Fakultät Informatik

CYBRAIL

IT-Projekt im Studiengang Bachelor Informatik

vorgelegt von

Mattis Krämer, Robin Rosner, Pascal Blank

Erstellungssemester SS2024 - WS2425

Betreuer: Dr. M. Geier

Inhaltsverzeichnis

1 Einleitung

CYBRAIL ist ein Tool, das es ermöglichen soll, mögliche Betrugsversuche in Online- oder Präsenzklausuren am Computer zu erkennen. Dabei liefert es eine nachvollziehbare Begründung anhand von verschiedenen Indizien, die während einer Onlineklausur in Logdateien gesammelt und später durch dieses Tool ausgewertet werden können.

1.1 Safe Exam Browser

Der Safe Exam Browser (SEB) ist ein abgesicherter Browser, der speziell für Prüfungen an Schulen entwickelt wurde. Seine Funktionalität ist stark eingeschränkt, um die den Studierenden zur Verfügung stehenden Hilfsmittel erheblich zu beschränken. Der SEB wird auch an unserer Hochschule eingesetzt, weshalb der Fokus dieses Projekts auf diesen Browser ausgerichtet ist. Zu den Funktionen des SEB gehören unter anderem Prüfungen, ob unerlaubte Programme im Hintergrund laufen, ob der Bildschirm geteilt wird, und es wird verhindert, dass Studierende die SEB-Umgebung verlassen, andere Programme öffnen oder Copy-Paste verwenden. Somit bildet der SEB eine solide Grundlage, um faire Prüfungen zu ermöglichen und Betrug zu verhindern.

1.2 Cheating with SEB

Jedoch hat jedes System auch gewisse schwächen.

1.2.1 Virtuelle Machine

So konnten frühere versionen des SEB mit dem einfachen Austausch einer Programmdatei so verändert werden, dass dieser nun auch ein einer Virtuellen Maschine (VM) startet. Dieses wird

eigentlich durch mehrere komplexe checks unterbunden und würde den start in einer VM abbrechen. Dieser Cheat war mit einer ca. 5 minuten Google suche einfach und unkompliziert anzuwenden. Durch diesen ist es dem zufolge mölgich den SEB in eine Fenster oder auf einem 2. Bildschirm zu haben während man auf dem Hostsystem - alo dem system was die VM ausführt - die lösungen googelt, aufnimmt, in einem Videocall ist oder ähnliches.

Trotz all dem gab es auch hier schon gewisse indizien in diversen log files, welche auf so einen misbrauch hinweisen. Z.b. wird geloggt wenn sich die display auflösung verändert, dies würde passieren wenn die Fenster größe der VM angepasst wird oder diese zu Vollbild wechselt. Auch verfügt der SEB über einen Integritätscheck um Modifikationen festzustellen.

Beide diese Indizien werden jedoch lediglich ein eine Logfile geschrieben. Ohne auswertung und einsammeln der Logfiles wird man dies all Prüfer oder Aufsich also nie erfahren.

1.2.2 Auto Typing

Ein anderer angriffsvektor ist quasi undedektierbar: Copy-Paste durch automatische eintippen von zeiche. Dies könnte entwer per auf dem Gerät laufender Software geschen, welche nicht durch den SEB geblockt wird; hier würde nur eine Whitlist helfen, da man so ein Programm ja beliebig benenen könnte oder aber es könnte ein Hardware dongel verwendet werden, welcher z.b. Text von einer anem anderen gerät schnell dort eintippt. Somit kann auf einem 2. Gerät die Lösung gesucht oder unter Studenten geteilt werden und dann schnell ung bequem automatisch eingetippt werden.

1.2.3 SEB Server

Alle infomrationen die in Logs gespeichert werden liegen zunächst auf dem Client des Users, also unzugänglich um auffälliges verhalten festzustellen. SEB liefert hierdoch einen optionalen SEB-Server. Dieser kann in der Hoschschule aufgesetzt werden und nach richtiger konfiguration des servers und der Konfigurationsdatei für Clienat - welche vor start der Klausur direkt von Moodle geladent werden kann - Logdateien von den Client empfangen. Dies ist jedoch seine ganze Funktionalit, es findet kein automaische auswertung der Logs stat, hier müsste ein Mensch

versuche - vermutlich stichproben artik - auffälligkeiten zu finde. Dies ist jedoch sehr schiwierug auser man ist vertraut mit der Funktionsweise des SEBs.

1.3 Lösung

Eine offensichtliche Lösung hierfür wäre ein Tool wleches diesen prozess möglichst generisch und erweiterbar auomatisiert: CYBRAIL - Cyber Barrier for Reliable Academic Integrity and Log-analysis -.

2 Projektziele

3 Projektverlauf und Meilensteine

4 Ergebnisse und Leistungen

5 Qualitätssicherung

6 Risikoanalyse und Risikomanagement

7 Kommunikation

8 Ressourcenmanagement

9 Lessons Learned

10 Abschlussbewertung

11 Fazit und Ausblick

Abbildungsverzeichnis

Anhang