Moodle MootDACH 2025
TH Lübeck | 2025-09-04

Securing Moodle

A brief introduction to web application security and
low-hanging fruits anyone should harvest

Who

Niels Gandraß
Moodle Administrator & Developer

Begging you to upgrade your Moodle and PHP since 2016 ...

Philipp Kropp
Moodle Administrator & PHP Developer (Symfony)

Security dive into Moodle last years Coderunner / Jobe deeper dive 2025 and PHP since 2002

Alexander Bias
Solutions Architect @ ssystems

Moodle Expert & Open Source Enthusiast since 2008

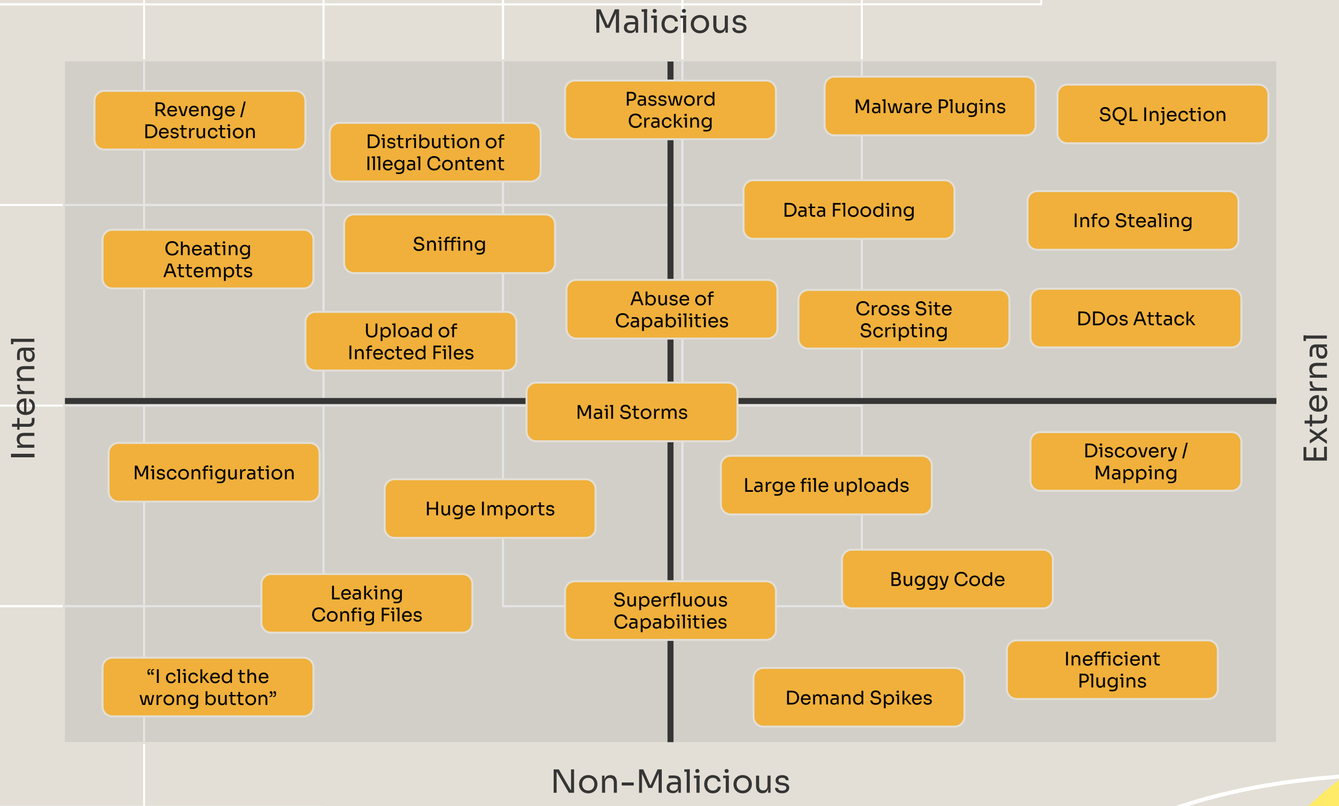
Disclaimer

This talk simply can not be complete, but we try to lead you in the **right direction** and give you **valuable resources**

Is your Moodle up-to-date?

We will get back to this at the end of the talk ...

Understanding potential problems



Take the **four fields** more as a **categorization** than a ranking against each other!

Approach | Establishing a strategic mindset when it comes to security

1

Knowledge

Knowledge is key.

Without being aware of what can, or worse, is happening to you, you won’t even notice that something is bad.

2

Immediate Actions

Lock all doors that should be locked.

Take actions to secure yourself or stop an ongoing attack.

3

Prevention

Implement measures that prevent you from being vulnerable in the future.

Establish a routine and do not rely on something that someone did somewhere some time ago.

4

Emergency Plan

Having a plan what to do for when … *“shit hits the fan”*

Do not only think about defense but also about recovery.

Making your whole system **more resilient** helps you with both **external** and **internal** problems.

Simple actions you can take right now!

1 **Securing Moodle**

Most effective configuration changes and best practices that will increase your application security.

2 **Hardening the Server**

Simple changes on web server level that help you stay secure and prevent misuse of your systems, with or without malicious intent.

3 **Staying ahead**

How to keep in the loop and get notified when bad things are happening or your action is required.

Moodle

The obvious actions you can take right now to secure your instance

Stay up-to-date

Update your Moodle instance and your plugins. Check for end-of-life (EOL). Schedule regular maintenance windows to develop a routine.

🌐 Upgrading & Releases

Restrict capabilities

Check roles, capabilities and assignments of global rights. Restrict to the absolutely necessary. Disable debug messages.

🌐 Roles and permissions & Force users to login & Automatic user delete & Debugging

Authentication

Enforce TLS for all pages. Disable unused auth plugins and the guest user. Use captchas and throttle login attempts.

🌐 Managing authentication & Account lockout & Transitioning to HTTPS

Restrict user content

Never trust user data. Limit the max file upload size and set usage quotas. Scan uploads via ClamAV and deny malicious files.

🌐 Maximum uploaded file size & User quota & Antivirus plugins

Reduce plugins and disable unused functions

Use as few plugins as you can but as many as you must. Production is not a test area. Disable unused core functions.

🌐 (Un)installing plugins & Advanced features

Enable MFA

Enforce strong passwords and MFA / 2FA for at least all privileged users. Use offline tokens if possible.

🌐 Multi-factor authentication & Password policy

Moodle

The not so obvious steps one should know about …

Upgrade key

Defining a secret key that needs to be entered prior to performing Moodle upgrades to deny unauthorized changes.

🌐 Upgrade key

Disable web plugin install

Restrict plugin installation via web upload. Currently being exploited.

🌐 Preventing installing plugins from within Moodle & Web plugin install exploitation

Deny running cron via web

The Moodle cron should run via CLI only. Prevent running cron via web and let your server run cron automatically for you instead.

🌐 Cron

Force core and plugin settings

Freeze critical core and plugin settings inside your *config.php* Deny any changes to utility binaries paths. Serve libraries locally (e.g., MathJax).

🌐 Forcing values of admin settings & MathJax Config

Add salt and pepper to your passwords

Moodle salts passwords by default. However, adding pepper must be enabled! Having both makes reversing password hashes way harder when compromised.

🌐 Password salting and peppering

Track and control source code changes

Do not treat your Moodle source code directory as a living place. Use version control or reproducible deployments to prevent unnoticed changes.

🌐 Download Moodle via Git & Git for Administrators

Simple actions you can take right now!

1 **Securing Moodle**

Most effective configuration changes and best practices that will increase your application security.

2 **Hardening the Server**

Simple changes on web server level that help you stay secure and prevent misuse of your systems, with or without malicious intent.

3 **Staying ahead**

How to keep in the loop and get notified when bad things are happening or your action is required.

Server

The most common things you should do right now within your web server environment

Stay up-to-date

Use up-to-date software for web server, host OS, and especially PHP. Know the age of your software and migrate when it is EOL. Enable auto updates if you like *(the thrill)*.

🌐 PHP Releases & Moodle Software Compatibility

File Permissions

The web server must only write to the Moodledata folder. Everything else must be set to read-only. Let nobody but the admin touch the config files.

🌐 Setup secure file permissions & Linux file permissions explained

Restrict Resources

Limit your system resources like memory, time, and sockets globally and per request. Use as few as you can and as much as you need. Move large jobs to the CLI instead of raising limits.

🌐 PHP resource limits & Disable dangerous PHP functions & Nginx FastCGI timeouts

Serve Public Files Only

Have only the required files in your webroot. Deny access whenever you can! Stuff like .git, vendor, .htaccess, or composer.lock contain critical information and should never be served to the public!

🌐 Deny web access to critical Moodle files: Nginx & Apache

Check Guidelines

Read up on security guidelines and best practices for the software you are using. If unsure, try to ask someone.

🌐 How to secure a Linux server & Moodle security & Common web application security risks & Moodle Community Matrix (#community:moodle.com)

Server

We are approaching wizard theretory …

Privileges and Isolation

Restrict database, ClamAV, Maxima (STACK) and other services to local access only and use private networks if possible. Isolate dangerous services, e.g. Coderunner. Block any external access.

🌐 **Linux firewalls** & Set database bind address: **PostgreSQL** & **MariaDB**

Hardening your OS

Take steps to harden your base system. Enforce key-based SSH-authentication, restrict sudo and login permissions, remove unused features, use separate users for each person, …

🌐 **Ansible OS hardening collection** & **Linux server security tips**

Auto-Ban Attackers

Let fail2ban monitor your logs and ban every IP that has multiple failed auth attempts or shows other malicious behavior.

🌐 **The Fail2Ban project** & **How to secure SSH with fail2ban**

Monitoring and Intrusion Detection

Use a monitoring solution to check system stability and metrics that can indicate malicious behavior. Regularly scan your servers and logs for indicators of compromise (IoC).

🌐 **Icinga monitoring** & **Lynis** & **rkhunter** & **Snort**

Read-only and Reproducible

Mount web server directories as read-only whenever possible. Create CI jobs that build your Moodle source tree in a reliable and reproducible way.

🌐 **Mounting filesystems as read-only** & **Get started with GitLab CI**

Simple actions you can take right now!

1 **Securing Moodle**

Most effective configuration changes and best practices that will increase your application security.

2 **Hardening the Server**

Simple changes on web server level that help you stay secure and prevent misuse of your systems, with or without malicious intent.

3 **Staying ahead**

How to keep in the loop and get notified when bad things are happening or your action is required.

Staying ahead

How to keep in the loop and
know when your action is required

1 **(Moodle) Security Announcements**

- **Register your site** to receive **Moodle security announcements** directly via email
- Check the security mailing lists of your host OS

2 **Security Checks & System Status**

- Execute the **security checks** and look at the **system status report**, but take it with a grain of salt (e.g., “XSS trusted users”)
- Make sure admin notification emails reach you and check system logs if something seems off

3 **Keep in touch with the community**

- If you are unsure about something feel free to ask the community!
- Dive into the **Moodle forum**, **Moodle tracker**, **global community matrix**, and the **Moodle an Hochschulen e.V. matrix** (German)

And last but not least …

You all waited for the upcoming slide ;)

Have a Backup!

Backups

Everyone tells you about it, but do you have it?

Make sure, that ...

- You have **automated** your backups
- You get **notified** whenever they fail
- You have **included everything** that you need
- You can actually **restore** them *(try it before you need to!)*
- You **do not lose** your backups in case of an emergency, cyber attack or system failure

And if you want the bonus points ...

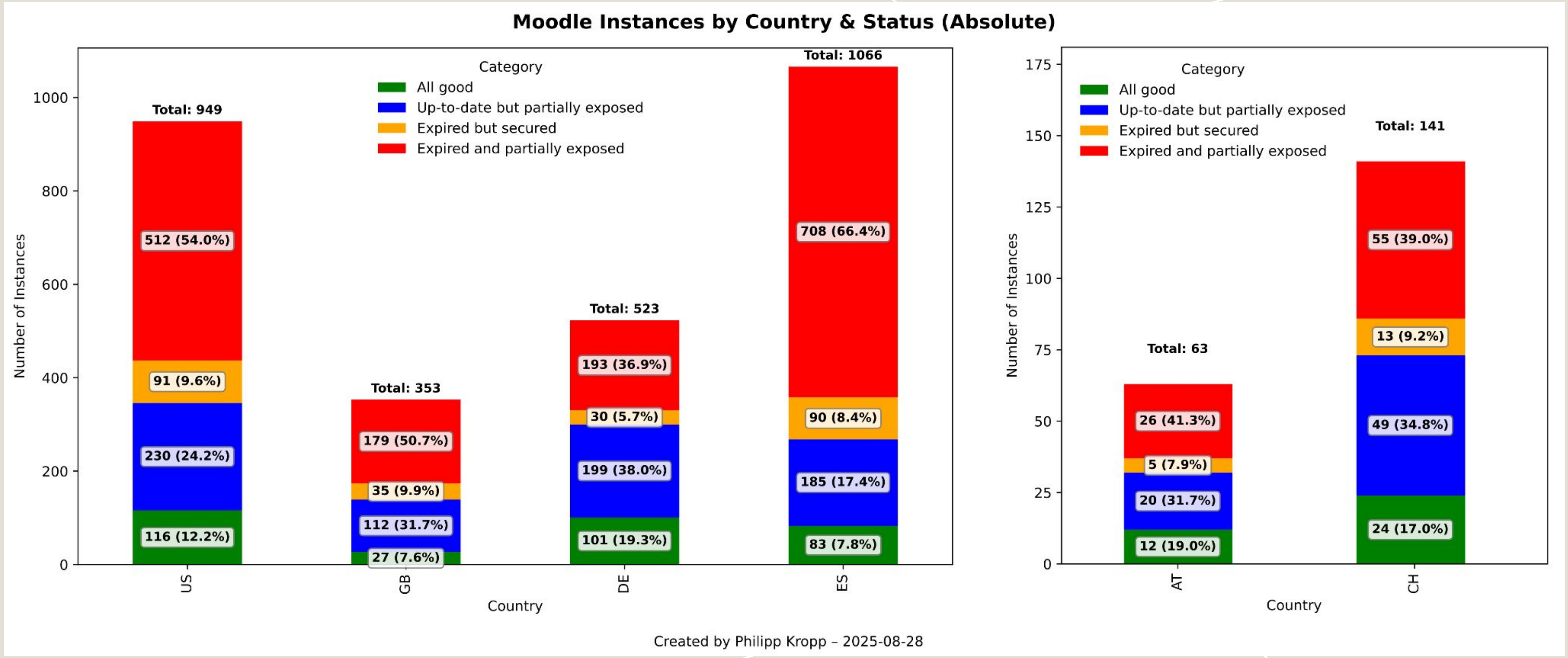
- You can perform **partial restores**
- You have **three copies** of your data on **two** different storage **types** and at least **one** backup location is **off-site**

Is your Moodle up-to-date?

We’ve checked for you ;)

- Philipp conducted a security scan of all publicly listed Moodle instances within various countries
- Tested for security.txt, composer.json, git directories, and more
- The data was collected in August 2025
- For in-depth details and tips how to act join Philipps session!

Is your Moodle up-to-date?

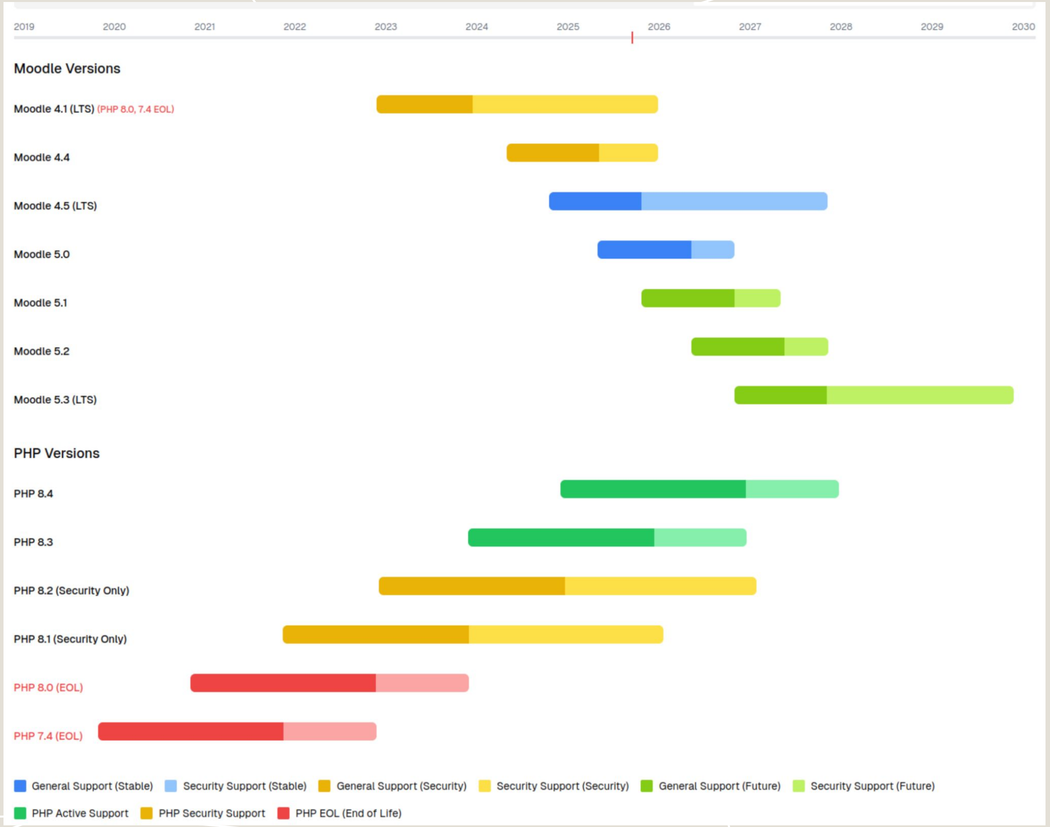


Is your Moodle up-to-date?

- **Moodle 4.1** (LTS) general support ended on 11. December 2023 and will be without security fixes (EOL) after **8. December 2025 !**
- **PHP 7.4** is EOL since **28. November 2022 !**
- If you are still running on old versions plan your upgrade to Moodle 4.5 (LTS) and PHP 8.4 now!
- Moodle 5.3 (LTS) will release on 5. October 2026
- If you can not avoid using it, there are some unofficial security backports for old PHP versions

🌐 Moodle Releases and EOL Dates

🌐 PHP Releases and EOL Dates



Now it’s your turn.

Thanks for your attention and stay safe!

Get the Slides

Any questions? Ask away!

Also feel free to contact us afterwards.

- Niels: `niels.gandrass@haw-hamburg.de`
- Philipp: `pk@informatik.uni-kiel.de`
- Alexander: `abias@ssystems.de`