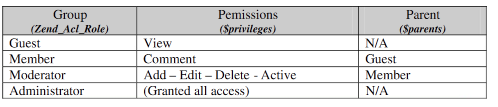
***Zend Acl***

* ***ACL =*** ***Access control list*** *(danh sách điều khiển truy cập)*
* ***Access control*** : quy định quyền hạn của các ***User*** trên các tài nguyên của ứng dụng.Trên web, ***Access* *control*** quyết định một ***User*** nào đó được cho phép xem một trang nào đó hoặc thực hiện một hành động nào đó hay ko (Vd như thêm một bình luận chẳng hạn).
* ***Access control list*** *(Acl)* là một danh sách điều khiển truy cập, danh sách này gắn liền với một tài nguyên nào đó. ***Acl*** quy định cụ thể ai được phép truy cập vào tài nguyên và những chức năng nào được cho phép trên tài nguyên này. Bất cứ khi nào người dùng muốn làm điều gì đó, ứng dụng sẽ kiểm tra danh sách để biết rằng người dùng này có được cho phép thực hiện hành động với mục dữ liệu đó hay ko. Ví dụ, một người sử dụng có thể được phép để xem một bài báo, nhưng ko được phép chỉnh sửa nó.
* **Acl** là một giải pháp rất linh hoạt để kiểm soát truy cập.
* ***Các thuật ngữ điều khiển truy cập (Access control )***  
  ■ role: nhóm các user  
  ■ resource: các tài nguyên cần được bảo vệ (chẳng hạn như 1 controller action, hay 1 dữ liệu nào đó…)  
  ■ privilege: các dạng yêu cầu truy cập (chẳng hạn như đọc, chỉnh sửa dữ liệu nào đó)

***Zend Acl:***

*Zend\_Acl giúp chúng ta phân quyền cho người dùng một cách chi tiếtquyền chi tiết trên từng action, controller, module.*



***Các bước cơ bản để sử dụng Zend\_Acl:***

1. Khởi tạo Zend\_Acl

$acl = **new** Zend\_Acl();

1. Tạo các Role, khai báo các nhóm thành viên có trong hệ thống

$acl->addRole(**new** Zend\_Acl\_Role('Guest'))

->addRole(**new** Zend\_Acl\_Role('Member'))

->addRole(**new** Zend\_Acl\_Role('Moderator'))

->addRole(**new** Zend\_Acl\_Role('Administrator'));

1. Khai báo sự phụ thuộc lẫn nhau của các nhóm

$acl->addRole(**new** Zend\_Acl\_Role('Guest'))

->addRole(**new** Zend\_Acl\_Role('Member'), 'Guest')

->addRole(**new** Zend\_Acl\_Role('Moderator'), 'Member')

->addRole(**new** Zend\_Acl\_Role('Sdministrator'));

* Khai báo quyền cho các nhóm:

$guestPrivileges = **array**('view');

$memberPrivileges = **array**('comment');

$moderatorPrivileges = **array**('add', 'edit', 'delete', 'active');

1. Cấp quyền truy cập cho các nhóm

$acl->allow('Guest', **null**, $guestPrivileges);

$acl->allow('Member', **null**, $memberPrivileges);

$acl->allow('Moderator', **null**, $moderatorPrivileges);

$acl->allow('Administrator');

//trong phương thức allow() nếu tham số $privileges ko xuất hiện thì nghĩa là nhóm đó có toàn quyền

1. Cấm truy cập

//ngược lại với thương thức allow90 là phương thức deny(). Phương thức này sẽ cấp quyền cập vào

$acl->allow('Guest', **null**, $guestPrivileges);

$acl->allow('Member', **null**, $memberPrivileges);

$acl->allow('Moderator', **null**, $moderatorPrivileges);

$acl->deny('Moderator', **null**, 'delete');

$acl->allow('Administrator');

1. Kiểm tra quyền truy cập

//Để kiểm tra quyền truy cập vào một phần nào đó của người dùng ta sẽ sử dụng phương thức kiểm tra isAllowed

**if**($acl->isAllowed('Moderator', **null**, 'delete'))

{

**echo** 'Bạn có quyền truy cập action này';

}

**else**

{

**echo** 'Bạn ko có quyền truy cập action này';

}

1. Resource

// Trong Acl này xuất hiện một tham ssos là $resource và để sử dụng nó chúng ta dùng phương thức add()

$acl->add(**new** Zend\_Acl\_Resource('admin'));

$acl->add(**new** Zend\_Acl\_Resource('admin:author'), 'admin');

$acl->add(**new** Zend\_Acl\_Resource('admin:member'), 'admin');

1. Kiểm tra quyền truy cập theo Resource

$module = 'admin';

$controller = 'author';

$action = 'view';

$resource = $module. ':'. $controller;

**if**($acl->isAllowed('Moderator', $resource, $action))

{

**echo** 'Bạn có quyền truy cập action này';

}

**else**

{

**echo** 'Bạn ko có quyền truy cập action này';

}

ACL là một giải pháp rất linh hoạt để kiểm soát truy cập.