

Mettre en place une sécurité sur les branches dans le cadre du workflow GitFlow

Base de connaissances techniques

Exported on 03/31/2020

Table of Contents

1 Besoin initial.....	3
2 Configuration des branches et des groupes	4
3 Mise en place des règles de sécurité	5
4 Cas d'usage.....	7

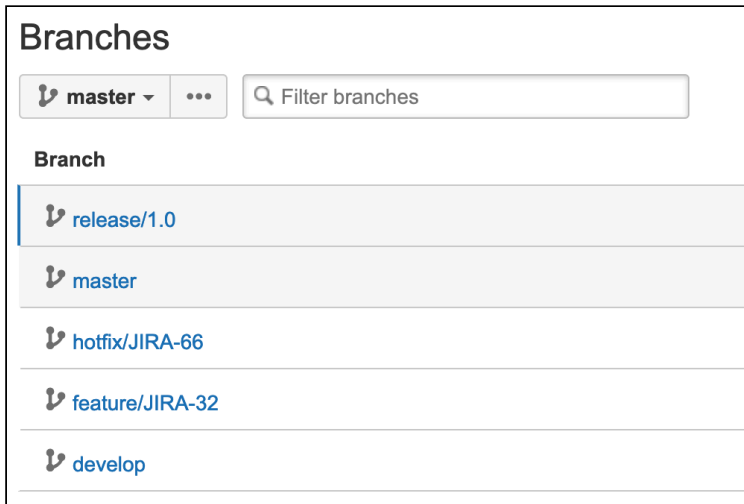
1 Besoin initial

Le besoin exprimé utilise le workflow Git-Flow :

- La branche **master** doit être en lecture seule.
- Un groupe d'utilisateur A réalise un "commit" sur une branche de **develop, feature/*, hotfix/*** ou de **release/*** et peut déclencher le build et le déploiement sur un environnement HORS-PROD.
- Ce même groupe d'utilisateur A peut créer toutes les Pull Request.
- Ce même groupe d'utilisateur A peut approuver toutes Pull Request touchant **develop, feature/*, hotfix/*** ou de **release/***.
- Toute modification de la branche **master** doit obligatoirement passer par une Pull Request (**processus encore à déterminer**).
- Un groupe d'utilisateur B (ou un ensemble de groupes) seul est autorisé à valider les Pull Request de **release/*** ou de **hotfix/*** vers **master** ce qui déclenche théoriquement un build (et le droit de relancer le build manuellement) en **production**

2 Configuration des branches et des groupes

La repository est configurée de la façon suivante :



En ce qui concerne cet exemple, nous partirons sur deux groupes d'utilisateurs, pour représenter les groupes A et B ci-dessus :

- Le groupe A est celui des développeurs (jira-users)
- Le groupe B est celui des administrateurs (bitbucket-admin)

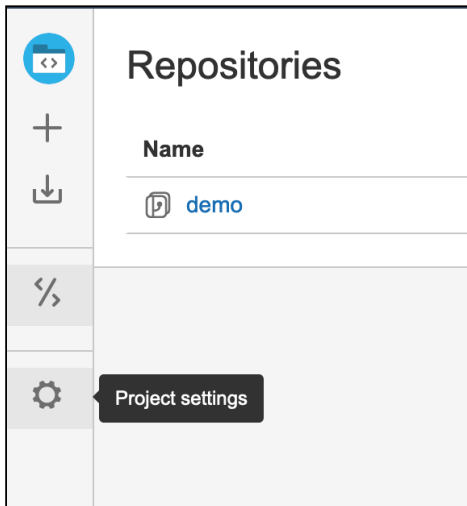
Bien évidemment il s'agit d'un exemple et ceci peut être adapté en fonction de la structure des groupes en place.

3 Mise en place des règles de sécurité

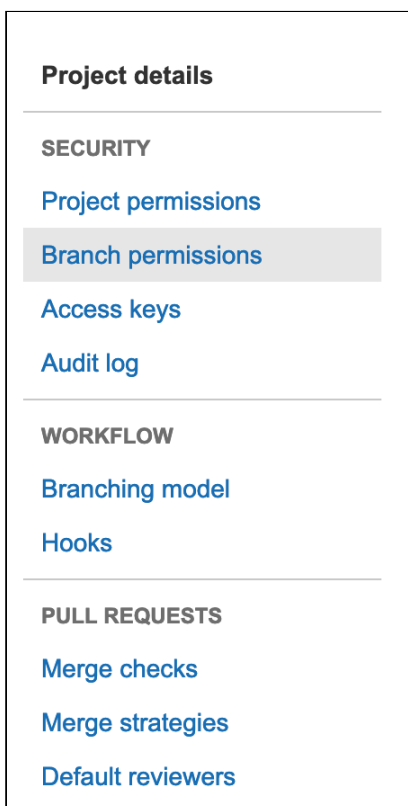
La configuration est réalisée au niveau du projet dans son ensemble. De cette façon, il n'est pas nécessaire de répliquer cette configuration pour chaque repository.

Ceci dit, si on le souhaite, on peut surcharger cette configuration dans chaque repository du projet.

Aller dans le projet et en tant qu'administrateur de celui-ci, cliquer sur Project Settings

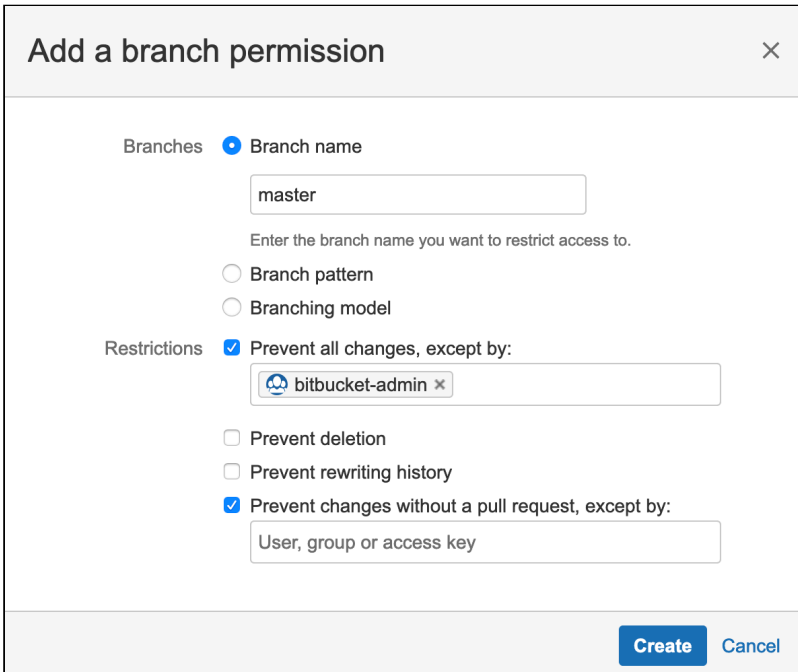


Cliquer ensuite sur Branch Permissions :



Cliquer sur **Add Permissions** en haut à droite de la fenêtre.

Configurer les droits sur la repository master :



Branches : on nomme la branche sur laquelle appliquer les restrictions (ici, master)

Restrictions : dans cet exemple, on interdit toute modification sur la branche, sauf pour les membres du groupe bitbucket-admin (on pourrait aussi nommer des utilisateurs individuels).

De ce fait, les suppressions et les réécritures d'historiques ne sont pas non plus possibles pour les membres des autres groupes. Donc inutile ici d'ajouter bitbucket-admin dans les utilisateurs ayant le droit de réaliser ces opérations. Néanmoins, on pourrait tout à fait mettre la configuration suivante :

Prevent all changes, except by: bitbucket-admin, bitbucket-super-admin

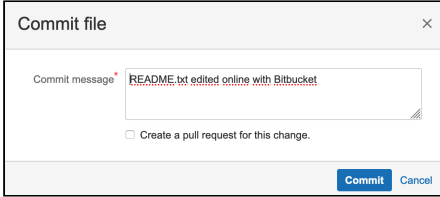
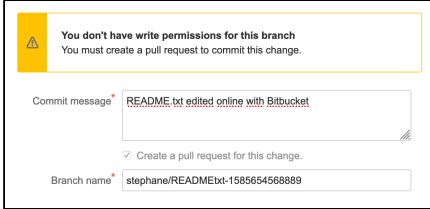
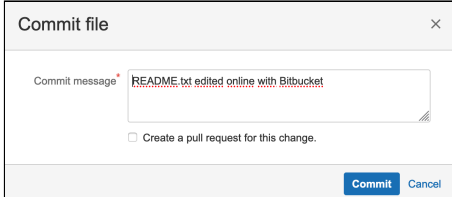
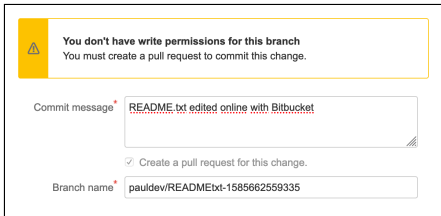
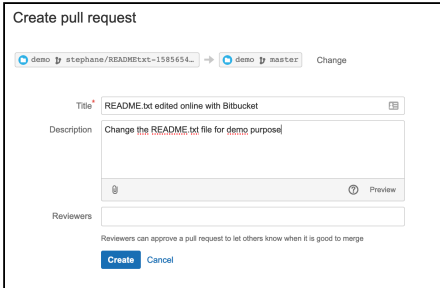
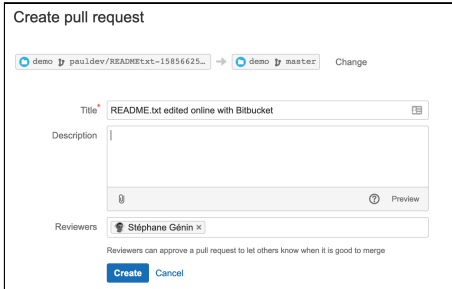
Prevent deletion : bitbucket-super-admin → dans ce cas, seuls les membres de bitbucket-super-admin peuvent supprimer, alors que les membres de bitbucket-admin pourront fusionner des pull-request par exemple mais pas faire de suppressions.

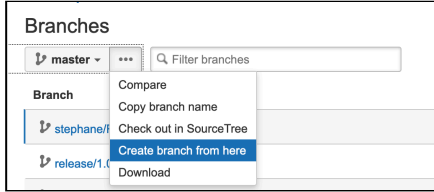
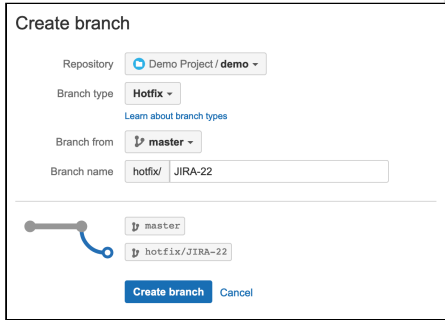

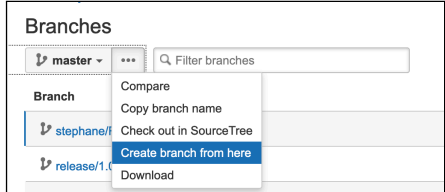
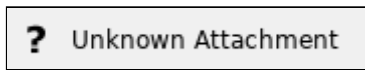
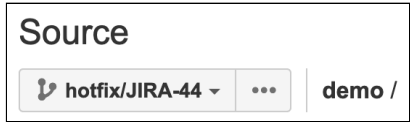
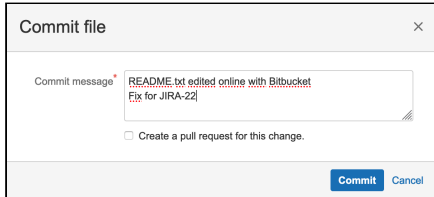
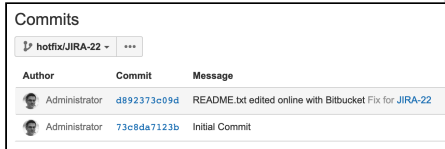
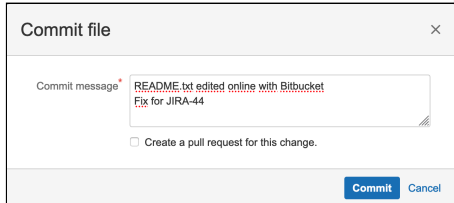

Prevent rewriting history : cocher sans mettre de groupe → dans ce cas personne ne pourra réécrire l'historique

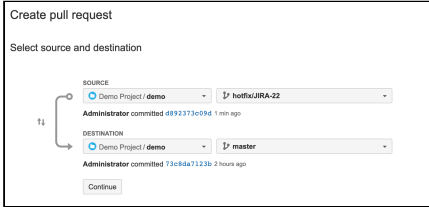
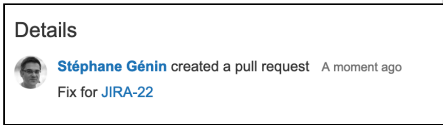
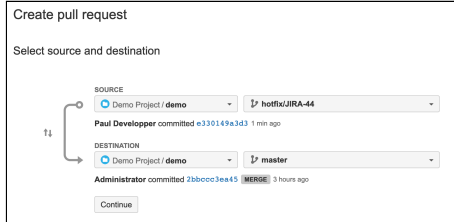
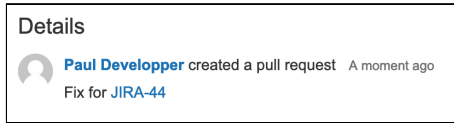

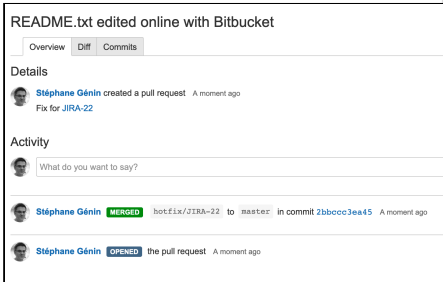
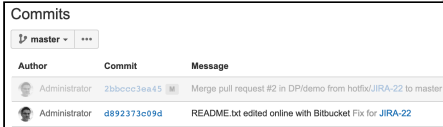
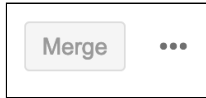
Ici, on interdit également pour tout le monde de pouvoir réaliser de modifications sans pull-requests (comme des commits directs). Dans notre exemple, cela signifie que même si les membres du groupe bitbucket-admin peuvent fusionner des pull requests sur le master, ils ne peuvent pas faire des commits directement dans la branche. La encore, on pourrait faire en sorte que les membres de bitbucket-super-admin puissent faire des commits directement.

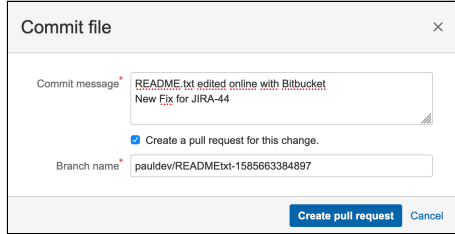
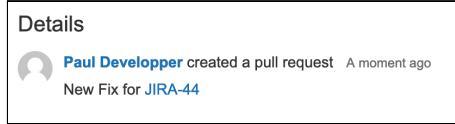

4 Cas d'usage

- ✓ Dans ces exemples, les manipulations sont faites directement en faisant les modifications de code dans Bitbucket. Dans les cas réels, les opérations sont faites via l'IDE en général. Ceci est juste fait pour démontrer la mise en oeuvre des règles de sécurité.

Action	Administrateur de la repo	Développeur sans privilège
Faire un commit directement dans le master	 <p>Au moment du commit, Bitbucket donne la possibilité de créer une pull-request.</p> <p>Dans notre exemple, nous essayons un commit direct.</p>  <p>Le commit est refusé, et Bitbucket propose de créer une pull request.</p>	 <p>Au moment du commit, Bitbucket donne la possibilité de créer une pull-request.</p> <p>Dans notre exemple, nous essayons un commit direct.</p>  <p>Le commit est refusé, et Bitbucket propose de créer une pull request.</p>
Créer une pull-request dans le master pour un commit réalisé dans le master	<p>On repart de l'exemple ci-dessus, et on valide la création de la Pull Request.</p>  <p>La pull request est créée.</p>	<p>On repart de l'exemple ci-dessus, et on valide la création de la Pull Request.</p>  <p>Dans cet exemple, nous avons configuré un relecteur par défaut.</p> <p>La pull request est créée.</p>

Action	Administrateur de la repo	Développeur sans privilège
Créer une branche Hot Fix à partir du master	<p>On demande la création de la branche Hot Fix</p>  <p>On demande la création d'une branche de type HotFix</p>  <p>La branche est créée.</p> 	<p>On demande la création de la branche Hot Fix</p>  <p>On demande la création d'une branche de type HotFix</p>  <p>La branche est créée.</p> 
Faire un commit direct dans la branche Hot Fix	<p>On réalise une modification dans la branche HotFix, et on demande un commit directement :</p>  <p>Le commit est réalisé directement.</p> 	<p>On réalise une modification dans la branche HotFix, et on demande un commit directement :</p>  <p>Le commit est réalisé directement.</p> 

Action	Administrateur de la repo	Développeur sans privilège
Créer une pull-request de la branche Hot Fix vers le master	<p>On demande la création d'une Pull Request de la branche JIRA-22 vers le master :</p>  <p>La Pull Request est créée :</p> 	<p>On demande la création d'une Pull Request de la branche JIRA-44 vers le master :</p>  <p>La Pull Request est créée :</p> 
Accepter une pull-request d'une branche Hot Fix (par exemple) dans le master	<p>Afficher la Pull Request et cliquer sur Merge en haut à droite de la fenêtre.</p>  <p>Cliquer sur Merge.</p> <p>La Pull Request est fusionnée :</p>  	<p>Afficher la Pull Request.</p> <p>Le bouton Merge n'est pas accessible :</p> 

Action	Administrateur de la repo	Développeur sans privilège
Réaliser une modification dans la branche HotFix et créer une Pull Request	Le comportement est identique à celui de la branche master car il n'y a pas de restriction sur la branche HotFix.	<p>Demander la création d'une Pull Request au moment du commit.</p>  <p>La Pull Request est créée</p> 
Accepter la pull-request dans la branche Hot Fix	Le comportement est identique à celui de la branche master car il n'y a pas de restriction sur la branche HotFix.	<p>Afficher la Pull Request et cliquer sur Merge en haut à droite de la fenêtre.</p>  <p>Cliquer sur Merge.</p> <p>La Pull Request est fusionnée :</p> 