Wikiprint Book

Title: 粒度が細かいパーミッション

Subject: SilverFrost - TracFineGrainedPermissions

Version: 3

Date: 06/04/26 05:17:28

SilverFrost 目次

粒度が細かいパーミッション

Trac 0.11 より前は、リポジトリブラウザ サブシステムだけで「粒度が細かいパーミッション (fine grained permissions)」を定義することができました。

0.11 以降、カスタマイズしたパーミッションポリシーのプラグインを各所に使用するための共通のメカニズムが導入されたので、すべての種類の Trac リソースのあらゆるアクションについて、そのリソースの特定バージョンのレベルまで含めて許可/拒否を設定できるようになりました。

パーミッションポリシー

AuthzPolicy

ポリシーの例として、 Authz 形式のシステムを基にしたポリシーが追加されました。詳しくは、 http://trac.edgewall.org/browser/trunk/sample-plugins/permissions/authz_policy.py を参照してください。 (より多くの例が http://trac.edgewall.org/browser/trunk/sample-plugins/permissions にあります。)

- ConfigObj をインストール (必須)
- authz_policy.py を plugins ディレクトリにコピーする
- authzpolicy.conf ファイルをどこか (できれば、 Web サーバ起動ユーザ以外が読み取りできないセキュアな領域) に置く。
  trac.ini ファイルをアップデートする:

```
[trac]
...
permission_policies = AuthzPolicy, DefaultPermissionPolicy, LegacyAttachmentPolicy

[authz_policy]
authz_file = /some/trac/env/conf/authzpolicy.conf

[components]
...
authz_policy = enabled
```

パーミッションポリシーを指定する順序はとても重要です。 ポリシーは設定された順序で評価されます。

個々のポリシーはパーミッションチェックに対して True, False, None を返します。 戻り値が None の場合のみ 次のパーミッションポリシーに問い合わせを行います。 どのポリシーも明示的にパーミッションを許可しない場合、最終的な結果は False になります (つまり、権限なしとみなされます)。

例えば、 authz_file が次の内容を含み:

```
[wiki:WikiStart@*]
* = WIKI_VIEW

[wiki:PrivatePage@*]
john = WIKI_VIEW
* =
```

デフォルトパーミッションが次のような内容の場合:

```
john           WIKI_VIEW
jack           WIKI_VIEW
# anonymous ■ WIKI_VIEW ■■■■■■■■■
```

パーミッションは以下の通りとなります:

- WikiStart の全てのバージョンは、 (匿名ユーザも含む) 全員が閲覧できます。
- PrivatePage は john が表示可能です。
- 他のページは john と jack が表示可能です。

mod_authz_svn ライクなパーミッションポリシー

この文書が書かれている時点では、 Trac 0.10
以前にリポジトリへの厳密なアクセス制御に使用されていた、古い「粒度が細かいパーミッション」システムは、まだパーミッションポリシーのコンポーネントにこ

「粒度が細かいパーミッション」の制御に定義ファイルを必要とします。この定義ファイルは Subversion の mod_authz_svn
で使用しているものを使います。 このファイルの形式と Subversion での用法に関する情報は Subversion Book (ディレクトリごとのアクセス制御)
を参照してください。

例:

```
[/]
* = r

[/branches/calc/bug-142]
harry = rw
sally = r

[/branches/calc/bug-142/secret]
harry =
```

- / = 全員 read アクセスが可能です。これはデフォルトの動作となります
- /branches/calc/bug-142 = harry は read/write アクセス権を持ち、 sally は read アクセス権のみを持ちます
- /branches/calc/bug-142/secret = harry はアクセス権を持たず、 sally は read アクセス権を持ちます
  (パーミッションはサブフォルダに継承されます)

Trac の設定

「粒度が細かいパーミッション」を有効にするには、 trac.ini ファイルの [trac] セクションに authz_file オプションを
設定しなければなりません 。オプションが空値に設定されていたり、そもそも指定されていない場合、パーミッションは適用されません。

```
[trac]
authz_file = /path/to/svnaccessfile
```

auth_file 内でシンタックス [modulename:/some/path] を使用する場合、以下の設定を追加してください:

```
authz_module_name = modulename
```

modulename には、 [trac] セクション中の repository_dir に設定したリポジトリと同じものを設定します。 (訳注: Subversion で
SVNParentPath を使用して複数のリポジトリをホストしている場合のリポジトリ指定方法です。 modulename は個々のリポジトリを指します。)

Note: Authz ファイルで使用するユーザ名と、 Trac で使用するユーザ名は 同じでなければなりません。

Subversion の設定

通常は同じアクセスファイルを対応する Subversion リポジトリに適用します。 Apache のディレクティブには以下のように設定してください:

```
<Location /repos>
 DAV svn
 SVNParentPath /usr/local/svn

 # our access control policy
 AuthzSVNAccessFile /path/to/svnaccessfile
</Location>
```

複数のプロジェクト Environment において、プロジェクト全体にどのようにアクセス制限を行うかについての情報は
http://trac.edgewall.org/wiki/TracMultipleProjectsSVNAccess を参照してください。

See also: TracPermissions