Fehleranalyse und Debugging

Im Folgenden wird das Vorgehen zur Identifizierung von Systemproblemen beschrieben.

1. Laufen Arbeiten an anderen erforderlichen Systemen oder Infrastruktur parallel? Z.B. Netz, Storage, VM oder IDM
2. Server-Überwachung checken:
    1. System überlastet? Storage voll gelaufen? CPU-Last?
3. Webserver + DB überprüfen
    1. Läuft der Webserver?
    2. Ist genügend Speicherplatz frei? `df -h`
    3. Läuft die Datenbank und hat genügend Speicherplatz?
4. Was wurde zuletzt geändert; mögliche Auslöser für Probleme? Changelog sichten.
5. Cronjobs überprüfen
    1. Läuft der Linux-cron korrekt?
    2. Werden die "Geplanten Vorgänge" sauber abgearbeitet? `/admin/tool/task/scheduledtasks.php`
    3. Cron oder einzelne Tasks ggf. manuell ausführen?
6. Log-Dateien analysieren + Moodle-Konfiguration
    1. Siehe auch Moodle Webserver
    2. Alle Systemklicks nachvollziehbar unter: `/report/log`
    3. Konfigurationsänderungen: `/report/configlog/index.php`
    4. Login via Debug-user ([$CFG->debugusers = '2';]), der für alle Aktionen im "Developer-Modus" agiert und Fehlermeldungen direkt im Browser anzeigt
7. Moodle-Dateien
    1. Wurden Plugins aktualisiert?
    2. Haben die Moodle-Dateien die erforderlichen Rechte?
8. Viren-Scanner
    1. Blockiert der Viren-Scanner etwas?
    2. Gibt es massive Performance-Beeinträchtigungen durch Viren-Scanner?
9. Netzwerk + Firewall
    1. Sind alle benötigten Freigaben erteilt auf Webservern und ggf. Loadbalancer?
    2. IPv6/v4; funktioniert beides? Z.B. testen mit curl -4 https://moodle-an-hochschulen.de/
    3. Netzwerkverkehr analysieren, z.B. via `tcpdump -i any -nn host [Moodle-Server]`

       tcpdump-Erklärungen: https://xinux.net/index.php/Tcpdump_grundlagen +

       https://geekflare.com/tcpdump-examples/

       in eine Datei schreiben: `-w [Dateiname]`

Debug-Modus aktivieren mittels config.php

```
//=======================================================================
// 7. SETTINGS FOR DEVELOPMENT SERVERS - not intended for production use!!!
//=======================================================================
//
// Force a debugging mode regardless the settings in the site administration
@error_reporting(E_ALL | E_STRICT);   // NOT FOR PRODUCTION SERVERS!
@ini_set('display_errors', '1');      // NOT FOR PRODUCTION SERVERS!
$CFG->debug = (E_ALL | E_STRICT);     // === DEBUG_DEVELOPER - NOT FOR PRODUCTION SERVERS!
$CFG->debugdisplay = 1;               // NOT FOR PRODUCTION SERVERS!
```

siehe auch: https://docs.moodle.org/en/Debugging#In_config.php

Bekannte Errors

Ungültige Kursmodul-ID (core_error invalidcoursemodule)

Wenn ein Kurs nicht mehr regulär aufgerufen werden kann und stattdessen die Fehlermeldung "*Ungültige Kursmodul-ID*" (lang-string *invalidcoursemodule* in *core_error*) angezeigt wird, liegt dies oft an Fehlern in der Reihenfolge von Kursabschnitten oder Kursmodulen. Dann hilft meist die Ausführung des CLI-Skripts fix_course_sequence.php (ohne ***--****fix* nur Prüfung auf Fehler):

```
php admin/cli/fix_course_sequence.php -c=* --fix
```

Siehe auch Moodle-Docs Reihenfolge der Kursmodule reparieren

Weiterführendes

Debugging allgemein: https://docs.moodle.org/en/Debugging

Debug-Informationen konfigurieren: https://docs.moodle.org/en/Debugging#Debug_messages

---

Autor: Klaus Steitz, Technische Universität Darmstadt

Version #8
Erstellt: 2025-09-09 10:38:41 UTC von Klaus Steitz
Zuletzt aktualisiert: 2026-06-13 07:53:56 UTC von Klaus Steitz