Making your PrestaShop installation more secure

Making your PrestaShop installation more secure

There are several ways anyone, whatever the technical level, can enhance the safety of his/her PrestaShop install.

Here are few easy-to-apply tips.

Always use the latest version of PrestaShop

That's a given, really, but it bears repeating. New versions of PrestaShop contains new features, improvements and bugfixes, and among those might also be some security improvements and fixes.

The PrestaShop team is always hard at work, making sure the software is safe and secure for both merchants and customers. But human mistakes happen, and new versions are here to fix them.

Upgrading PrestaShop is not always easy, since an e-commerce solution is such a complex piece of work, coupled to the fact that you need to put your store in maintenance mode for a couple of hours (or maybe a full day, depending on the complexity of your installation), which means less customer visits and thus less sales. But trust us, you'd rather spend some time making sure your store and its modules are up to date, rather than having to deal with hackers and loss of data.

The 1-Click Upgrade module (available natively) will help you, but always always always make back-ups of both your files and your database, should anything wrong happen with the upgrade.
If an automatic upgrade makes you queasy, you can also use the manual upgrade method (but it takes a lot more time).
In both cases, test first on a copy of your store, then make sure that copy did get properly upgraded before you upgrade your main site.

Set up a server-side password for your back office folder

You can build a secondary password protection in order to further limit the access to your PrestaShop back office folder (`http://www.example.com/`**`admin123456/`**, for instance).

Establishing a basic authentication on the back office folder requires adding a `.htaccess` and a `.htpasswd` file. Both are simple text files without a name, only an extension.

In Windows, you cannot easily create a file with no name. There are two easy ways to solve this:

- You can name the file `htaccess.txt`, then upload it to your FTP server, and there rename it to `.htaccess`.
- A Windows trick is to name the file with a dot on each side of its name: "`.htaccess.`". Windows will automatically change the name to the correct "`.htaccess`".

One of the aims of the `.htaccess` file is to protect your folders and all of its sub-folders (read http://en.wikipedia.org/wiki/Htaccess). **It only works on Apache servers**. Make sure your web server is Apache before creating a `.htaccess` file: ask your host!

To protect a folder, you need to put those two files at the root of that folder (for instance, through your FTP software, in `/var/www/prestashop/admin123456` or maybe `/public_html/prestashop/admin123456`).

Here is an example content for your file:

```
AuthUserFile /var/www/.htpasswd
AuthName "Prestashop Admin Access"
AuthType Basic
Require valid-user
Options -Indexes
```

Explanation:

- `AuthUserFile`: Shows the path to the file containing allowed users and their passwords. `.prestashop_admin` is a text file.
- `AuthName`: Defines the message to show when the authentication window pops up.
- `AuthType`: Defines the authentication type.
- `Require`: Requires users to log in in order to access the content. `valid-user` enables multiple users to connect and access the folder.
- `Options`: Defines the folder's options. `-Indexes` disables automatic generation of a directory index if no index file is available.

Here is a sample content for the `.htpasswd` file, with a login and a password:

```
login1:$apr1$/wJeliK8$e9OzgRaVL8J8wSsFBXjor1
login2:$apr1$yV65Kqqz$cFt3sV2.Q7hhLRRUJDo5a/
```

This file contains logins and hashed password who are allowed to access to the folder.
You can generate both files on our own generator: http://build.prestashop.com/tools/htaccess-generator-protect-your-prestashop-backoffice/
To hash password manually, you can use a `.htpasswd` file generator: http://aspirine.org/htpasswd_en.html.

It is strongly recommended to put this file into a directory that is inaccessible to your web applications, so before the `/openbase_dir` folder. It prevents `.htpasswd` file injection, in case one of yours web applications is vulnerable.

It is also possible to perform IP and domain restrictions using your `.htaccess` file:

```
Order Allow, Deny
Deny from all
Allow from .example.com
Allow from 127.0.0.1
```

However, you **should not** put this kind of directive:

```
<LIMIT GET POST>
Require valid-user
</LIMIT>
```

Use stronger passwords

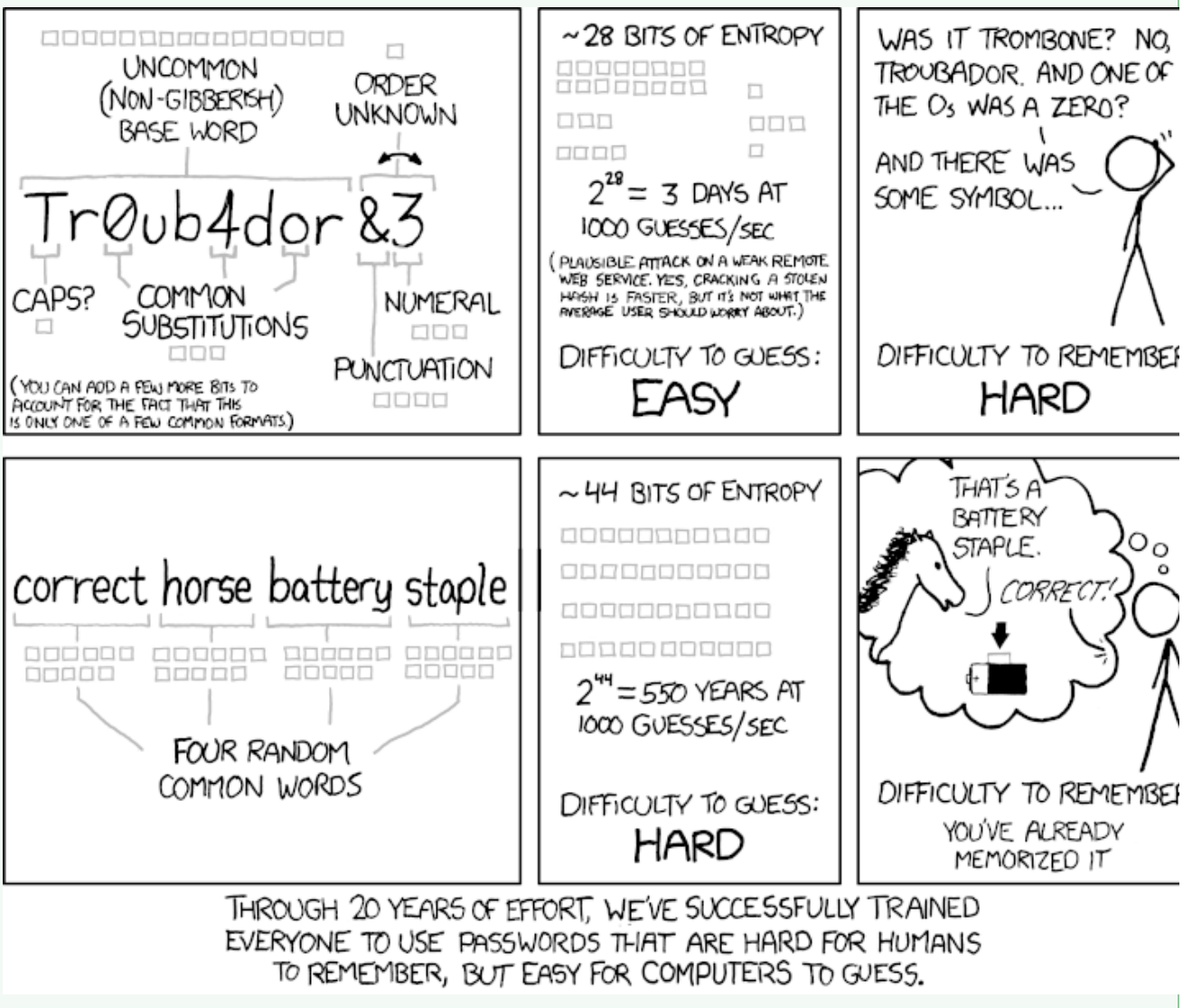
Pick a complex password, by mixing letters, numbers and even punctuation marks, such as "5r3XaDR#". You can and should use a password generator, such as Symantec's (http://www.pctools.com/guides/password/) or GRC's (https://www.grc.com/passwords.htm).

Safer than a password: you can use a passphrase. Not only is a passphrase easier to remember, but it is also much harder to crack, even when the hacker is using automatic tools (brute force attack or dictionary attack).

A passphrase only needs to be long and easy to remember for you. Any popular saying should do ("Don't Throw the Baby Out with the Bathwater"), but an absurd phrase will have even less risk of being discovered by a hacker. For instance, "Many reckless drivers confuse tractor with r ecord sleeves".

There are some good passphrase generators online, which help you get a unique phrase for you only. For instance: http://passphra.se/ or http://www.fourmilab.ch/javascrypt/pass_phrase.html.

PrestaShop's passwords are not limited in either number of characters or types of characters.



(original comic by XKCD)

Change the name of your back office folder

Rename your `/admin` folder after the PrestaShop installation. This is a must, and you actually cannot access your PrestaShop administration if you haven't performed that change. Make sure to pick a really unique name, ideally a mix of letters and numbers, such as `/my4dm1n` or anything you can remember.

Delete useless default files

1. Always delete the `/install` folder after having installed or updated PrestaShop.
2. Always delete useless files from production server:
    a. The README.md file.
    b. The CONTRIBUTING.md and CONTRIBUTORS.md files.
    c. The `/docs` folder and all its content.

Block direct access to your templates

Forbid access to your theme's files/templates, using a `.htaccess` file with the following content:

```
<FilesMatch "\.tpl$">
order deny,allow
deny from all
</FilesMatch>
```

Update your server software

Your applications' PHP code is the only vulnerable path to your server. It is therefore strongly recommended to always update your server's applications: PHP, MySQL, Apache and any other application on which your web hosting.