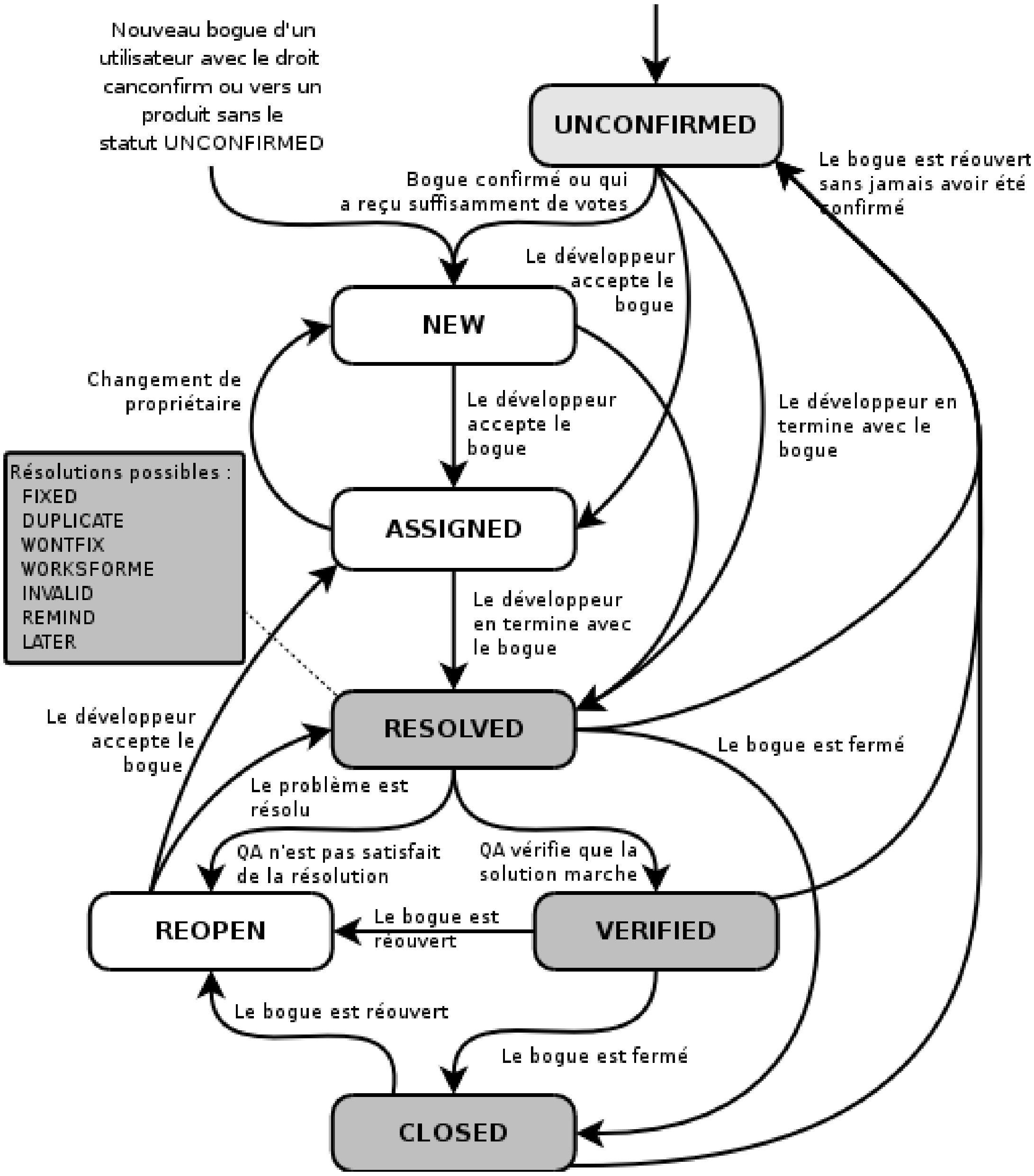
17. **Votes :* si ce bogue a reçu des voix.

18. *Commentaires Supplémentaires :* vous pouvez ajouter ici votre grain de sel à la discussion sur le bogue, si vous avez quelque chose d'intéressant à dire.

4. Cycle de vie d'un bogue de Bugzilla

Le cycle de vie, aussi appelé déroulement des opérations, d'un bogue est actuellement codé en dur dans Bugzilla. Figure 6.1, « Cycle de vie d'un bogue de Bugzilla » [p 76] contient un représentation graphique de ce cycle de vie. Si vous souhaitez personnaliser cette image pour votre site, le fichier diagramme est disponible dans le format XML native de Dia

Figure 6.1. Cycle de vie d'un bogue de Bugzilla



5. Recherche de bogues

La page de recherche de Bugzilla est l'interface où vous pouvez trouver n'importe quel compte-rendu de bogue, commentaire, ou correctif actuellement dans le système Bugzilla. Vous pouvez jouer avec en allant à : http://landfill.bugzilla.org/bugzilla-2.18-branch/query.cgi.

La page de recherche propose des commandes pour choisir différentes valeurs possibles pour tous les champs d'un bogue, comme décrit ci-dessus. Pour certains champs, plusieurs valeurs peuvent êtres sélectionnées. Si c'est le cas, Bugzilla renvoie les bogues dont le contenu du champ correspond à l'une quelconque des valeurs choisies. Si aucune n'est sélectionnée, le champ peut prendre n'importe quelle valeur.

Une fois que vous avez lancé une recherche, vous pouvez la sauvegarder en tant que recherche nommée, option qui apparaît dans le pied de page.

Les requêtes hautement avancées sont exécutées en utilisant des diagrammes booléens. Reportez vous aux liens d'aide à propos des diagrammes booléens sur la page de recherche pour de plus amples informations.

6. Listes de Bogues

Si vous lancez une recherche, une liste de bogues correspondants vous sera renvoyée.

Le format de la liste est paramétrable. Par exemple, on peut effectuer un tri en cliquant sur les en-têtes de colonne. D'autres fonctionnalités utiles sont accessibles en utilisant les liens en bas de la liste :

Format Long : vous donne une grande page avec un résumé, qui ne peut pas être édité, des champs de chaque bogue.

CSV : pour obtenir la liste de bogues sous forme de valeurs séparées par des virgules, afin par exemple de l'importer dans une feuille de calculs.

Changer les colonnes : pour modifier les attributs des bogues qui apparaissent dans la liste.

Changer plusieurs Bugs à la fois : si votre compte possède l'habilitation suffisante, vous pouvez appliquer la même modification à tous les bogues de la liste; par exemple, changer leur propriétaire.

Envoyer un mail aux propriétaires des bugs : Sends mail to the owners of all bugs on the list.

Éditer cette requête : si vous n'avez pas obtenu exactement les résultats que vous recherchiez, vous pouvez retourner à la page de requête via ce lien et effectuer de légères révisions dans la requête que vous venez de créer afin d'obtenir des résultats plus précis.

Sauvegarder cette requête sous : vous pouvez donner un nom à une recherche et en conserver l'association; un lien apparaîtra alors dans votre pied de page, vous en fournissant un accès rapide afin de relancer la recherche plus tard.

7. Établissement d'un rapport de bogue

Des années d'expérience sur le signalement de bogues ont été condensées pour votre plus grand plaisir dans les Recommandations d'Écriture de Bug. Si certains de ces conseils sont spécifiques à Mozilla, les principes de base de signalement de bogues reproductibles spécifiques, d'identification du produit que vous utilisez, sa version, le composant dont provient l'erreur, la plate-forme matérielle, et le système d'exploitation que vous utilisiez au moment du plantage, sont d'une aide précieuse pour être sûr d'obtenir des corrections responsables et précises du bogue qui vous a attaqué.

La procédure pour renseigner un fichier de bogue est la suivante :

1. Allez sur le site de Landfill dans votre navigateur et cliquez Nouveau.

2. Choisissez un produit : n'importe lequel fera l'affaire.
3. Complétez les champs. Bugzilla devrait par déduction proposer, d'après votre navigateur, les bonnes listes déroulantes « Platforme » et « SE ». Si celles-ci sont erronées, corrigez les.
4. Choisissez « Soumettre » et envoyez votre rapport de bogue.

Veillez à faire en sorte que tout ce qui a été dit dans le résumé est aussi présent dans le premier commentaire. Les résumés sont souvent mis à jour, ce qui rendra votre information initiale facilement accessible.

Il est inutile d'indiquer « any » ou des chaînes de caractères similaires dans le champ URL. S'il n'y a pas d'URL spécifique associée au bogue, laissez ce champ vide.

Si vous pensez qu'un bogue que vous avez classé a été marqué par erreur comme dupliqué à partir d'un autre, veuillez soulever la question dans votre bogue, pas dans celui où il a atterri. N'hésitez pas à ajouter la personne responsable de la manipulation dans la liste CC si elle n'y figure pas déjà.

8. Visualisateur de correctifs

Consulter et passer des correctifs en revue dans Bugzilla est souvent difficile à cause de la perte de contexte, des formats inadaptés et des problèmes inhérents de lisibilité que les correctifs bruts présentent. Patch Viewer est une amélioration de Bugzilla conçue pour résoudre ce problème en proposant un meilleur contexte, des liens vers des sections, ainsi qu'en s'intégrant à Bonzai, LXR et CVS.

Patch Viewer vous permet de :

Voir les correctifs en couleur, avec une représentation de type côte à côte, plutôt que de tenter d'interpréter le contenu du correctif.

Souligner les différences entre deux correctifs.

Obtenir plus de contexte pour un correctif.

Réduire ou étendre les sections d'un correctif pour une lecture aisée.

Établir un lien vers une section particulière d'un correctif pour une discussion ou une révision.

Aller sur Bonzai ou LXR pour accéder à un contexte plus fourni, retrouver les responsables et les références croisées correspondant à la partie du correctif que vous êtes en train d'examiner.

Créer un diff au format unifié texte brut depuis n'importe quel correctif, peu importe le format dont ce dernier provient.

8.1. Consultation des correctifs dans Patch Viewer

Le moyen principal pour visualiser un correctif dans Patch Viewer est de cliquer sur le lien « Diff » situé à côté d'un correctif dans la liste des pièces joints d'un bogue. Vous pouvez également y arriver en cliquant sur le bouton « Visualiser le fichier joint en tant que Diff » dans l'écran d'édition des pièces jointes de la fenêtre d'édition.

8.2. Voir la différence entre deux correctifs

Pour voir la différence entre deux correctifs, vous devez en premier lieu visualiser le correctif le plus récent dans Patch Viewer. Puis sélectionnez le correctif le plus ancien dans le menu déroulant en haut de la page (« Différences entre menu déroulant et ce correctif ») et cliquez sur le bouton « Diff ». Vous verrez ainsi ce qui est nouveau ou ce qui a changé dans le correctif le plus récent.

8.3. Obtenir plus de contexte dans un correctif

Afin d'obtenir plus de contexte dans un correctif, indiquez un nombre dans la zone de texte en haut de Patch Viewer (« Correctif / Fichier / zone de texte ») et appuyez sur Entrée. Cela vous fournira un nombre de lignes relatives au contexte avant et après chaque modification. Une autre possibilité est de cliquer sur le lien « Fichier » présent; cela fera apparaître chaque changement dans l'intégralité du contexte du fichier. Cette fonctionnalité fonctionne avec les fichiers qui ont été mis au format « diff » en utilisant « cvs diff ».

8.4. Réduction et déploiement des sections d'un correctif

Pour visualiser uniquement un assortiment de fichiers dans un correctif (par exemple, si un correctif est absolument gigantesque et que vous souhaitez n'en revoir qu'une partie à la fois), vous pouvez cliquer sur les liens « (+) » et « (-) » à côté de chaque fichier (afin de le réduire ou de l'étendre). Si vous voulez réduire tous les fichiers ou déployer tous les fichiers, vous pouvez cliquer sur les liens « Réduire tout » et « Étendre tout » du haut de la page.

8.5. Établir un lien vers une section d'un correctif

Pour établir un lien vers une section d'un correctif (par exemple, si vous voulez être en mesure de donner à quelqu'un une URL pour lui indiquer de quelle partie vous parlez), cliquez simplement sur le lien « Établir un lien ici » de l'entête de section. L'URL qui en résulte peut alors être copiée et utilisée dans une discussion. (Copier la cible du lien dans Mozilla fonctionne tout aussi bien.)

8.6. Se rendre sur Bonzai et LXR

Pour aller sur Bonsai afin de voir les responsables des lignes qui vous intéressent, vous pouvez cliquer sur le lien « Lignes XX-YY » de l'entête de section qui vous intéresse. Cela fonctionne même si le correctif correspond à une version ancienne du fichier, puisque Bonsai stocke toutes les versions du fichier.

Pour accéder à LXR, cliquez sur le nom de fichier de l'entête du fichier (malheureusement, puisque LXR ne s'occupe que de la version la plus récente, il est probable que les numéros de ligne ne correspondent plus).

8.7. Créer un diff unifié

Si le correctif n'est pas dans le format que vous souhaitez, vous pouvez le convertir dans un format diff unifié en cliquant le lien « brut unifié » en haut de la page.

9. Trucs et astuces

Cette section détaille quelques-unes des astuces de Bugzilla et pratiques d'usage qui ont été développées.

9.1. Création automatique de liens

Les commentaires de Bugzilla sont sous forme de texte ordinaire ; Ainsi le fait de taper <U> produira inférieur à, U, supérieur à, au lieu de souligner le texte. Cependant, Bugzilla fera automatiquement la transformation de certains types de textes en liens dans les commentaires. Par exemple, le texte « http://www.bugzilla.org » sera transformé en lien : http://www.bugzilla.org. Les autres chaînes de caractères qui sont transformées en liens de façon classique sont les suivants :

bug 12345

comment 7

bug 23456, comment 53

attachment 4321

mailto:george@example.com

george@example.com

ftp://ftp.mozilla.org

Most other sorts of URL

Un corollaire à ceci est que, si vous tapez un numéro de bogue dans un commentaire, vous devez mettre le mot « bug » devant, de façon à ce que le lien soit fait de façon automatique pour que ce soit pratique pour tous.

9.2. Quicksearch

Quicksearch est un outil de requête muni d'un champ de recherche unique qui emploie des méta-caractères pour indiquer ce qui doit être recherché. Par exemple, si vous tapez « `foo|bar` » dans Quicksearch, Quicksearch cherchera le mot « foo » ou bien le mot « bar » dans le résumé et le tableau blanc des états d'un bogue ; ajouter « `:BazProduct` » effectuera la requête seulement dans ce produit.

Vous trouverez le cadre de recherche Quicksearch sur la page d'accueil de Bugzilla, avec un lien d' aide qui explique comment l'utiliser.

9.3. Commentaires

Si vous modifiez les champs d'un bogue, n'ajoutez un commentaire que si vous avez quelque chose de pertinent à dire, ou si Bugzilla l'exige. Autrement, votre modification pourrait entraîner l'envoi de courriels inutiles à certaines personnes. Pour prendre un exemple : un utilisateur peut configurer son compte de façon à filtrer les messages par lesquels quelqu'un s'ajoute simplement à la zone de cc d'un bogue (ce qui se produit souvent). Si vous vous ajoutez dans la zone « cc », et ajoutez un commentaire disant « je m'ajoute dans la rubrique CC », la personne obtiendra un courriel sans intérêt qu'elle aurait autrement évité.

N’utilisez pas les signatures dans les commentaires. Signer avec votre nom (« Bill ») est acceptable, en particulier si vous avez l’habitude de le faire, mais des créations artistiques surchargées en ASCII de plusieurs lignes ne le sont pas.

9.4. Pièces jointes

Utilisez les pièces jointes, plutôt que les commentaires, pour de grandes quantités de données ASCII, tels que les fichiers de traçage, ou les fichiers de log de débogage. C’est pour éviter d’infliger des bogues surchargés à ceux qui veulent les lire, et pour qu’ils ne reçoivent pas des courriers inutiles et de taille importante.

Allégez les copies d’écrans. Il n’est pas nécessaire de montrer l’écran entier pour signaler un problème qui peut être illustré avec seulement quelques pixels.

Ne joignez pas des cas qui constituent des précédents (par exemple un fichier HTML, un fichier CSS et une image) comme fichier ZIP. Au lieu de cela, envoyez-les dans l’ordre inverse et éditez le fichier référent de sorte qu’il pointe vers les fichiers joints. De cette façon, le cas qui constitue un précédent fonctionne immédiatement hors du bogue.

Bugzilla stocke et utilise un type de contenu pour chaque pièce jointe (par exemple text/html). Pour télécharger une pièce jointe présentant un type de contenu différent (par exemple application/xhtml+xml), vous pouvez prendre la main en utilisant un paramètre ’content-type’ dans l’URL, par exemple `&content-type=text/plain`.

10. Préférences utilisateurs

Une fois que vous avez ouvert une session, vous pouvez personnaliser divers aspects de Bugzilla par l’intermédiaire du lien « Edit prefs » en pied de page. Les préférences sont divisées en trois onglets :

10.1. Configuration du compte

Dans cette partie, vous pouvez changer les informations de base sur votre compte, y compris votre mot de passe, votre adresse électronique et votre vrai nom. Pour des raisons de sécurité, avant de changer quoi que ce soit sur cette page, vous devez taper votre mot de passe *actuel* dans la zone « Mot de passe » située en haut de la page. Si vous essayez de changer votre adresse électronique, un courriel de confirmation sera envoyé aussi bien à votre ancienne qu’à votre nouvelle adresse, avec un lien à utiliser pour confirmer le changement. Ceci a pour but d’essayer d’empêcher le détournement de compte.

10.2. Paramètres des courriels

Cet onglet contrôle la quantité de courriels que Bugzilla vous envoie.

Le premier élément sur cette page s’intitule « Utilisateurs à observer ». Si vous entrez un ou plusieurs comptes utilisateurs, séparés par des virgules, dans la zone de texte, vous recevrez une copie de tous les courriels de rapport de bogue envoyés à ces utilisateurs (si les paramètres de sécurité le permettent). Cette puissante fonctionnalité permet de réaliser des transitions en douceur lorsque des développeurs changent de projet, ou que des utilisateurs partent en vacances.

Note

La capacité à observer d’autres utilisateurs pourrait ne pas s’avérer disponible sur toutes les installations de Bugzilla. Si vous ne voyez pas cette fonctionnalité et que vous pensez en avoir besoin, parlez-en à votre administrateur.

En général, les utilisateurs ont quasiment un contrôle complet de la quantité (importante ou faible) de courriels que Bugzilla leur envoie. Si vous souhaitez recevoir la quantité maximum de courriels possible, cliquez sur le bouton « Activer tous les courriels ». Si vous souhaitez ne recevoir aucun courriel de Bugzilla, cliquez sur le bouton « Désactiver tous les courriels ».

Note

Votre administrateur Bugzilla peut empêcher un utilisateur de recevoir des courriels de rapport de bogue en ajoutant le nom de l’utilisateur dans le fichier `data/nomail`. Il s’agit d’un dernier recours, utilisé à bon escient uniquement dans le cas des comptes désactivés, puisqu’il supplante les préférences de courriels individuelles des utilisateurs.

Si vous désirez régler vos courriels de rapport de bogue à un niveau autre que « Complètement Activé » et « Complètement Désactivé », c’est précisément le tableau « Champ/Options spécifique du destinataire » qui vous permet de le faire. Les lignes du tableau définissent les évènements pouvant survenir sur un bogue, comme par exemple l’ajout de pièces jointes, de nouveaux commentaires, un changement de priorité, etc. Les colonnes du tableau définissent votre relation avec le bogue :

- Rapporteur — vous identifie comme étant la personne qui a signalé le bogue. Votre nom/compte apparaît dans le champ « Rapporteur: ».
- Assignée — vous identifie comme étant la personne désignée responsable du bogue. Votre nom/compte apparaît dans le champ « Assigné À : » du bogue.
- Contact QA — vous êtes l’un des contacts Assurance Qualité désignés pour le bogue. Votre compte apparaît dans la zone de texte « Contact QA : » du bogue.
- CC — vous êtes dans la liste CC (copies carbone) du bogue. Votre compte apparaît dans la zone de texte « CC: » du bogue.
- Votant — vous avez émis un ou plusieurs vote au sujet du bogue. Votre compte apparaît seulement si quelqu’un clique sur le lien « Montrer les votes concernant ce bogue » du bogue.

Note

Certaines colonnes peuvent s’avérer être invisibles dans votre installation, cela dépend de la configuration de votre site.

Afin de paramétrer au mieux vos courriels de rapports de bogue, choisissez donc les évènements pour lesquels vous souhaitez recevoir des courriels de rapport de bogue; puis choisissez si vous voulez en recevoir en permanence (activez la case à cocher pour toutes les colonnes), ou seulement lorsque vous avez une certaine relation avec un bogue (n’activez la case à cocher que pour ces colonnes). Par exemple : si vous ne vouliez pas recevoir de courriel lorsque quelqu’un s’ajoute à la liste CC, vous pourriez décocher toutes les cases de la ligne « Changements dans le champ CC ». Un autre exemple : si vous vouliez ne jamais recevoir de courriels à propos des bogues que vous signalez tant que ceux-ci n'ont pas été traités, vous pourriez décocher toutes les cases de la colonne « Rapporteur » à part celle correspondant à la ligne « Le bogue est traité ou vérifié ».

Note

Bugzilla ajoute l'entête « X-Bugzilla-Reason » à tous les courriels qu'il envoie, en y décrivant la relation du destinataire avec le bogue (AssignéÀ, Rapporteur, ContactQA, CC ou Votant). Cet en-tête peut être utilisé pour poursuivre le filtrage du côté client.

Deux éléments hors du tableau (« M'envoyer un courriel lorsque quelqu'un me demande de positionner un fanion » et « M'envoyer un courriel lorsque quelqu'un positionne un fanion dont j'ai fait la demande ») définissent la manière dont vous souhaitez recevoir les courriels de rapport de bogue au vu des fanions. Leur utilisation est assez simple : activez les cases à cocher si vous voulez que Bugzilla vous envoie des courriels à l'une ou l'autre des conditions ci-dessus.

Par défaut, Bugzilla envoie des courriels sans se préoccuper de qui a effectué le changement... même si vous étiez au départ le responsable de l'envoi du courriel. Si ça ne vous intéresse pas de recevoir des courriels de rapport de bogue dûs à vos propres changements, cochez la case marquée « M'envoyer uniquement les rapports de changements effectués par d'autres personnes ».

10.3. Permissions

C'est une page purement informative qui décrit vos permissions en cours sur cette installation de Bugzilla : à quels groupes de produit vous appartenez, et si vous pouvez éditer les bogues ou assurer diverses fonctions d'administration.

11. Rapports et diagrammes

En plus de la liste de bogues standard, Bugzilla propose deux autres manières de consulter des groupes de bogues. Ces méthodes sont les rapports (qui donnent différentes vues de l'état courant de la base de données) et les diagrammes (qui donnent une représentation graphique des changements dans un groupe de bogues particuliers dans la durée).

11.1. Rapports

Un rapport est une vue de l'état courant de la base de données des bogues.

Vous pouvez lancer soit un rapport basé sur des tableaux HTML, soit un rapport graphique de type courbe/camembert/histogramme. Les deux présentent différentes pages pour les définir, mais sont de proches cousins ; une fois que vous avez défini et visualisé un rapport, vous pouvez passer à votre guise de l'une à l'autre des différentes représentations des données.

Les deux types de rapports ont pour principe de définir un groupe de bogues en utilisant l'interface de recherche standard, puis en choisissant certains aspects de ce groupe que vous voulez représenter sur les axes horizontal et/ou vertical. Vous pouvez également obtenir une sorte de rapport tridimensionnel en choisissant d'avoir de multiples images ou tableaux.

Ainsi, par exemple, vous pourriez utiliser le format recherche pour choisir « tous les bogues du produit WorldControl », puis tracer leur degré de gravité par rapport à leur composants, afin d'observer quel composant a eu le plus grand nombre de mauvais bogues signalés.

Une fois que vous avez défini vos paramètres et cliqué sur « Générer le rapport », vous pouvez choisir entre HTML, CSV, Histogramme, Courbe et Camembert. (Note : Le camembert n'est disponible que si vous ne définissez pas d'axe vertical, puisque les diagrammes camembert n'en ont pas) Les autres commandes sont largement explicites ; vous pouvez changer la taille de l'image si les textes se superposent, ou que vous ne voyez pas les barres parce-qu'elles sont trop fines.

11.2. Les diagrammes

Un diagramme est une représentation de l'état de la base de données par rapport au temps.

Bugzilla possède deux systèmes de diagrammes : les anciens diagrammes et les nouveaux diagrammes. Les anciens diagrammes font partie de Bugzilla depuis longtemps ; ils représentent chaque statut et chaque résolution pour chaque produit, et c'est tout. Ils sont dénigrés et vont bientôt disparaître ; nous ne nous étendrons pas plus là-dessus. L'avenir est dans les nouveaux diagrammes : ils vous permettent de représenter tout ce que vous pouvez définir comme étant une recherche.

Note

Les deux formes de diagrammes requièrent la configuration du script collecteur de données par l'administrateur. Si vous ne pouvez voir aucun diagramme, demandez lui s'il a pris les dispositions nécessaires.

Une courbe particulière sur un diagramme s'appelle une série de données. Tous les ensembles de données sont organisés en catégories et sous-catégories. Les séries de données que Bugzilla définit automatiquement utilisent le nom de produit comme catégorie et les noms des composants comme sous-catégorie, mais il n'est pas nécessaire de respecter ce schéma de noms avec vos propres diagrammes si vous ne le souhaitez pas.

Les séries de données peuvent être publics ou privés. Tout le monde voit les séries de données publics dans la liste mais seul leur créateur peut voir les séries de données privés. Seuls les administrateurs peuvent rendre un ensemble de données public. Deux ensembles de données, mêmes s'ils sont privés, ne peuvent pas avoir le même ensemble de catégorie, sous-catégorie et nom. Ainsi, si vous créez des séries de données privés, c'est une bonne idée que de mettre votre nom d'utilisateur dans la catégorie.

11.2.1. Création de diagrammes

On crée un diagramme en sélectionnant un certain nombre de séries de données de la liste et en appuyant sur « Ajouter à la liste » pour chacun. Dans la liste des séries de données à représenter, vous pouvez définir l'étiquette que portera cette série de données dans la légende du diagramme, ainsi que demander à Bugzilla de faire la somme d'un certain nombre de séries de données (par exemple vous pourriez additionner les séries de données représentant RESOLVED, VERIFIED et CLOSED dans un produit particulier afin d'obtenir une série de données représentant tous les bogues résolus dans ce produit).

Si vous avez ajouté par erreur une série de données à la liste des données à représenter, sélectionnez le en utilisant la case à cocher et cliquez sur Supprimer. Dès que vous ajoutez plus d'une série de données, une ligne « Total général » apparaît automatiquement en bas de liste. Si vous n'en voulez pas, supprimez la simplement comme vous le feriez pour n'importe quel autre ligne.

Vous pourriez aussi vouloir représenter les données seulement pour une certaine période de temps, et cumuler les résultats, c'est-à-dire, représenter chaque donnée en utilisant la précédente comme ligne zéro, afin que la ligne du haut donne la somme de toutes les séries de données. C'est plus facile à essayer qu'à expliquer :-)

Une fois qu'une série de données a été placé dans la liste, on peut aussi effectuer certaines actions dessus. Par exemple, on peut éditer les paramètres de la série de données (nom, fréquence, etc.) si c'est une liste que vous avez créée ou si vous êtes administrateur.

Dès que vous êtes satisfait, cliquez sur « mettre la liste sous forme de diagramme » pour visualiser le diagramme.

11.2.2. Création de nouvelles séries de données.

Vous pouvez aussi créer des séries de données de votre cru. Pour ce faire, cliquez sur le lien « créer une nouvelle série de données » sur la page de création de diagrammes. Ce lien vous emmène vers une interface de type recherche où vous pouvez définir la recherche que Bugzilla va représenter. En bas de la page, vous choisissez la catégorie, la sous-catégorie et le nom de votre nouvelle série de données.

Si vous avez les autorisations suffisantes, vous pouvez rendre la série de données public et réduire la fréquence de collecte des données à moins des sept jours par défaut.

12. Fanions

Un fanion est une sorte de statut pouvant être associé à un bogue ou à une pièce jointe pour indiquer un état particulier dans lequel le bogue/la pièce jointe se trouve. Chaque installation peut définir ses propres groupes de fanions pouvant être associés à un bogue ou une pièce jointe.

Si votre installation a défini un fanion, vous pouvez activer ou désactiver ce fanion, et si votre administrateur a autorisé les demandes de fanions, vous pouvez soumettre une requête pour qu'un autre utilisateur active le fanion.

Pour activer le fanion, sélectionnez soit '+', soit '-' dans le menu déroulant à côté du nom du fanion dans la liste « Fanions ». Les significations de ces valeurs sont spécifiques aux fanions et de ce fait ne peuvent être décrites dans cette documentation, mais pour donner un exemple, mettre un fanion nommé « review » à la valeur '+' peut indiquer que le bogue/la pièce jointe a passé avec succès l'étape où on l'examine, tandis que le régler à '-' indiquerait que le bogue/la pièce jointe ne l'a pas franchie.

Pour désactiver un fanion, cliquez sur son menu déroulant et sélectionnez la valeur vide.

Si votre administrateur a permis les demandes de fanion, effectuez une requête de fanion en sélectionnant « ? » dans le menu déroulant et entrez le nom de l'utilisateur que vous souhaitez voir activer le fanion, dans le champ de texte à côté du menu.

Un fanion activé apparaît dans les rapports de bogues ainsi que dans les pages « édition des pièces jointes » avec, placé avant le fanion, le nom abrégé de l'utilisateur qui l'a activé. Par exemple, si Jack règle un fanion « review » sur « + », cela donne :

```
Jack : review [ + ]
```

Un fanion demandé apparaît avec le nom de l'utilisateur qui en a fait la demande placé avant le nom du fanion, et l'utilisateur auquel on demande de mettre le fanion est ajouté entre parenthèses après le nom du fanion. Par exemple, si Jack demande à Jill d'examiner le bogue, cela donne :

```
Jack : review [ ? ] (Jill)
```

Annexe A. La foire aux questions de Bugzilla

Cette FAQ contient des questions qui ne sont pas traitées ailleurs dans ce guide.

A.1. Questions générales

A.1.1. Sous quelle licence est distribué Bugzilla ?

Bugzilla est couvert par la Mozilla Public License. Pour plus de détails, reportez-vous à http://www.mozilla.org/MPL/.

A.1.2. Comment obtenir une assistance commerciale pour Bugzilla ?

http://bugzilla.org/consulting.html contient une liste d'entreprises et de particuliers qui nous ont demandé de les faire apparaître en tant que consultants pour Bugzilla.

On trouve plusieurs inconditionnels de Bugzilla expérimentés sur la liste de diffusion ou le forum et qui sont disposés à rendre service en échange d'une généreuse indemnité. Essayez en envoyant un message à la liste de diffusion pour rechercher un volontaire.

A.1.3. Quelles sont les entreprises ou les projets de grande envergure qui utilisent actuellement Bugzilla pour le suivi de bogues ?

Il existe des douzaines d'entreprises de première importance qui ont des sites publiques Bugzilla pour suivre les bogues au sein de leurs produits. Une liste assez complète est disponible sur notre site à l'adresse http://bugzilla.org/installation-list/. Si vous possédez une installation de Bugzilla et que vous souhaitez être ajouté à cette liste, que votre installation soit publique ou non, envoyez un courriel à Gerv <`gerv CHEZ mozilla POINT org`> (en anglais).

A.1.4. Qui assure la maintenance de Bugzilla ?

Un noyau dur de développeurs, dirigé par Dave Miller (<`justdave CHEZ bugzilla POINT org`>).

A.1.5. Que vaut Bugzilla par rapport aux autres bases de données pour le suivi de bogues ?

Nous n'avons pu trouver aucune étude comparative entre Bugzilla et d'autres logiciels de suivi d'anomalies. Si vous en connaissez, merci de nous contacter. Cependant, d'après l'expérience personnelle de Matthew Barnson (qui a démarré cette FAQ), Bugzilla offre des performances supérieures sur le matériel grand public, à un meilleur prix (gratuit !), plus de fonctionnalités facilitant la tâche des développeurs (comme la sauvegarde des requêtes, l'intégration du courrier électronique et l'indépendance vis-à-vis de la plate-forme), une adaptabilité améliorée, une plus grande flexibilité et une plus grande facilité d'utilisation par rapport aux logiciels de suivi de bogues commercial.

Si vous développez un système de suivi de bogues commercial et si vous souhaitez nous soumettre une liste des avantages de votre produit par rapport à Bugzilla, envoyez la tout simplement à <`documentation CHEZ bugzilla POINT org`>. Nous serions heureux d'inclure ce comparatif dans notre documentation.

A.1.6. Pourquoi Bugzilla ne propose-t-il pas telle ou telle fonctionnalité ou compatibilité avec tel autre logiciel de suivi de bogues ?

Il se peut que le support n'ait pas encore été développé ou que vous ne l'ayez pas trouvé. Bien que Bugzilla fasse à chaque version des progrès en termes de facilité d'emploi, de personnalisation, d'adaptabilité et d'interface utilisateur, cela ne signifie pas que Bugzilla n'est pas améliorable !

Le meilleur moyen d'effectuer une demande d'amélioration est de répertorier un bogue auprès de bugzilla.mozilla.org et d'en définir le degré de gravité (*SEVERITY* au niveau « amélioration » (*ENHANCEMENT*). Votre demande d'amélioration (RFE, *Request For Enhancement*) sera tout d'abord à l'état NON-CONFIRMÉE, et y restera jusqu'à ce que la personne compétente pour CONFIRMER le bogue l'examine. Si cette personne pense qu'il s'agit d'une requête justifiée qui va dans le sens des objectifs de Bugzilla, son état passera à NOUVEAU ; dans le cas contraire, cette personne vous expliquera vraisemblablement pourquoi, et le bogue passera dans l'état RÉSOLU/NE SERA PAS CORRIGÉ. Si quelqu'un d'autre a soumis la même requête (ou presque) avant vous, votre requête passera dans l'état RÉSOLU/DOUBLON, et un pointeur vers la RFE antérieure sera placé.

Même si votre RFE est approuvée, cela ne signifie pas qu'elle sera prise en compte dès la version suivante; nous avons un nombre limité de développeurs et de nombreuses requêtes... dont certaines sont plutôt complexes. Si vous êtes programmeur, vous pouvez participer au projet en élaborant vous même une correction qui produit la fonctionnalité que vous désirez. Si vous n'avez jamais contribué au développement de Bugzilla auparavant, merci de bien vouloir lire le Developers' Guide (guide du développeur) et le Contributors' Guide (guide du contributeur) avant toute chose.

A.1.7. Pourquoi MySQL ? J'aimerais bien que Bugzilla fonctionne avec Oracle/Sybase/Msql/PostgreSQL/MSSQL.

Au départ, MySQL a été choisi car il était gratuit, facile à installer et compatible avec le matériel sur lequel Netscape comptait le faire fonctionner.

Le travail est actuellement en cours pour faire fonctionner Bugzilla avec PostgreSQL ; vous pouvez suivre l'avancement de cette initiative dans le bogue 98304 (http://bugzilla.mozilla.org/show_bug.cgi?id=98304).

La prise en charge de Sybase a été abandonnée. Même si le développement reprenait, il est probable que cela ne fonctionnerait pas sans que l'entreprise utilisatrice y change à la main beaucoup de choses. Sybase n'est tout simplement PAS très conforme aux standards (malgré tout le battage qui a été fait) et il nous est apparu qu'il fallait modifier trop de choses pour que cela fonctionne comme par exemple transférer la moitié de la logique métier dans des procédures stockées afin d'en obtenir des performances convenables. Le bogue concerné est le bogue 173130.

Red Hat avait une version de Bugzilla fonctionnant sous Oracle, mais c'était il y a très, très longtemps. Cette version (Bugzilla 2.8) est maintenant obsolète, peu sûre et elle n'est plus du tout prise en charge. Chez Red Hat, l'actuelle version de Bugzilla (basée sur Bugzilla 2.17.1) utilise PostgreSQL, et des actions sont en cours pour introduire ces changements dans la distribution principale (voir ci-dessus). À ce jour, nous ne connaissons pas de version de Bugzilla portée pour Oracle (à notre humble avis, Bugzilla n'a aucun besoin de ce qu'Oracle apporte).

Le bogue 237862 est intéressant à lire si vous souhaitez savoir comment progresse la compatibilité générale des bases de données.

A.1.8. Qu'est-ce que `/usr/bonsaitools/bin/perl` ?

Le chemin pour Perl indiqué dans la première ligne des scripts Bugzilla était `/usr/bonsaitools/bin/perl` car, lorsque Terry a commencé à écrire le code pour mozilla.org, il avait besoin de contrôler complètement une version de Perl ainsi que d'autres outils. Ce chemin a été abandonné dans la version 2.18 en faveur du chemin, plus raisonnable, `/usr/bin/perl`. Si vous aviez installé une ancienne version de Bugzilla et créé le lien symbolique que nous suggérions alors, vous pouvez maintenant supprimer ce dernier (pourvu que vous n'ayez rien d'autre, comme par exemple Bonsai, qui en fasse usage et que vous n'envisagiez pas de réinstaller une ancienne version de Bugzilla).

A.1.9. Mon Perl se situe dans `/usr/local/bin/perl` et pas dans `/usr/bin/perl`. Existe-t-il un moyen simple de changer cela dans tous les fichiers où cela a été codé en dur ?

Le moyen le plus simple de résoudre ce problème est de créer un lien du premier répertoire vers le second : **ln -s usr/local/bin/perl /usr/bin/perl**. Si ce n'est pas possible, le petit morceau de Perl suivant remplacera toutes les premières lignes des fichiers (commençant par #! et contenant le chemin) par quelque chose d'autre :

```
perl -pi -e 's@#\!/usr/bin/perl@#\!/usr/local/bin/perl@' *cgi *pl
```

Malheureusement, cette ligne de commande ne fonctionnera pas sous windows à moins que vous ayez Cygwin. Cependant, MySQL contient un exécutable nommé **replace** qui se charge du travail :

```
C:\mysql\bin\replace "#!/usr/bin/perl" "#!C:\perl\bin\perl" -- *.cgi *.pl
```

Note

Si votre chemin vers Perl est encore différent, il vous suffit de remplacer `/usr/local/bin/perl` par votre propre chemin vers Perl dans les exemples ci-dessus.

Si vous utilisez Apache sous Windows, vous pouvez éviter tous ces problèmes en indiquant `Registry` (ou, si vous utilisez Apache dans une version ultérieure à 2, `Registry-Strict`) comme valeur de ScriptInterpreterSource. ScriptInterpreter-Source a besoin d'une entrée de la base de registres « HKEY_CLASSES_ROOT\.cgi\Shell\ExecCGI\Command » pour associer l'exécutable Perl aux fichiers `.cgi`. Si celle-ci n'existe pas encore, créez-en une avec une valeur par défaut de « *`chemin vers perl`* -T », par exemple « C:\Perl\bin\perl.exe -T ».

A.1.10. Existe-t-il un moyen simple de changer le nom des cookies de Bugzilla ?

Pas pour le moment.

A.1.11. Est-ce que Bugzilla tourne sous mod_perl ?

Pas pour le moment. Le travail visant à supprimer les variables globales et utiliser $cgi et DBI, progresse lentement. Tout cela est nécessaire à mod_perl (et est également une bonne chose, pour d'autres raisons). Consultez le bogue 87406 pour un aperçu de la question et des progrès.

A.2. Questions posées par les gestionnaires de l'entreprise

A.2.1. Bugzilla est-il un logiciel Web ou bien nécessite-t-il des logiciels particuliers ou un système d'exploitation spécial sur la machin ?

Il utilise le web et la messagerie électronique.

A.2.2. Bugzilla nous permet-il de définir nos propres priorités et niveaux ? Avons nous l'entière liberté de modifier les libellés des champs ainsi que leur format et le choix des valeurs admises ?

Oui. Cependant, modifier certains champs, notamment ceux liés aux états de progression des bogues, nécessite aussi d'ajuster la logique du programme pour compenser le changement.

Il n'existe pas actuellement d'interface utilisateur graphique pour ajouter des champs à Bugzilla. Vous pouvez suivre le développement de cette fonctionnalité dans le bogue 91037

A.2.3. Bugzilla propose-t-il des fonctions de rapports d'état, des métriques, des graphes, ... le genre de choses que les chefs aiment bien voir ?

Oui. Consultez la page http://bugzilla.mozilla.org/reports.cgi, vous y trouverez des échantillons de ce que Bugzilla sait faire en terme de génération de rapports et de graphiques. Une documentation complète est fournie dans Section 11, « Rapports et diagrammes » [p 84] .

Si vous n'obtenez pas les rapports désirés à partir des scripts de rapports fournis, vous pouvez inclure un paquetage professionnel pour les rapports comme par exemple Crystal Reports qui utilise ODBC. Si vous choisissez de procéder ainsi, soyez prudent car donner un accès direct à la base de données peut avoir des effets sur la sécurité. Même si vous donnez un accès en consultation seule à la base de données, il y aura contournement des éléments de sécurité des bogues de Bugzilla.

A.2.4. Existe-t-il une notification par courrier électronique ? Si oui, que voit-on lorsque l'on reçoit un courrier électronique ?

La notification par courrier électronique est paramétrable par l'utilisateur. Par défaut, l'identifiant du bogue et un résumé de son rapport accompagnent chaque notification par courrier électronique, ainsi qu'une liste des changements effectués.

A.2.5. Les utilisateurs doivent-ils disposer d'un type particulier d'application pour les courriers électroniques ?

Les courriers électroniques de Bugzilla sont envoyés en texte simple, le format de courrier le plus compatible de la planète.

Note

Si vous décidez d'utiliser les caractéristiques d'intégration de bugzilla_email pour permettre à Bugzilla d'enregistrer les réponses aux courriers électroniques avec les bogues associés, vous devrez peut-être prévenir vos utilisateurs de configurer leur logiciel de courrier électronique pour répondre aux messages dans le format où ils ont été reçus. Pour des raisons de sécurité, Bugzilla ignore les balises HTML en commentaire et si un utilisateur envoie un courrier électronique en HTML à Bugzilla, le commentaire en résultant sera assez affreux.

A.2.6. Bugzilla permet-il d'importer et d'exporter des données ? Si des personnes extérieures rédigent un rapport de bogue avec un modèle (*template*) de bogue MS Word, ce modèle peut-il être importé dans les champs correspondants ? Si je veux obtenir le résultat d'une requête et exporter cette donnée vers MS Excel est-ce possible ?

Bugzilla peut exporter des listes de bogues en HTML (par défaut), CSV ou RDF. Le lien pour le CSV se trouve à la fin de la liste de bogues au format HTML. On peut importer ce format CSV dans MS Excel ou dans un autre gestionnaire de feuilles de calculs.

Afin d'utiliser le format RDF de la liste de bogues, il est nécessaire d'ajouter un `&ctype=rdf` à l'URL. Le format RDF est censé n'être compréhensible que par une machine et, de ce fait, on considère que l'URL serait générée par programmation, c'est pourquoi il n'existe pas de lien vers ce format visible pour l'utilisateur.

Actuellement le seul script livré avec Bugzilla qui peut importer des données est `importxml.pl` qui est destiné à l'importation de données générées par le ctype XML de `show_bug.cgi` associé au déplacement de bogues. Les autres utilisations sont laissées à titre d'exercice.

Il y a aussi des scripts inclus dans le répertoire `contrib/` pour l'utilisation des courriers électroniques afin d'importer des données dans Bugzilla, mais ces scripts ne sont pas pris en charge actuellement et sont inclus dans un but pédagogique.

A.2.7. Quelqu'un a-t-il traduit Bugzilla dans une autre langue afin de l'utiliser dans d'autres pays ? Peut-on localiser Bugzilla ?

Oui. Vous trouverez plus d'informations ainsi que les traductions disponibles à l'adresse http://www.bugzilla.org/download.html#localizations. Des modèles ont été définies pour certaines interfaces d'administration (afin de faciliter la localisation) mais beaucoup ne sont encore disponibles qu'en anglais. Il peut également y avoir des problèmes si le jeu de caractères n'est pas défini. Pour plus d'informations, reportez-vous au bogue 126226.

A.2.8. Un utilisateur peut-il créer et enregistrer des rapports ? Peut-il le faire au format Word ? Au format Excel ?

Oui. Non. Oui (grâce au format CSV).

A.2.9. Existe-il des fonctions de sauvegarde ?

MySQL, la base de donnée principale de Bugzilla, permet d'effectuer des sauvegardes des données à chaud. Vous trouverez des stratégies pour traiter des problèmes de sauvegarde à l'adresse http://www.mysql.com/doc/B/a/Backup.html

A.2.10. En termes de ressources humaines, quels sont les critères pour constituer l'équipe qui installera et assurera la maintenance de Bugzilla ? Pour être plus précis, quelles sont les compétences nécessaires ? J'aimerais savoir quel genre de personnes devrons-nous embaucher. J'aimerais aussi comparer le coût de l'adoption de Bugzilla par rapport à celui de l'achat d'une solution prête à l'emploi ?

Si Bugzilla est configuré correctement dès le début, les besoins pour assurer la continuité de la maintenance sont minimes et peuvent être réalisés facilement via l'interface Web.

Les logiciels de suivi de bogues commerciaux dépassent généralement les $20,000 ou plus pour un nombre de 5 à 10 licences partagées. Des conseils pour Bugzilla sont donnés par les spécialistes, membres de la liste de diffusion. On y trouve tout de suite les réponses aux questions simples.

A.2.11. En terme de durée, que doit-on prévoir si nous décidons d'embaucher du personnel pour installer et assurer la maintenance de Bugzilla ? Est-ce que cela prend quelques heures ou quelques jours à installer et deux ou trois heures par semaine pour assurer la maintenance et l'adaptation à nos besoins ou bien s'agit-il d un processus d'installation de plusieurs semaines, plus un travail à plein temps pour une personne ou deux, etc. ?

Cela dépend complètement de ce que vous voulez y mettre. Quelqu'un avec une grande expérience de Bugzilla peut tout mettre sur pied et en état de marche en moins d'une journée, et votre installation Bugzilla peut fonctionner sans attention particulière pendant des années. Si votre stratégie Bugzilla est cruciale par rapport à votre projet d'entreprise, embauchez quelqu'un avec des compétences raisonnables en Perl, ainsi qu'une connaissance du système d'exploitation sur lequel va tourner Bugzilla, et chargez cette personne de la conduite de votre gestion de processus, la maintenance et l'adaptation au niveau régional de votre suivi de bogues.

A.2.12. Y a-t-il des droits de licence ou d'autres frais pour utiliser Bugzilla ? Faut-il prévoir d'ajouter d'autres coûts en plus de ceux liés au personnel nécessaire mentionné ci-dessus ?

Non. Bugzilla, Perl, le Template Toolkit, ainsi que tout autre logiciel supplémentaire nécessaire au fonctionnement de Bugzilla, peut être téléchargé gratuitement. MySQL, la base de données utilisée par Bugzilla, est également libre, mais vous propose, si vous appréciez la valeur de leur produit, de souscrire à un contrat d'assistance qui convient à vos besoins.

A.2.13. Nous n'aimons pas le terme « bogue » pour parler de problèmes, pouvons-nous remplacer ce terme ?

Oui! Depuis Bugzilla 2.18, remplacer le mot 'bogue' par n'importe quel mot employé dans votre entreprise est chose simple. Reportez-vous à la documentation concernant la personnalisation [Customization] pour de plus amples détails, en particulier à Section 1.5, « Modèles particuliers » [p 60] .

A.3. Questions relatives à l'administration

A.3.1. Bugzilla utilise-t-il le verrouillage des enregistrements lorsqu'il y a des accès simultanés au même bogue ? La deuxième personne est-elle informée que le bogue est en cours d'utilisation ou comment sont-ils avertis tous les deux ?

Bugzilla ne verrouille pas les entrées. Il permet la détection des accès concurrentiels, ce qui signifie qu'il avertit un utilisateur lorsqu'une validation de transaction (*commit*) est susceptible d'entrer en conflit avec une autre venant d'être effectuée par un autre utilisateur, et offre au deuxième utilisateur le choix des options pour traiter le conflit.

A.3.2. Les utilisateurs peuvent-ils utiliser le système alors qu'une sauvegarde est en cours ?

Oui. Mais les validations de transaction dans la base de données doivent attendre que les tables soient débloquées. Les bases de données Bugzilla sont généralement très petites et les sauvegardes prennent systématiquement moins d'une minute. Si votre base de données est plus vaste, vous souhaiterez probablement consulter d'autres techniques de sauvegarde, comme la duplication de base de données, ou encore sauvegarder depuis un miroir en lecture seule (à ce propos, veuillez consulter les documentations de MySQL sur le site de MySQL).

A.3.3. Comment puis-je mettre à jour le code et la base de données avec CVS ?

1. Faites une sauvegarde de votre répertoire Bugzilla et de votre base de données. Pour le répertoire de Bugzilla, il suffit de faire un **cp -rp bugzilla bugzilla.bak**. Pour la base de données, il existe un certain nombre d'options —consultez la documentation MySQL et choisissez celle qui vous convient le mieux (le moyen le plus simple est de faire une copie physique de la base de données sur le disque dur, mais pour cela vous devez faire éteindre le serveur de la base de données afin d'éviter tout risque de perte de données).

2. Placez-vous dans le répertoire de Bugzilla.

3. Lancez **cvs -q update -AdP** si vous voulez obtenir la toute dernière version ou **cvs -q update -dP -rÉTIQUETTE** si vous désirez une version donnée (auquel cas, vous devrez remplacer ÉTIQUETTE par une étiquette CVS (*tag*) comme `BUGZILLA-2_16_5`).

   Si vous n'avez fait aucun changement local, ça devrait être parfait. Si vous avez modifié des choses dans votre copie locale, vérifiez la présence de résultats notés C en sortie de CVS. Si vous obtenez des lignes commençant par C, cela signifie qu'il existe des conflits entre vos changements locaux et ceux qui se trouvent sur le CVS. Vous devrez régler cela à la main avant de pouvoir continuer.

4. Après avoir résolu tout conflit généré par l'opération de mise à jour du CVS, **./checksetup.pl** se chargera de mettre à jour la base de données pour vous et effectuera tout autre changement nécessaire pour que la nouvelle version puisse fonctionner.

   Avertissement

   Une fois que vous avez exécuté **checksetup.pl**, le seul moyen de revenir en arrière est de restaurer les sauvegardes de la base de données. Vous ne pourrez pas ramener le système à sa version initiale de manière propre dans la plupart des cas.

Lisez également les instructions de Exemple 3.1, « Mise à niveau par le CVS » [p 50] .

A.3.4. Comment faire pour que les bogues aient un statut « NON-CONFIRMÉ » (*UNCONFIRMED*) ?

Pour utiliser le statut « NON-CONFIRMÉ », le paramètre `usevotes` doit être activé (*ON*). Vous devez ensuite accéder à la page `editproduct.cgi` et régler le « nombre de voix qu'un bogue de ce produit doit recevoir afin de sortir automatiquement du statut NON-CONFIRMÉ » (*Number of votes a bug in this product needs to automatically get out of the UNCONFIRMED state*) à une valeur non-nulle. (Vous devrez faire de même pour chaque produit pour lequel vous voudrez utiliser le statut NON-CONFIRMÉ). Si en fait vous ne voulez pas que les utilisateurs puissent voter pour les bogues de ce produit, laissez le « nombre maximal de voix par personne » (*Maximum votes per person*) à zéro.

Le découplage du statut NON-CONFIRME par rapport au paramètre `usevotes` pour de prochaines versions de Bugzilla est en cours. Vous pouvez suivre le débat et l'avancement de ce travail sur le bogue 162060.

A.4. La sécurité de Bugzilla

A.4.1. Comment désactiver complètement la sécurité dans MySQL si elle me cause des problèmes (j'ai suivi les instructions de la partie installation de ce guide) ?

Exécutez MySQL comme suit : **mysqld --skip-grant-tables**. N'oubliez pas que cela rend MySQL aussi sûr que de scotcher un billet de 100 euros par terre dans les toilettes d'un stade de foot pour ne pas se le faire piquer.

Avertissement

On n'insistera jamais assez là-dessus : on ne vous conseille pas de le faire. Veuillez consulter Section 2, « MySQL » [p 53] de ce guide ainsi que la documentation de MySQL afin de trouver de meilleures solutions.

A.4.2. Y a-t-il des problèmes de sécurité avec Bugzilla ?

Le code de Bugzilla a subi un audit de sécurité relativement complet, et les scripts CGI de présentation pour les utilisateurs sont exécutés avec l'option `taint` de Perl. Cependant il est recommandé d'examiner attentivement les permissions de votre installation de Bugzilla et de suivre les recommandations en matière de sécurité dans le guide Bugzilla.

A.5. Les courriers électroniques de Bugzilla

A.5.1. Un de mes utilisateurs ne veut plus recevoir de courriers électroniques de Bugzilla. Comment les arrêter complètement pour cet utilisateur ?

L'utilisateur peut ordonner à Bugzilla d'arrêter de lui envoyer des courriels en décochant toutes les cases de la page Edit prefs+Email settings (depuis la version 2.18, cette opération est facilitée par l'ajout d'un bouton Disable All Mail, *désactiver tous les envois de courriels*) Autrement, vous pouvez ajouter leur adresse électronique au fichier `data/nomail` (une seule adresse par ligne). Cette option passe outre leurs préférences personnelles et ils ne recevront plus jamais de courriels de Bugzilla.

A.5.2. Je suis en train d'évaluer/tester Bugzilla et je ne veux pas qu'il envoie des courriels à d'autres personnes que moi. Comment faire ?

Pour désactiver l'envoi de courriels, réglez le paramètre

```
$enableSendMail
```

à `0` soit dans `Bugmail.pm` (pour les versions 2.18 et supérieures), soit dans `processmail` (jusqu'à la version 2.16.x)

Note

Jusqu'à une version 2.16.x, modifier `$enableSendMail` affectera seulement les courriels concernant un bogue; les courriels relatifs aux changements de mot de passe, d'adresse électronique, aux importations de bogues, aux changements d'options (*flags*), etc. seront toujours envoyés. Depuis la sortie de la version définitive 2.18, néanmoins, l'étape expliquée ci dessus désactivera tous les envois de courriels de Bugzilla quel qu'en soit le but.

Pour que les courriels relatifs aux bogues (et seulement ceux-ci) soient redirigés vers vous en lieu et place de leurs destinataires prévus, laissez `$enableSendMail` tranquille ; à la place, éditez le paramètre `newchangedmail` comme suit :

- Remplacez « To: » par « X-Real-To: »
- Remplacez « Cc: » par « X-Real-CC: »
- Ajoutez une ligne « To: *`votre_adresse_de_courriel`* »

A.5.3. Je souhaite que whineatnews.pl génère une réclamation pour autre chose que pour signaler de nouveaux bogues ou des bogues réouverts. Comment faire ?

Pour d'anciennes versions de Bugzilla, vous devriez pouvoir appliquer le correctif de Klaas Freitag pour « whineatasigned », que l'on trouve dans le bogue 6679. Remarquez que ce correctif a été réalisé en 2000, un certain travail sera vraisemblablement nécessaire afin de l'appliquer proprement sur n'importe quelle version de Bugzilla plus récente, mais vous pouvez l'utiliser comme point de départ.

Il est prévu d'inclure à Bugzilla 2.20 une version mise à jour (et plus approfondie) de cette fonctionnalité; reportez vous au bogue 185090 pour suivre le débat ou pour obtenir des correctifs plus à jour si vous ne pouvez vraiment pas attendre.

A.5.4. Comment régler l'interface de courrier électronique pour soumettre/changer des bogues par courrier électronique ?

Vous trouverez un fichier `README.mailif` mis à jour dans le répertoire `contrib/` de votre distribution Bugzilla qui vous guidera dans la configuration.

A.5.5. Les courriers électroniques provenant de Bugzilla mettent une éternité pour me parvenir. C'est vraiment très lent. Que se passe-t-il ?

Si vous utilisez sendmail, essayez d'activer `sendmailnow` dans `editparams.cgi`. Pour des versions plus anciennes de sendmail, on peut améliorer significativement les performances de l'interface utilisateur (au prix d'un retard dans l'envoi de courriel) en réglant ce paramètre sur `off`. Pour les sites avec une version 8.12 (ou supérieure) de sendmail, il vaut mieux le laisser sur `on`, puisqu'ils ne bénéficieront alors d'aucun gain de performance.

Si vous utilisez un gestionnaire de courrier électronique (ou MTA, *Mail Transfer Agent*) différent, assurez-vous que les options présentes dans `Bugzilla/Bugmail.pm`, ainsi que dans tous les autres endroits d'où est appelé sendmail, correspondent à l'emplacement de votre MTA.

A.5.6. Comment se fait-il que les courriers électroniques concernant les modifications sur Bugzilla ne me parviennent pas ?

Assurez vous que vous n'avez pas désactivé les courriers électroniques dans vos préférences d'utilisateur. Vérifiez que Bugzilla est capable d'envoyer des courriers électroniques en cliquant sur le lien « Connexion » (*Log In*) de votre installation Bugzilla puis en cliquant sur le bouton « Envoyez moi un mot de passe par courrier électronique » (*E-mail me a password*) après avoir saisi votre adresse électronique.

Si vous ne recevez pas de courrier électronique de Bugzilla, il y a des chances que vous n'ayez pas sendmail dans `/usr/lib/sendmail`. Assurez-vous que sendmail se trouve à l'intérieur ou qu'il existe un lien symbolique pointant vers lui dans `/usr/lib/sendmail`.

Si vous utilisez un autre MTA que sendmail, le paramètre `sendmailnow` doit être réglé sur `on`, sinon aucun courriel ne sera envoyé.

A.6. La base de données de Bugzilla

A.6.1. Je pense que ma base de données doit être altérée, ou bien qu'elle contient des enregistrements non valides. Que faire ?

Exécutez l'utilitaire de « bilan de santé » (`sanitycheck.cgi`) via votre navigateur internet pour voir ! Si il se termine sans erreur, vous n'avez *probablement* aucun problème. S'il ne se termine pas bien (c'est à dire avec des messages en rouge), certaines choses peuvent être réparées par Bugzilla et d'autres non. Si la réparation est impossible, j'espère pour vous que vous avez l'habitude des commandes d'administration d'une base de données mysql ou bien que vous avez installé un autre moyen d'administrer votre base de données. Le bilan de santé, bien qu'étant un bon test de base pour vérifier l'intégrité de votre base de données, ne remplace en aucun cas une administration compétente de la base de données ni n'évite la suppression de données. Il n'est pas exhaustif et a été créé pour effectuer une vérification de base sur les problèmes les plus communs dans des bases de données Bugzilla.

A.6.2. Je veux éditer à la main des enregistrements de ma base de données. Comment faire ?

Il n'existe pas de moyen de faire cela dans Bugzilla lui-même. De plus ce n'est généralement pas intelligent de le faire si l'on ne sait pas exactement ce qu'on veut faire. Cependant si vous connaissez le SQL, vous pouvez utiliser **mysql** en ligne de commande pour insérer, supprimer et modifier les informations d'une table à la main. Des clients graphiques plus intuitifs sont également disponibles. Les favoris de l'équipe Bugzilla sont phpMyAdmin et MySQL Control Center

Gardez à l'esprit que les sauvegardes sont vos amies. Tout le monde fait des erreurs, et il est agréable d'avoir un filet de sécurité dans le cas où vous sèmeriez la pagaille quelque part. Pensez à utiliser **mysqldump** pour faire un double de votre base de données avant de la modifier à la main.

A.6.3. Je pense que j'ai correctement réglé les permissions de MySQL mais Bugzilla ne parvient toujours pas à se connecter.

Essayez d'exécuter MySQL à partir de son fichier exécutable : **mysqld --skip-grant-tables**. Cela vous permettra de complètement contourner les permissions sur les tables qui sont la cause de votre frustration. Si dans ce cas Bugzilla est capable de se connecter, vous devez vérifier que vous avez donné les bonnes permissions au couple utilisateur-mot de passe défini dans `localconfig`.

Avertissement

Exécuter MySQL avec cette option dans la ligne de commande n'est pas sûr du tout et doit être uniquement effectué dans le cadre d'une démarche de dépannage lorsque l'on n'est pas connecté au réseau extérieur.

A.6.4. Comment synchroniser les informations sur les bogues entre plusieurs bases de données ?

En fait, vous pouvez synchroniser ou déplacer les bogues. La synchronisation s'opèrera uniquement dans un sens ; vous avez la possibilité de créer une copie de la base en lecture seule sur un site et la faire mettre à jour à partir de la base de données principale à intervalles réguliers.

MySQL intègre des fonctionnalités de synchronisation dans les dernières versions. Ce serait génial si quelqu'un se penchait sur les possibilités dans ce domaine et proposait un compte rendu sur le forum expliquant comment synchroniser efficacement deux installations de Bugzilla.

Si vous avez simplement besoin de transférer des bogues d'un Bugzilla à un autre, jetez un coup d'œil au script `move.pl` dans la distribution de Bugzilla.

A.7. Bugzilla et Win32

A.7.1. Quel est le moyen le plus facile d'utiliser Bugzilla sur Win32 (Win98+/NT/2K) ?

Enlevez Windows. Installez Linux. Installez Bugzilla. Le chef ne verra jamais la différence B^).

Plus sérieusement, un des objectifs majeurs de ce grand cru 2.18 était de faire fonctionner Bugzilla facilement sous Windows. Si les composants nécessaires sont présents (perl, un serveur web, un MTA, etc.), l'installation de Bugzilla sur une station Windows ne devrait pas être plus difficile que sur une autre plateforme. Comme pour toute installation, nous vous recommandons de suivre complètement et avec beaucoup d'attention les instructions d'installation de Section 4.1, « Microsoft Windows » [p 25] .

Ce faisant, n'oubliez pas de consulter cet excellent guide écrit par Byron Jones intitulé Installing Bugzilla on Microsoft Windows (Comment installer Bugzilla sous Microsoft Windows). Merci, Byron !

A.7.2. Existe-t-il un équivalent de « Bundle::Bugzilla » pour Win32 ?

Pas pour le moment. Bundle::Bugzilla simplifie énormément l'installation de Bugzilla sur les systèmes UNIX. Si quelqu'un se porte volontaire pour réaliser un bundle PPM [1 [p 108]] qui convient à Win32, une telle initiative sera appréciée.

A.7.3. Des fichiers CGI se terminent prématurément avec une erreur « something.cgi is not a valid Windows NT application » (quelquechose.cgi n'est pas une application Windows NT valide). Pourquoi ?

En fonction du serveur web que vous utilisez, vous devez configurer celui-ci pour traiter les fichiers `*.cgi` comme des scripts CGI. Pour IIS, vous pouvez faire cela en ajoutant *.cgi dans la mise en correspondance des applications avec <chemin>\perl.exe %s %s comme exécutable.

Microsoft donne quelques indications à ce sujet, que voici :

> « Configurer la mise en correspondance des applications. Dans l'ISM, liez l'extension du ou des fichier(s) script à l'exécutable de l'interpréteur du script. Par exemple, vous devez faire correspondre l'extension .py à Python.exe, l'exécutable de l'interpréteur de scripts Python. Notez que pour l'interpréteur de scripts Perl ActiveState, l'extension.pl est associée à PerlIS.dll par défaut. Si vous voulez changer l'association pour .pl par perl.exe, vous devez changer la correspondance d'applications. Dans la correspondance, vous devez ajouter deux caractères "pourcent" (%) à la fin du chemin jusqu'à perl.exe, comme indiqué dans cet exemple : **c:\perl\bin\perl.exe %s %s** »

A.7.4. J'ai des problèmes avec les modules de perl pour NT qui n'arrivent pas à communiquer avec la base de données.

Vos modules sont peut-être obsolètes ou inappropriés. Essayez ce qui suit :

1. Allez sur http://www.activestate.com/ActivePerl.
2. Téléchargez ActivePerl.
3. Lancez votre invite de commande.
4. Saisissez **ppm**
5. `PPM>` **install DBI DBD-mysql GD**

Je pense que TimeDate et Date::Dumper sont inclus dans activeperl. Vous pouvez aller vous renseigner sur le site d'ActiveState à propos des packages pour l'installation via PPM : http://www.activestate.com/Packages/.

A.8. Utilisation de Bugzilla

A.8.1. Comment puis-je changer mon nom d'utilisateur (adresse de courrier électronique) dans Bugzilla ?

Nouveau dans la version 2.16 : allez dans la section compte dans les préférences. Vous recevrez un courrier électronique aux deux adresses pour confirmation.

A.8.2. La page pour les requêtes est très déroutante. N'y a-t-il pas un moyen plus simple d'effectuer une requête ?

L'interface a été simplifiée par un concepteur d'interfaces dans la version 2.16. Les suggestions d'amélioration sont les bienvenues, mais nous ne sacrifierons pas la puissance à la simplicité.

Depuis la version 2.18, une recherche « plus simple » est également disponible. En haut de la page de recherche il y a deux liens : « Recherche avancée » (*Advanced search*) vous conduit à la page de recherche la plus puissante, que vous connaissez bien, mais aussi la plus complexe. Le lien « Chercher un bogue en particulier » (*Find a specific Bug*) vous amène sur une page très simplifiée où vous pouvez choisir un produit et un statut (ouvert *OPEN*, fermé *CLOSED* ou les deux), vous pourrez ensuite entrer les mots qui apparaissent dans le bogue que vous cherchez. Cette recherche ira fouiller les champs « Résumé » (*Summary*) et « Commentaires » (*Comments*), et renverra une liste de bogues triés, de sorte que les bogues comportant le plus de coïncidences/correspondances se trouvent en début de liste.

Note

Les correspondances issues du champ Summary prendront le pas sur celles du champ Comments et les bogues comportant des correspondances dans le champ Summary seront placés plus haut dans la liste de bogue, même si un bogue classé en dessous présente plus de correspondances dans le champ commentaires.

Bugzilla utilise un cookie afin de se souvenir quelle version d'une page vous avez visitée en dernier et ressortira cette page lors de votre recherche suivante. La page par défaut pour les nouveaux utilisateurs (ou après une mise à jour) est la page de recherche « simple ».

A.8.3. Le comportement du bouton Accept dans le formulaire Show Bug me perturbe. Pourquoi le bogue ne m'est-il pas affecté lorsque je l'ai accepté ?

Le comportement actuel convient à bugzilla.mozilla.org ainsi qu'à la plupart des utilisateurs. Si vous désirez changer ce comportement malgré tout, vous pouvez utiliser un des correctifs suivants :

> Le bogue 35195 vise à ajouter une case à cocher « ... et accepter les bogues » (*... and accept the bug*) à l'interface utilisateur. Deux correctifs lui sont attachés : le correctif 8029 fut créé à la base pour Bugzilla 2.12, tandis que le correctif 91372 en est une version mise à jour pour Bugzilla 2.16.
>
> Le bogue 37613 fournit lui aussi deux correctifs (pour Bugzilla 2.12) : l'un pour ajouter une option « Admettre le bogue » (*Take Bug*), l'autre afin de réassigner automatiquement le bogue lorsqu'il est accepté.

Ces correctifs commencent maintenant à dater un peu et ne peuvent être appliqués directement, mais ils sont suffisamment simples pour montrer comment Bugzilla peut être personnalisé et mis à jour afin de correspondre à vos besoins.

A.8.4. Je ne peux rien inscrire dans la base de données via le lien « Create Attachment ». Où est l'erreur ?

La cause la plus probable est que vous utilisez un très vieux navigateur ou un navigateur incompatible avec le téléchargement de fichiers vers le serveur via POST. Téléchargez la dernière version de votre navigateur internet préféré pour effectuer les téléchargements vers le serveur correctement.

A.8.5. Comment puis-je changer un mot-clé dans Bugzilla, à partir du moment où plusieurs bogues l'utilisent ?

Dans l'interface administrateur de Bugzilla, éditez le mot-clé — vous pouvez ainsi remplacer l'ancien mot-clé par le nouveau. Cela causera un problème avec la mémoire cache des mots-clés. Exécutez **sanitycheck.cgi** pour le résoudre.

A.8.6. Je ne peux pas fermer les bogues depuis la page « Change Several Bugs at Once ». Pourquoi ?

La réponse est simple : si, vous pouvez le faire.

Le mécanisme derrière la page vérifie chaque bogue dans la liste pour déterminer les changements d'états autorisés, et c'est seulement ensuite qu'elle vous montre les commandes permettant de réaliser les actions applicables à *chaque* bogue de la liste. La raison est que si vous essayez d'effectuer une action non autorisée sur un bogue, le processus entier va s'arrêter net et tous les changements effectués après celui qui a échoué vont eux aussi échouer, ce qui n'est pas satisfaisant du tout, et c'est pourquoi la page ne vous propose même pas cette option.

En français courant, cela signifie que chaque bogue de la page doit être soit déjà résolu (*RESOLVED*) soit déjà vérifié (*VERIFIED*), afin de déclarer plusieurs bogues clos (*CLOSED*) ; si ce n'est pas le cas, l'option permettant de fermer les bogues n'apparaîtra pas sur la page.

Ceci repose sur la logique suivante : si vous essayez de fermer un bogue qui n'a pas été vérifié, la mise à jour du bogue échouera lamentablement (et ainsi détruira tout changement suivant dans la liste pendant l'opération globale de changement), ce qui ne vous donne donc même pas le choix.

A.9. Bidouiller Bugzilla

A.9.1. Quel style de programmation suis-je censé utiliser pour la création de modèles ?

Gerv et Myk suggèrent d'utiliser une indentation de deux espaces, avec les sections de code imbriquées sur leur propre ligne, en ligne avec les balises externes.

```
<fred>
[% IF foo %]
  <bar>
  [% FOREACH x = barney %]
    <tr>
      <td>
        [% x %]
      </td>
    <tr>
  [% END %]
[% END %]
</fred>
```

Myk recommande en outre d'activer PRE_CHOMP dans l'initialisation de modèles afin d'éviter de surcharger le code HTML avec des espaces inutiles.

Veuillez noter qu'il y a de nombreux avis divergents à ce sujet, et les templates existant dans Bugzilla respectent à la fois la convention ci-dessus et la convention à 4 espaces. Les deux sont acceptables ; on préfère celle que nous présentons ici.

A.9.2. Quels sont les bugs de Bugzilla actuellement ?

Consultez ce lien pour voir les bogues actuels ou les demandes d'amélioration de Bugzilla.

Vous pouvez consulter des bogues marqués pour la version 2.18.1 ici. Cette liste contient les bogues de la version 2.18.1 qui ont déjà été corrigés et consignés dans le CVS. Vous trouverez sur la page du projet Bugzilla pour des informations sur la manière de trouver les sources actuelles sur le CVS pour pouvoir récupérer ces corrections.

A.9.3. Comment donner à la priorité par défaut une valeur nulle ? Par exemple, avoir « --- » comme valeur par défaut au lieu de « P2 » ?

Ce point est bien documenté dans le bogue 49862. En fin de compte, ce n'est pas plus compliqué que d'ajouter le champ de priorité « --- » à l'endroit approprié de votre fichier `localconfig`, de relancer **checksetup.pl**, puis de changer la priorité par défaut dans votre navigateur au moyen de **editparams.cgi**.

A.9.4. Quelle est la meilleure façon de soumettre des correctifs ? Quelles sont les directives à suivre ?

1. Ajoutez un bogue sur bugzilla.mozilla.org concernant le logiciel « Bugzilla ».

2. Téléchargez votre correctif vers le serveur en tant que unified diff (en utilisant **diff -u** sur les *sources actuelles* récupérées sur CVS), ou en tant que nouveau fichier source en cliquant sur le lien Create a new attachment dans la page de bogue que vous venez juste de créer, et ajoutez dans la description du bogue que vous avez soumis à l'étape 1 toute description que vous pouvez faire concernant des changements dans la base de données. N'oubliez surtout pas de cocher la case Patch pour indiquer que le texte que vous envoyez est un correctif !

3. Faites part de l'arrivée de votre correctif et de l'URL associée (http://bugzilla.mozilla.org/show_bug.cgi?id=XXXXXX) afin d'en discuter dans le forum (netscape.public.mozilla.webtools). Vous verrez, les réactions sur les implications de votre correctif seront vraiment bonnes et assez rapides, ce qui nous donnera aussi une idée de la manière dont seraient reçus les changements.

4. S'il passe les contrôles avec un minimum de modifications, la personne à qui le bogue est assigné chez Bugzilla sera chargée d'assurer que le programme de correction soit consigné sur CVS.

5. Réjouissez-vous de la gloire d'avoir participé à la réalisation du logiciel de suivi de bogues le plus réputé de la planète :)

[1 [p 103]] NdT : module pour le gestionnaire de packetages de Perl

Annexe B. Résolution des problèmes

Table des matières

Cette section propose des solutions aux problèmes d'installation de Bugzilla. Si aucune des têtes de chapitres ne semble correspondre à votre problème, lisez les conseils généraux.

1. Conseils généraux

Normalement, si `checksetup.pl` ne parvient pas à s'exécuter complètement, il vous explique ce qui ne va pas et comment régler le problème. Si vous n'arrivez pas à vous en sortir, ou si le logiciel fait preuve de mutisme, publiez ces erreurs sur le site de diffusion netscape.public.mozilla.webtools

Si vous avez franchi avec succès toutes les étapes de Section 1, « Installation » [p 10] (Installation) et Section 2, « Configuration » [p 16] (Configuration) mais que l'accès à l'URL de Bugzilla ne fonctionne pas, la première chose à vérifier est le journal d'erreur de votre serveur Web. Dans le cas d'Apache, il se situe souvent dans `/etc/logs/httpd/error_log`. Les messages d'erreur que vous y trouverez seront peut-être suffisamment explicites pour vous permettre de diagnostiquer et corriger le problème. Si ce n'est pas le cas, regardez ce qui est dit ci-dessous sur certaines erreurs fréquemment rencontrées. Si cela ne vous aide toujours pas, publiez ces erreurs sur le groupe de diffusion.

2. Le serveur Web Apache ne m'ouvre pas les pages de Bugzilla

Après avoir lancé **checksetup.pl** deux fois, lancez **testserver.pl http://votre_site.votredomaine/votre_url** afin de confirmer que votre serveur Web est correctement configuré pour Bugzilla.

```
bash$ ./testserver.pl http://landfill.bugzilla.org/bugzilla-tip
TEST-OK Webserver is running under group id in $webservergroup.
TEST-OK Got ant picture.
TEST-OK Webserver is executing CGIs.
TEST-OK Webserver is preventing fetch of http://landfill.bugzilla.org/bugzilla-tip/localconfig.
```

3. J'ai installé un module Perl mais `checksetup.pl` affirme qu'il n'est pas installé !

Cela peut avoir l'une des deux causes suivantes :

1. Vous avez deux versions de Perl sur votre machine. Vous installez les modules sous l'une, alors que Bugzilla en utilise une autre. Exécutez à nouveau les commandes CPAN (ou recompilez manuellement) en donnant le chemin complet vers Perl depuis le répertoire de `checksetup.pl`. Ainsi vous serez sûr que les modules sont installés au bon endroit.

2. Les permissions de répertoires des bibliothèques ne sont pas paramétrées correctement. Ils doivent, à tout le moins, être lisibles par l'utilisateur ou le groupe serveur Web. Il est conseillé qu'ils soient accessibles à tous.

4. Bundle::Bugzilla met à niveau Perl à la version 5.6.1

Exécutez la commande **perl -MCPAN -e 'install CPAN'** pour voir et continuez.

Certaines versions plus anciennes des outils CPAN étaient un peu naïves quand elles mettaient à jour des modules Perl. Quand des modules entraient dans la distribution Perl 5.6.1, CPAN pensait que le meilleur moyen de les garder à jour était de récupérer la distribution Perl elle-même et de la reconstruire. Inutile de vous dire que cela a causé un mal de tête à presque tout le monde. Une mise à niveau vers la nouvelle version de CPAN grâce à la commande ci-dessus devrait régler le problème.

5. « La préparation de DBD::Sponge::db a échoué » [DBD::Sponge::db prepare failed]

Le message suivant peut apparaître à cause d'un bogue dans DBD::mysql (sur lequel l'équipe Bugzilla n'a aucun contrôle) :

```
DBD::Sponge::db prepare failed: Cannot determine NUM_OF_FIELDS at D:/Perl/site/lib/DBD/mysql.pm line 248.
 SV = NULL(0x0) at 0x20fc444
 REFCNT = 1
 FLAGS = (PADBUSY,PADMY)
```

Pour régler le problème, éditez le fichier `<chemin-vers-perl>/lib/DBD/sponge.pm` dans votre répertoire d'installation de Perl et remplacez

```
my $numFields;
if ($attribs->{'NUM_OF_FIELDS'}) {
    $numFields = $attribs->{'NUM_OF_FIELDS'};
} elsif ($attribs->{'NAME'}) {
    $numFields = @{$attribs->{NAME}};
```

par

```
my $numFields;
if ($attribs->{'NUM_OF_FIELDS'}) {
    $numFields = $attribs->{'NUM_OF_FIELDS'};
} elsif ($attribs->{'NAMES'}) {
    $numFields = @{$attribs->{NAMES}};
```

(notez le S ajouté à NAME.)

6. « Impossible d’exécuter chdir... » [cannot chdir(/var/spool/mqueue)]

Si vous installez Bugzilla sur SuSE Linux ou sur une autre distribution avec des options de sécurité « paranoïaques », le script checksetup.pl peut se bloquer avec l’erreur suivante :

```
cannot chdir(/var/spool/mqueue): Permission denied
```

Cette erreur se produit parce que votre répertoire `/var/spool/mqueue` a des droits insuffisants : `drwx------`. Tapez **chmod 755 `/var/spool/mqueue`** sous root pour régler le problème. Cela permettra à n’importe quel processus s’exécutant sur votre machine de *lire* le répertoire `/var/spool/mqueue`.

7. « Votre fournisseur n’a pas défini... » [Your vendor has not defined Fcntl macro O_NOINHERIT]

Cette erreur est provoquée par un bogue dans la version de File::Temp™ distribuée avec Perl 5.6.0. De nombreuses variantes légèrement différentes de cette erreur ont été signalées :

```
Your vendor has not defined Fcntl macro O_NOINHERIT, used
at /usr/lib/perl5/site_perl/5.6.0/File/Temp.pm line 208.

Your vendor has not defined Fcntl macro O_EXLOCK, used
at /usr/lib/perl5/site_perl/5.6.0/File/Temp.pm line 210.

Your vendor has not defined Fcntl macro O_TEMPORARY, used
at /usr/lib/perl5/site_perl/5.6.0/File/Temp.pm line 233.
```

Beaucoup d’utilisateurs ont signalé qu’une mise à niveau vers la version 5.6.1 et suivantes a réglé leur problème. Une solution moins définitive consiste à appliquer le correctif suivant, également disponible sous forme d’un correctif.

```
--- File/Temp.pm.orig   Thu Feb  6 16:26:00 2003
+++ File/Temp.pm        Thu Feb  6 16:26:23 2003
@@ -205,6 +205,7 @@
     # eg CGI::Carp
     local $SIG{__DIE__} = sub {};
     local $SIG{__WARN__} = sub {};
+    local *CORE::GLOBAL::die = sub {};
     $bit = &$func();
     1;
   };
@@ -226,6 +227,7 @@
     # eg CGI::Carp
     local $SIG{__DIE__} = sub {};
     local $SIG{__WARN__} = sub {};
+    local *CORE::GLOBAL::die = sub {};
     $bit = &$func();
     1;
   };
```

8. On est constamment obligés de se reconnecter

La cause la plus probable est que le paramètre « cookiepath » n'est pas correctement réglé dans la configuration de Bugzilla. Ça peut s'arranger (si vous êtes administrateur Bugzilla) sur la page editparams.cgi par le web.

La valeur du paramètre cookiepath doit être précisément le répertoire contenant votre installation de Bugzilla, *telle que la voit le navigateur Internet d'un utilisateur*. Les slash de début et de fin sont obligatoires. Vous pouvez également paramétrer le cookiepath vers n'importe quel répertoire parent du répertoire Bugzilla (comme « / », le répertoire racine). Mais vous ne pouvez pas indiquer un chemin qui ne correspond pas au moins partiellement, car cela ne marchera pas. Ce que l'on fait là, en fait, c'est de limiter l'action du navigateur utilisateur au renvoi de cookies uniquement dans ce répertoire.

Comment savoir si vous avez besoin de votre répertoire Bugzilla particulier ou du site complet ?

Si vous n'avez qu'un seul Bugzilla installé sur votre serveur, que cela ne vous dérange pas d'avoir d'autres applications sur le même serveur et qu'il soit capable de voir les cookies (ça pourrait être fait exprès si vous avez d'autres outils sur votre site qui partagent l'authentification avec Bugzilla), vous n'aurez qu'a régler le cookiepath à « / », ou à un répertoire suffisamment élevé dans l'arborescence afin que toutes les applications concernées puissent voir les cookies.

Exemple B.1. Exemples de paires urlbase/cookiepath pour le partage des cookies d'ouverture de session

> urlbase : http://bugzilla.mozilla.org/ cookiepath : / urlbase : http://tools.mysite.tld/bugzilla/ mais vous avez http://tools.mysite.tld/someotherapp/ partageant l'authentification avec Bugzilla cookiepath : /

D'un autre côté, si vous avez plus d'une version de Bugzilla installée sur votre serveur (quelques utilisateurs le font; nous le faisons pour landfill), il faut que le cookiepath soit suffisamment restreint afin que les différents Bugzilla ne confondent pas leurs cookies avec ceux d'un autre.

Exemple B.2. Exemples de paires urlbase/cookiepath pour la restriction du cookie d'ouverture de session

> urlbase : http://landfill.bugzilla.org/bugzilla-tip/ cookiepath : /bugzilla-tip/ urlbase : http://landfill.bugzilla.org/bugzilla-2.16-branch/ cookiepath : /bugzilla-2.16-branch/

Si vous aviez paramétré votre cookiepath à « / » auparavant et que vous devez le régler à un niveau plus restrictif (c'est à dire « /bugzilla/ »), vous pouvez effectuer cela de manière sûre sans demander aux utilisateurs de supprimer dans leur navigateur Internet leurs cookies relatifs à Bugzilla (ceci est vrai depuis Bugzilla 2.18 et Bugzilla 2.16.5).

9. Certains utilisateurs sont constamment obligés de se reconnecter

Tout d'abord, assurez vous que les cookies sont activés dans le navigateur de l'utilisateur.

Si cela ne règle pas le problème, il se peut que le fournisseur d'accès Internet de l'utilisateur implémente un serveur proxy tournant. Cela provoque un changement périodique de l'adresse IP réelle de l'utilisateur (l'adresse d'où provient l'utilisateur du point de vue du serveur Bugzilla). Puisque les cookies de Bugzilla sont liés à une adresse IP spécifique, chaque fois que cette adresse réelle change, l'utilisateur devra se connecter à nouveau.

Si vous utilisez la version 2.18, il y a un paramètre appelé « loginnetmask » que vous pouvez utiliser afin de régler le nombre de bits que nécessite l'adresse IP de l'utilisateur lors de l'authentification des cookies. Si vous indiquez un nombre inférieur à 32, une case à cocher sera mise à disposition de l'utilisateur sur l'écran de connexion pour « Restreindre cet accès à mon adresse IP » [Restrict this login to my IP address], case cochée par défaut. Si l'utilisateur laisse la case cochée, Bugzilla se comportera de la même manière qu'auparavant, exigeant que l'adresse IP corresponde exactement afin de rester connecté. Si l'utilisateur décoche la case, alors seule la partie gauche de son adresse IP (à hauteur du nombre de bits que vous avez spécifié dans le paramètre) devra correspondre pour rester connecté.

10. `index.cgi` ne s'affiche pas à moins qu'il ne soit spécifié dans l'URL

Il vous faut probablement paramétrer votre serveur Web de sorte qu'il considère la page index.cgi comme une page d'index.

Si vous utilisez Apache, vous pouvez faire ceci en ajoutant `index.cgi` à la fin de la ligne `DirectoryIndex` comme indiqué dans Section 2.4.1, « httpd™ d'Apache » [p 19] .

Annexe C. Contrib

Table des matières

Il existe un certain nombre d'extensions non officielles pour Bugzilla dans le répertoire `$BUGZILLA_ROOT/contrib/`. Cette section leur tient lieu de documentation.

1. L'interface de recherche en ligne de commande

Il existe une série d'utilitaires Unix pour faire des recherches dans Bugzilla depuis la ligne de commande. On les trouve dans le répertoire `contrib/cmdline`. Il y a trois fichiers : `query.conf`, `buglist` et `bugs`.

> **Avertissement**
>
> Ces fichiers datent d'avant le travail de création de modèles effectué dans la version 2.16, et n'ont pas été mis à jour.

`query.conf` contient la correspondance entre les options de ligne de commande et les noms de champs ainsi que les types de comparaison. Les noms d'option entre guillemets sont rassemblés de telle manière qu'il soit facile d'éditer ce fichier. Les commentaires (#) n'ont aucun effet ; assurez vous que ces lignes ne contiennent pas « d'option » entre guillemets.

`buglist` est un script shell dont le rôle est d'adresser une requête à Bugzilla et d'écrire le résultat sous forme d'une page HTML sur la sortie standard. Il prend en compte à la fois les options abrégées (comme « -Afoo » ou « -Rbar ») et les options en format long (comme « --assignedto=foo » ou « --reporter=bar »). Si le premier caractère d'une option n'est pas le signe « - », elle est considérée comme étant préfixée de « --default= ».

La liste en colonnes est extraite du contenu de la variable d'environnement COLUMNLIST. C'est équivalent à l'option « Change columns » lorsque vous inscrivez les bogues dans buglist.cgi. Si vous avez déjà utilisé Bugzilla, lancez grep sur COLUMNLIST dans vos fichiers de cookies pour voir le contenu actuel de COLUMNLIST.

`bugs` est un simple script qui appelle `buglist` et extrait les numéros de bogues à partir du résultat en sortie. Le fait d'ajouter le préfixe « http://bugzilla.mozilla.org/buglist.cgi?bug_id= » transforme la liste de bogues en liens qui fonctionnent si des bogues sont trouvés. Compter les bogues est une chose aisée. Récupérez le résultat avec un tube vers **sed -e 's/,/ /g' | wc | awk '{printf $2 "\n"}'**.

Akkana Peck affirme avoir de bons résultats en plaçant un tube au niveau du résultat de `buglist` vers **w3m -T text/html -dump**

2. L'outil « envoi de courriel de bogue non-envoyé » en ligne de commande

Au sein du répertoire `contrib` existe un utilitaire portant le nom explicite (bien que compact) de `sendunsentbugmail.pl`. Le but de ce script est simplement d'émettre tout courriel relatif à un bogue qui aurait déjà dû être envoyé mais qui, pour une raison ou une autre, ne l'a pas été.

Pour accomplir cette tâche, `sendunsentbugmail.pl` utilise le même mécanisme que le script `sanitycheck.cgi`; il parcourt entièrement la base de donnée en cherchant les bogues présentant des changements effectués il y a plus de 30 minutes, là où il ne trouve aucune trace indiquant qu'un courriel relatif à ce bogue ait été envoyé à qui que ce soit. Après avoir établi une liste, il utilise ensuite les règles standard pour déterminer qui recevra le courriel, et l'émet.

Lorsque le script s'exécute, il indique le bogue pour lequel il envoie le présent courriel; lorsqu'il a fini, il fournit un total chiffré des courriels envoyés et du nombre de personnes qui en ont été exclues. (Les noms d'utilisateurs personnels ne sont pas enregistrés ni affichés.) Si le script ne produit aucun résultat en sortie, cela signifie qu'aucun courriel non-envoyé n'a été détecté.

Mode d'emploi : faites remonter le script sendunsentbugmail.pl dans le répertoire principal, assurez vous qu'il possède les droits en exécution et lancez le depuis la ligne de commande (ou depuis un utilitaire de tâches planifiées) sans aucun paramètre.

Annexe D. Installation manuelle des modules Perl

Table des matières

1. Instructions

Si vous devez installer des modules Perl à la main, voici comment procéder. Téléchargez le module en utilisant le lien donné dans la section suivante, puis appliquez la formule magique suivante, en tant que root :

```
bash# tar -xzvf <module>.tar.gz
bash# cd <module>
bash# perl Makefile.PL
bash# make
bash# make test
bash# make install
```

Note

Pour compiler le code source sous Windows, il vous faudra obtenir un utilitaire de type « make ». Vous pourrez par exemple utiliser l'utilitaire **nmake** accompagnant le Visual C++ de Microsoft. Il existe également dans CPAN un utilitaire entièrement écrit en Perl nommé **nmake**. La plupart des liens donnés ci-dessous, cependant, sont prévus pour les versions précompilées des modules, qui peuvent être installés sous Windows simplement en entrant la commande suivante, après que vous avez téléchargé le fichier PPD (qui est probablement contenu dans un fichier ZIP) :

```
> ppm install <nom_du_fichier.ppd>
```

2. Sites de téléchargement

Note

Le lancement de Bugzilla sous Windows nécessite l'utilisation de ActiveState Perl 5.8.1 ou supérieur. Certains modules existent déjà dans la distribution principale d'ActiveState Perl, aucun lien PPM n'est par conséquent donné (cela sera mentionné le cas échéant).

AppConfig :

Page de téléchargement CPAN : http://search.cpan.org/src/ABW/AppConfig-1.56/lib/AppConfig.pm Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/AppConfig.ppd Documentation : http://search.cpan.org/~abw/AppConfig-1.56/lib/AppConfig.pm

CGI :

Page de téléchargement CPAN : http://search.cpan.org/dist/CGI.pm/ Lien de téléchargement PPM : intégré à la distribution principale. Documentation : http://www.perldoc.com/perl5.8.0/lib/CGI.html

Data-Dumper :

Page de téléchargement CPAN : http://search.cpan.org/src/ILYAM/Data-Dumper-2.121/Dumper.pm Lien de téléchargement PPM : intégré à la distribution principale. Documentation : http://search.cpan.org/~ilyam/Data-Dumper-2.121/Dumper.pm

Date::Format (contenu dans TimeDate) :

Page de téléchargement CPAN : http://search.cpan.org/dist/TimeDate/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/TimeDate.ppd Documentation : http://search.cpan.org/dist/TimeDate/lib/Date/Format.pm

DBI :

Page de téléchargement CPAN : http://search.cpan.org/dist/DBI/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/DBI.ppd Documentation : http://dbi.perl.org/docs/

DBD::mysql :

Page de téléchargement CPAN : http://search.cpan.org/dist/DBD-mysql/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/DBD-mysql.ppd Documentation : http://search.cpan.org/dist/DBD-mysql/lib/DBD/mysql.pm

File::Spec :

Page de téléchargement CPAN : http://search.cpan.org/dist/File-Spec/ Lien de téléchargement PPM : intégré à la distribution principale. Documentation : http://www.perldoc.com/perl5.8.0/lib/File/Spec.html

File::Temp :

Page de téléchargement CPAN : http://search.cpan.org/dist/File-Temp/ Lien de téléchargement PPM : intégré à la distribution principale. Documentation : http://www.perldoc.com/perl5.8.0/lib/File/Temp.html

Template-Toolkit :

Page de téléchargement CPAN : http://search.cpan.org/dist/Template-Toolkit/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/Template-Toolkit.ppd Documentation : http://www.template-toolkit.org/docs.html

Text::Wrap :

Page de téléchargement CPAN : http://search.cpan.org/dist/Text-Tabs+Wrap/ Lien de téléchargement PPM : intégré à la distribution principale. Documentation : http://www.perldoc.com/perl5.8.0/lib/Text/Wrap.html

GD :

Page de téléchargement CPAN : http://search.cpan.org/dist/GD/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/GD.ppd Documentation : http://stein.cshl.org/WWW/software/GD/

3. Modules optionnels

Chart::Base :

Page de téléchargement CPAN : http://search.cpan.org/dist/Chart/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/Chart.ppd Documentation : http://search.cpan.org/src/CHART-GRP/Chart-2.3/doc/Documentation.pdf

GD::Graph :

Page de téléchargement CPAN : http://search.cpan.org/dist/GDGraph/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/GDGraph.ppd Documentation : http://search.cpan.org/dist/GDGraph/Graph.pm

GD::Text::Align (contenu dans GD::Text::Util) :

Page de téléchargement CPAN : http://search.cpan.org/dist/GDTextUtil/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/GDTextUtil.ppd Documentation : http://search.cpan.org/dist/GDTextUtil/Text/Align.pm

MIME::Parser (contenu dans MIME-tools) :

Page de téléchargement CPAN : http://search.cpan.org/dist/MIME-tools/ Lien de téléchargement PPM : http://ppm.activestate.com/PPMPackages/zips/8xx-builds-only/Windows/MIME-tools-5.411a.zip Documentation : http://search.cpan.org/dist/MIME-tools/lib/MIME/Parser.pm

XML::Parser :

Page de téléchargement CPAN : http://search.cpan.org/dist/XML-Parser/ Lien de téléchargement PPM : intégré à la distribution principale. Documentation : http://www.perldoc.com/perl5.6.1/lib/XML/Parser.html

PatchReader :

Page de téléchargement CPAN : http://search.cpan.org/author/JKEISER/PatchReader/ Lien de téléchargement PPM : http://landfill.bugzilla.org/ppm/PatchReader.ppd Documentation : http://www.johnkeiser.com/mozilla/Patch_Viewer.html

Annexe E. GNU Free Documentation License

Table des matières

How to use this License for your documents [p 123]

Version 1.1, March 2000

0. Preamble

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. Applicability and Definition

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. Verbatim Copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. Copying in Quantity

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. Combining Documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. Collections of Documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. Aggregation with Independent Works

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. Future Revisions of this License

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

. How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Glossaire

0-9, caractères spéciaux

.htaccess

> Le serveur Web Apache, ainsi que d'autres serveurs web compatibles avec le NCSA, adoptent la convention visant à utiliser des fichiers appelés `.htaccess` à l'intérieur des répertoires pour restreindre l'accès à certains fichiers. Dans Bugzilla, ils sont utilisés pour cacher des fichiers secrets qui pourraient, s'ils étaient découverts, mettre en péril votre installation ; par exemple, le fichier `localconfig` contient le mot de passe pour votre base de données. Curieux.

A

Apache

Dans ce contexte, Apache représente le serveur web le plus couramment utilisé pour mettre en ligne les pages de Bugzilla. Contrairement à une vieille croyance populaire, le serveur web apache n'a rien à voir avec l'ancienne et noble tribu d'indiens d'Amérique mais tient son nom du fait qu'il s'agissait d'une version ayant eu des correctifs (« a patchy version ») du serveur web original du NCSA. NdT : en anglais, Apache se prononce comme « A patchy ».

Instructions utiles pour configurer Bugzilla

`AddHandler`

Indique à Apache qu'il a le droit d'exécuter les scripts CGI.

`AllowOverride`, `Options`

Ces instructions sont utilisées pour indiquer à Apache un certain nombre de choses relatives au répertoire auquel elles s'appliquent. Pour Bugzilla par exemple, nous en avons besoin pour autoriser l'exécution de scripts et l'utilisation de fichiers `.htaccess` pour les droits.

`DirectoryIndex`

Utilisé pour indiquer à Apache quels fichiers sont utilisés en tant qu'index. Si vous ne pouvez pas ajouter `index.cgi` à la liste des fichiers valides, vous devez fixer la valeur de `$index_html` à 1 dans `localconfig` de telle manière que **./checksetup.pl** crée un `index.html` qui redirige vers `index.cgi`.

`ScriptInterpreterSource`

Utilisé lorsque l'on fait fonctionner Apache sous windows afin de ne pas avoir à changer la ligne et tout le toutim dans chaque script de Bugzilla.

Pour de plus amples informations afin de configurer Apache pour Bugzilla, jetez un œil à Section 2.4.1, « httpd™ d'Apache » [p 19] .

Attaque de type DOS

Un DOS, ou attaque de déni de service, a lieu quand un utilisateur tente de bloquer l'accès à un serveur Web en accédant de manière répétée à une page ou en envoyant des requêtes erronées au serveur. On peut l'éviter efficacement en utilisant `mod_throttle` comme décrit dans Section 3.2, « Utilisation de `mod_throttle` pour éviter un déni de service » [p 55] . Une attaque de déni de service distribué a lieu quand ces requêtes proviennent de multiples sources au même moment. Malheureusement, elles sont beaucoup plus difficiles à combattre.

B

Bogue

Dans Bugzilla, un « bogue » correspond à une anomalie que l'on a ajoutée dans la base de données, associé à un numéro, des affectations, des commentaires, etc. Certains parlent aussi de « tickets » ou « issues » ; dans le contexte Bugzilla, ces termes sont synonymes.

Bugzilla

Bugzilla est le leader mondial du logiciel libre de suivi de bogues.

C

Cibles Jalons

Les cibles jalons ont les objectifs du produit. Ils sont paramétrable produit par produit. La plupart des boîtes de développement logiciel utilisent le concept de « jalons » lorsque ceux qui financent un projet attendent certaines fonctionnalités pour certaines dates. Bugzilla facilite l'utilisation de ce concept de jalons en vous donnant la possibilité de déclarer à quel jalon un bogue sera réparé ou une amélioration sera apportée.

Common Gateway Interface

CGI est l'acronyme de Common Gateway Interface. Il s'agit d'un standard pour interfacer une application externe grâce à un serveur web. Bugzilla est un exemple d'application CGI.

Composant

Un composant est une sous-partie d'un Produit. Il faut que ce soit une catégorie limitée, à la mesure de votre organisation. Tout Produit doit obligatoirement contenir au moins un Composant (et d'ailleurs, créer un Produit sans Composant générera une erreur dans Bugzilla).

Comprehensive Perl Archive Network

CPAN signifie « Comprehensive Perl Archive Network ». Le CPAN assure le suivi d'un grand nombre de modules *Perl* extrêmement utiles, qui sont de gros morceaux de code encapsulés pour réaliser une tâche particulière.

`contrib`

Le répertoire `contrib` est un endroit où l'on place les scripts qui participent au fonctionnement de Bugzilla mais qui ne font pas partie de la distribution officielle. Ces scripts sont écrits par des personnes tierces et sont susceptibles d'être codés dans d'autres langages que Perl. Pour ceux qui sont écrits en Perl, il peut exister des modules supplémentaires ou des requis autres que ceux de la distribution officielle.

> **Note**
>
> Les scripts du répertoire `contrib` ne sont pas officiellement pris en charge par l'équipe de Bugzilla, il se pourrait donc qu'ils ne fonctionnent plus entre différentes versions.

D

Démon

Un démon est un programme informatique qui s'exécute en tâche de fond. En général, la plupart des démons s'exécutent au démarrage du système via des scripts d'initialisation System V ou à l'aide de scripts RC sur les systèmes BSD. *mysqld*, le serveur MySQL et *Apache*, un serveur web, sont généralement des démons.

E

Expression rationnelle

Une expression rationnelle est une expression utilisée pour la reconnaissance de formulations. Documentation

G

Gestionnaire de Courrier Électronique

Un gestionnaire de courrier électronique sert à gérer le flot de courriers électroniques d'un système. Plusieurs systèmes UNIX utilisent sendmail qui est ce que Bugzilla s'attend à trouver par défaut dans `/usr/sbin/sendmail`. Plusieurs autres MTA fonctionneront, mais ils nécessitent tous que le paramètre `sendmailnow` soit fixé à `on`.

Groupes

Le mot « Groups » possède un sens très spécial dans Bugzilla. Le principal système de sécurité de Bugzilla intervient en plaçant les utilisateurs dans des groupes et en donnant à ces groupes certaines permissions pour voir les bogues de *Produits* particuliers dans la base de données de *Bugzilla*.

J

JavaScript

Le JavaScript, c'est cool, nous devrions en parler.

M

MySQL

MySQL est actuellement le *SGBD* [p 128] utilisé par Bugzilla. MySQL peut être téléchargé à partir de http://www.mysql.com. Il est nécessaire de prendre connaissance de l'ensemble de la documentation, mais voici les principaux points :

Sauvegardes

Méthodes pour faire une sauvegarde de votre base de données Bugzilla.

Fichier d'options

Informations pour configurer MySQL en utilisant `my.cnf`.

Règles de sécurité et droits d'accès

Informations beaucoup plus détaillées sur les suggestions de Section 2, « MySQL » [p 53] .

N

Numéro de Bogue

À chaque bogue de Bugzilla est affecté un numéro qui identifie ce dernier de manière unique. On peut récupérer le bogue associé à un numéro de bogue via une requête, ou simplement à partir de la toute première page en tapant le numéro dans le champ « Trouver ».

P

Perl

Écrit à l'origine par Larry Wall, Perl est un langage de programmation remarquable. Il possède les avantages d'un langage de scripts interprété (comme les scripts shell), associés à la rapidité et à la puissance de langages compilés comme le C. *Bugzilla* est programmé en Perl.

Perl Package Manager

http://aspn.activestate.com/ASPN/Downloads/ActivePerl/PPM/

Produit

Un Produit rassemble toute une catégorie de types de bogues, et représente normalement un seul logiciel ou une seule entité. En général, il existe plusieurs Composants dans un Produit. Un Produit peut définir un groupe (à des fins de sécurité) pour tous les bogues de ses Composants.

Q

QA

« QA », « Q/A » et « Q.A. » sont des abréviations de « Quality Assurance » (Assurance Qualité). Dans la plupart des grandes sociétés de développement logiciel, il y a une équipe chargée d'assurer que le produit respecte un minimum de standards avant sa livraison. Souvent cette équipe voudra aussi suivre la progression des bogues tout au long de leur cycle de vie, d'où la présence d'un champ « Responsable Qualité » dans un bogue.

S

Service

Dans l'environnement Windows NT, une application lancée en arrière-plan lors du démarrage s'appelle un service. C'est généralement le panneau de commandes qui les gère lorsque vous êtes connecté sur un compte possédant le niveau d'accréditation « Administrateur ». Pour plus d'informations, consultez votre manuel Windows ou le MSKB.

SGML

SGML signifie « Standard Generalized Markup Language ». Créé dans les années 80 pour fournir un moyen évolutif de conserver les documents, basé sur le contenu et non sur la présentation, SGML a résisté à l'épreuve du temps et s'est affirmé comme un langage puissant et robuste. *XML* est le « petit frère » de SGML ; tout document XML valide est, par définition un document SGML valide. Le document que vous êtes en train de lire est écrit et maintenu en SGML, et est également du XML valide si vous modifiez la DTD.

Système de Gestion de Base de Données Relationnelle

Un système de gestion de base de données relationnelle est un système de base de données qui stocke les informations dans des tables liées entre elles.

T

Tool Command Language

TCL est un langage de scripts libre et disponible sur les plate-formes Windows, Macintosh et Unix. Bugzilla 1.0 a été écrit en TCL mais n'a jamais vu le jour. La première version de Bugzilla fut la 2.0 lorsqu'il a été porté en Perl.

Z

Zarro Boogs Found

C'est juste une façon rigolote de dire qu'aucun bogue correspondant à votre requête n'a été trouvé. Lorsqu'on lui a demandé d'expliquer ce message, Terry a déclaré :

> On m'a demandé d'expliquer ce truc... Il y a bien longtemps, lorsque Netscape a fait paraître la version 4.0 de son navigateur, nous avons fait une fête pour la sortie. Naturellement, ça avait été la grosse panique pour essayer de réparer tous les bogues identifiés avant la sortie. Naturellement, nous n'y sommes pas parvenus. (Ce genre de chose n'est pas arrivée qu'à Netscape ou à la version 4.0 ; la même chose s'est produite dans chaque projet logiciel auquel j'ai participé). Enfin bref, à la fête de sortie, des T-shirts où il était marqué quelque chose comme « Netscape 4.0: Zarro Boogs » furent distribués. Tout comme le logiciel, le T-shirt n'avait aucun bogue d'identifié. Hé hé.
>
> Donc lorsque vous effectuez une requête pour obtenir une liste de bogues et que vous n'obtenez aucun résultat en retour, vous penserez à cela avec un petit sourire. Bien sûr qu'il y a des bogues correspondant à votre requête, c'est simplement qu'ils ne sont pas encore listés dans le système de bogues...

--Terry Weissman