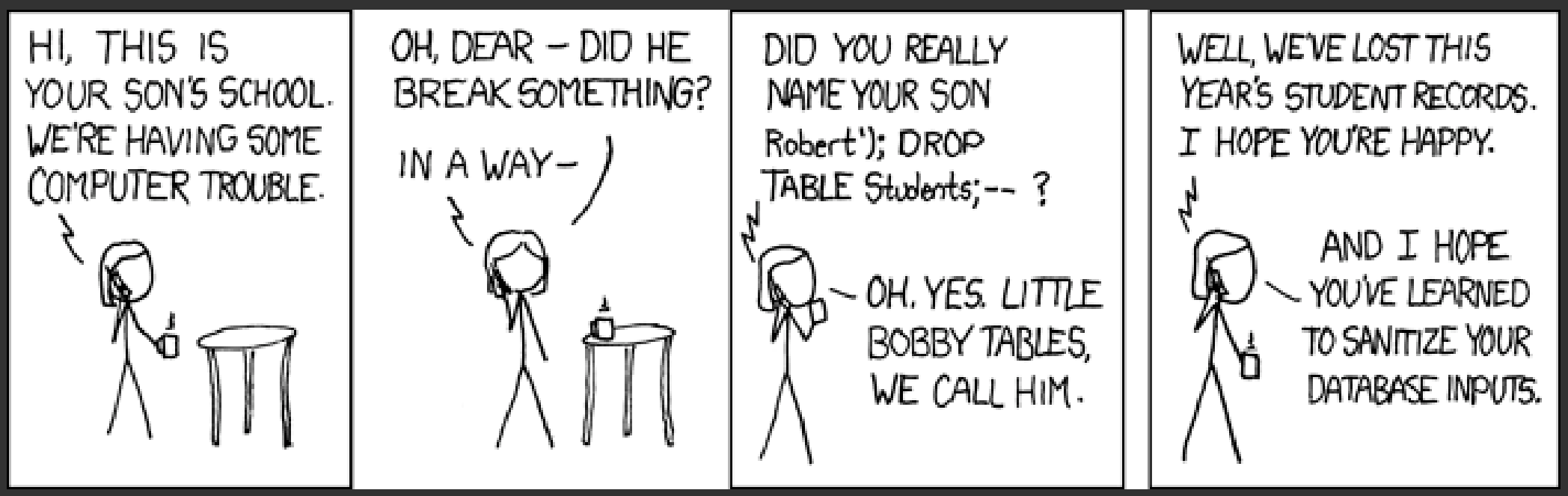
MOST COMMON SECURITY VULNERABILITIES WHEN CODING IN WORDPRESS

Whether you want to tailor your Wordpress experience, create a plugin or a theme, you need to write code. Here are the common security vulnerabilities you need to avoid when adding PHP code to your Wordpress installation.

INCORRECTLY MANAGED USER IDENTITIES & PERMISSIONS

Whenever a user does an action or tries to accesss a ressource, you have to verify their identity and if they authorized to do so. You can get info on user using internal Wordpress functions like wp_get_current_user() to decide if they are allowed access the data or not.

SQL INJECTIONS



Whenever a user enters data and sends it to Wordpress, you are vulnerable to a SQL injection. When this string is entered in a database using a SQL Query by a malicious user, it can hijack the Query to do whatever it wants.

CROSS SITE SCRIPTING

Name :

Hackerman

Comment :

This is an XSS<script>do malicious js thing</script>

Submit

Hacker sends a script tag in the form of a vulnerable website. The script is then stored in the database.

Vulnerable website

Comments :

Guy #1 :

This is a cool website

Hackerman :

This is an XSS

When the page is rendered by a user, the computer executes the javascript code

XSS is the most commonly used attack today. It consists of a malicious user adding Javascript or other HTML tags to a vulnerable website or page to manipulate other users' browsers behaviour.

CROSS-SITE REQUEST FORGERY

Malicious website :

Name :

Innocent Guy

Comment :

Wow! I love this site

Submit

Link:
https://www.my-bank.com/transfer-money/submit.php?from="Innocent Guy"&to="Hacker"&sum="10000"

Vulnerable website
https://www.my-bank.com/transfer-money

From account :

To account :

Sum :

Submit

Link:
https://www.my-bank.com/transfer-money/submit.php

The website or email contains a hidden link that unknowingly redirects the user to a legitimate website in the same browser.

Without completing the form, the money transfer request is automatically sent from the users personal my-bank account to the hackers's.

The CSRF attack consists of sending a request from the user's computer unbeknownst from them by concealing a link within a malicious website or email. To prevent this, you can add a nonce to the url using the function wp_nonce_url().