BEST PRACTICES FOR WOORDPRESS SECURITY

Jason Yingling (@jason_yingling)

Cross Site Scripting (XSS)

Attackers can inject malicious scripts into a page that can be used to steal data from users.

Common Cause(s)

Bad / outdated plugins

Non-sanitized code

Not validating input data

Brute-force Attacks

Attackers run scripts to try to crack logins by trying known user and password combos.

Common Cause(s)

Weak passwords

Unlimited login attempts

Tip

https://haveibeenpwned.com/

SQL Injections

Attackers inject SQL into the site's database, typically through plugin vulnerabilities, allowing them access to the database to do as they please.

Common Cause(s)

Bad / outdated plugins

Non-sanitized code

Not validating input data

DDOS

Multiple systems are used to target a single site causing it to overload and go down

Common Cause(s)

Lack of firewalls

PROTECT YOUR LOGIN

Don't use admin username

Use strong passwords

Hide the login page

Two Factor Authentication

Limit Login Attempts

Plugins

iThemes Security

WordFence

Limit Login Attempts reloaded

Pro Tip

Use site specific emails: example+domain@gmail.com

KEEP THINGS UP TO DATE

Keep WordPress core updated
Keep plugins and themes updated
Use the latest version of PHP

Pro-tip

Used managed WordPress hosting

PICK GOOD PLUGINS

Trusted sources

Recent updates

Active installs

Check the support forums

More

Picking Good WordPress Plugins

TAKE FREQUENT BACKUPS

Database and file system backups
Test backups from time to time

Store the backups off server

USE A FIREWALL

Protect against known attacks Helps with DDoS mitigation

Services

Cloudflare

Sucuri

USEFUL PLUGINS

iThemes Security

WordFence

Limit Login Attempts Reloaded

SECURITY RESOURCES

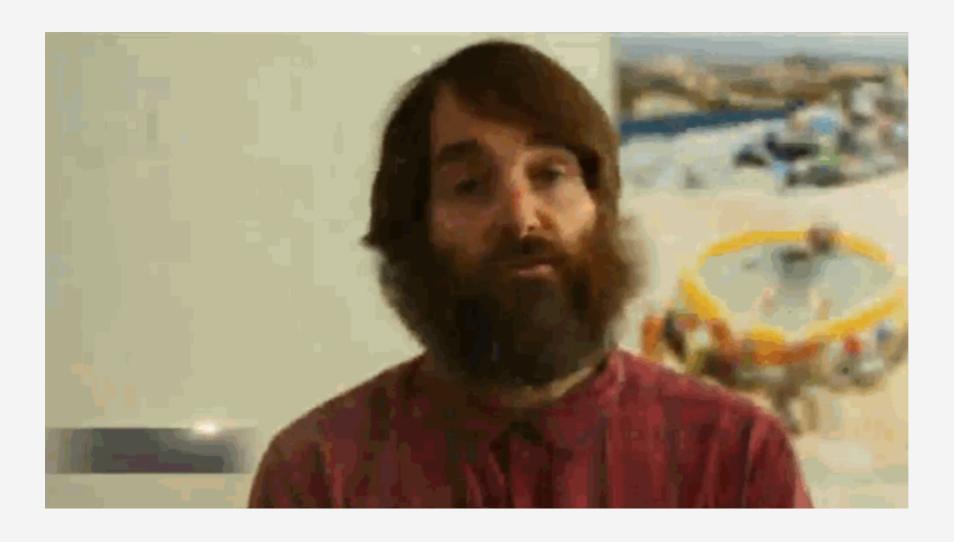
Kinsta WordPress Security Guide

Wordfence - How to Prevent Cross Site Scripting Attacks

wpbeginner - The Ultimate WordPress Security Guide

WordPress Security: An Introduction to Hardening WordPress

SECURITY QUESTIONS QUESTIONS



Twitter: @jason_yingling

Email: jason@jasonyingling.me