

Programmation web et multimédia 4 – 2024

Validation côté serveur

Pourquoi ?

- NE JAMAIS faire confiance à la validation côté client car elle peut être très facilement contournée par un programmeur malicieux notamment avec un outil comme **Curl**.
- TOUJOURS valider côté serveur les valeurs reçues dans les superglobales telles que `$_POST`, `$_GET`, `$_COOKIE`, etc.

Fonctions PHP utiles avant une validation :

- **isset()**: retourne **true** si une variable ou une clé existe et si la valeur n'est pas **null**.
- **trim()**: enlever les caractères invisibles avant et après dans une chaîne tels que les espaces.

Expression régulière

- Il est important d'entourer un motif avec **^** et **\$** afin d'éviter une injection avant ou après le motif. Exemple `#^motif-ici$#`

Faire une redirection http après une validation

- Exemple de redirection vers une autre URL :

```
<?php
header('Location: index.php?controleur=inscription&action=creer'); // redirection
exit(); // terminer immédiatement le script
```

- Les redirections sont des en-têtes HTTP. Or, selon le protocole HTTP, les en-têtes HTTP doivent être envoyés avant tout autre type de contenu, ce qui signifie qu'aucun caractère ne doit être envoyé avant l'appel de la fonction `header`, pas même un espace.

Cas particulier :

- Un champ case à cocher « non cochée » n'est pas envoyé au serveur et n'apparaît pas dans `$_POST`.
- Un champ case à cocher « cochée » est envoyé au serveur et apparaît dans `$_POST` avec la valeur « On » (par défaut).

Étapes de validation d'un formulaire en MVC

