Internal Proposal: Make Security / Vulnerability Team

[ATTENTION]
This proposal is **a first version** that has not detailed all elements and aims to serve as a preliminary guide to a final proposal.
If you believe something is missing or that it is necessary to specify any detail, please comment it.
[/ATTENTION]

WordPress has many tools and management related to security, whether it's for the websites and plugins managed by the Community through HackerOne, or the Core-Security team to review possible bugs in the WordPress core code.

In terms of documentation, security has always been handled by several teams, providing suggestions for each of the parties involved, both in Core, Plugins, Themes, and Hosting.

Nevertheless, the security of plugins and themes has always been the responsibility of their respective developers.

What happens when a vulnerability is found?

There are many possibilities and branches when it comes to vulnerabilities. The simplest one is the vulnerability in the WordPress core, which is directly managed by the Community through the Core-Security team.

However, it's not as straightforward when it comes to plugins and themes, as the Community provides review and management tools, such as the ability to disable a plugin or theme from the directory, but it doesn't have the tools or resources for full management. It shouldn't be something that the Community should handle on its own, although it is partially responsible for it.

When a vulnerability appears in a plugin, the Community and Teams initiate a process to try to notify the developer for them to review and fix it, with or unsuccessfully, but the responsibility lies with the developer.

Another approach is through standard systems like the National Vulnerability Database (NVD), which provides commonly known identifiers known as CVEs.

In these cases, a series of organizations are responsible for providing information to this database and maintaining it.

In the case of WordPress, there are three well-known organizations (Jetpack Protect, Patchastack, and Wordfence) that have this capability, although they are not the only ones. When a notification reaches any of them, they may try to contact the developer directly or do it through the Community teams. If, after a certain period of time, the vulnerability is not fixed, or even if it is, the information is published publicly and made available to everyone, either in real-time or with a certain delay.

Shared responsibility

While it is true that the primary responsibility of the Community is to keep WordPress secure, it is also true that many elements at various levels (such as the use of PHP or performance analysis) are not solely focused on the WordPress core and default themes, but also consider the extensibility of the platform.

Often, when you hear “WordPress is not secure”, it is not referring specifically to WordPress itself but rather to its extensibility, meaning plugins and themes.

In the past, attempts have been made to create some kind of platform for the Community to take full responsibility, but it was not feasible because, until recently, there were no relatively stable systems or companies maintaining that information. Additionally, other databases attempt to aggregate that data to provide a minimal service to all WordPress users.

WordCamp Europe 2023

Thanks to the work of the Hosting Team, conversations have been held with various stakeholders in the security ecosystem during WordCamp Europe 2023. It also coincides with the creation of the new Plugins Review Team, which may have more resources to better manage this situation.

In discussions, @JavierCasares, as part of the Hosting Team, received some feedback and engage with Patchstack, plugins, and security companies (such as WPSec, WPGuardian, or Really Simple SSL). Considering the increasing amount of information related to plugins and themes, and within the Five for the Future project, an opportunity has arisen to open a debate and conversation so that the Community can have a minimum level of security information that can be maintained by multiple sources.

Discussions have also taken place with the new Plugins Team and the Core-Security team, and initially, it seems like an interesting idea that can be pursued in this matter.

NOTE: It should be noted that @JavierCasares created an idea (WPVulnerability.com) for this project in early 2022 because the Community was not taking charge, based on previous occasions. That database is public and does not require any special access for anyone who wants to use it.

Security-Vulnerability Team

Based on this entire situation, the option arises for these companies to participate through the Five for the Future project by donating and maintaining certain public information that can be utilized internally by both the WordPress.org/Meta website and the WordPress Core. Additionally, basic security information would be made public.

Furthermore, this system would allow these companies to continue their business while improving the entire ecosystem, enhancing security not only for WordPress itself but also for plugin and theme updates.

The team would be composed of two groups. These groups can consist of different individuals from existing teams or form an entirely new team.

The differentiation between the two groups is temporary, with the disclosure of a vulnerability serving as a turning point.

Vulnerability Group

This working group would be responsible for maintaining the vulnerability database. These vulnerabilities would exclusively include those that have been previously published in some way and are specific to themes and plugins.

This database would operate based on the combination of the following elements:

```
slug + operator + version + fix
```

An example could be:

```
plugin-name <= 1.0.0 nofix
```

In addition to this basic information, it is necessary to use and unify information from different sources, using unique identifiers and creating unique identifiers specific to the Community.

From a technical standpoint, this system could be a Custom Post Type with multiple Custom Fields, and it could also utilize users as sources of information, as well as the API system.

The maintenance of this database could be done through a closed REST API, accessible only to verified sources approved by the Vulnerability Group.

This way, providers like Jetpack Protect, Patchstack, Wordfence, or others could update the data for the vulnerabilities they are responsible for, with the plugin and theme teams serving as other sources. Additionally, other potential participants with the knowledge to contribute could also be involved.

Although this project can be carried out in various phases, it would initially allow the Community to have its database for the use of the Teams, then for Meta, followed by WordPress, and finally open it to the Community for public access (not necessarily in that order).

Internal use by the teams can serve to temporarily block a plugin or theme from the directory if a vulnerability is deemed serious, or simply as a precautionary measure.

The use by Meta could enable displaying historical information about a plugin or theme's vulnerabilities on its listing page, or indicate if there is an active vulnerability at the moment. Additionally, if a plugin or theme is closed for security reasons, direct links to the information sources explaining the details can be provided.

For WordPress itself, through the existing API, it would allow for expanding information on whether a plugin may have any active vulnerabilities directly from the Site Health or the listings of plugins and themes.

Opening the information to the Community would allow anyone to use the open data, enabling the creation of plugins or tools that leverage this information to enhance the security of installations.

Security Group

The security group is the preliminary stage before the disclosure of a vulnerability. Currently, there are already some issues with those who discover a potential vulnerability and need to communicate it to the developers.

Although it is possible to automate the process without revealing data from either party, it is a delicate matter that requires some minimal supervision, either directly from a group of knowledgeable individuals, potentially creating the Plugin Security Team and the Theme Security Team, or by utilizing the same resources as the existing Review Teams, relieving Meta from this responsibility that they currently handle.

Security has its own global cycles in the industry, so efforts should be made to understand and adapt to them, something that is currently not being done.

This would help enhance the perception of security, making WordPress a more secure tool overall.

Documentation

Furthermore, this team could take charge of centralizing the security documentation that currently falls under the responsibility of the Hosting, Core, Plugins, and Themes teams, primarily, and is spread across almost all projects within the Community, as well as the documentation for end users.

Summary

Undoubtedly, this project is very delicate due to its implications, but it is also very fascinating for the future of the WordPress project. It is highly likely that many companies would want and be able to contribute to maintaining the project, both in terms of tools and personnel, as there are several open-source options available. They could also assist in including profiles within the WordPress Community that are currently not part of any team and may therefore be unwilling or unable to participate.

Having a global Security team for the WordPress project could also instill a greater sense of security in end users, knowing that the Community itself is dedicating resources, alongside hosting companies and other stakeholders, to ensure certain minimum security standards are met at all levels that affect a project.

Proposed by: @JavierCasares

Reviewed by:

Special thanks to: @crixu, @davidperez, @desrosj, @frantorres, @mrfoxtalbot, @oliversild, @pharar.