Protokoll Termin Polyas 16. Januar 2026

28. Januar 2026, 15:00 bis 17:15
POLYAS GmbH, Marie-Calm-Straße 1-5, 34131 Kassel

1 Anwesende

1.1 TU Dresden

- Cao Son Ta, Studierende
- Gregor Düster, Studierende
- Simon Bruder, Studierende
- Michael Theodor Wasiutinski, Studierende
- Moritz Schulz, Studierende
- Marco Lehner, Studierende
- Dr. Katharina Kern, Büro des Kanzlers
- Nick Dannenberg, TUD-CERT
- Mandy Dziubanek, Wahlbüro (online)
- Anja Löffler-Seifert, Justiziariat (online)

1.2 Polyas

- Ann-Kathrin Crede
- Jan Wegner, Geschäftsführer
- Janina Weineck, Election Managerin
- Wolfgang Jung, Softwareentwickeler

2 Ablauf

1. Vorstellungsrunde
2. Allgemeines zu Polyas von Jan Wegner
3. Vorstellen von Auszügen der BSI-Sicherheitsvorgaben
4. Beantwortung von Fragen der Studierenden

3 Allgemeines

Ziel seitens Polyas war die Vermittlung eines vollständigen Bildes gegenüber den Studierenden und die Beantwortung der vorab gestellten Fragen.

Polyas ist ein Unternehmen mit 33 Mitarbeitenden. Jährlich organisiert es etwa 2000 Onlinewahlen. Ziel des Unternehmens sei es, die demokratische Beteiligung durch Online-Wahlen zu fördern. Die Verwendung von Polyas im kleinen Rahmen hat an der TU Dresden bereits eine längere Geschichte. Im Jahr 2025 wurde für die Fernwahlen der Universitätswahlen erstmal Polyas anstelle der Briefwahl für alle Statusgruppen eingesetzt. Hierbei wurde Version CORE 2 eingesetzt, daher lag der Fokus des Termins auf dieser Version der Software. Polyas stellte auch Bezüge zur Folgeversion CORE 3 her.

Die Version 2 ist noch bis Mitte 2026 nach dem BSI-Profil BSI-CC-PP-0037-2008 zertifiziert. Polyas strebt eine Zertifizierung von CORE 3 gemäß dem BSI Schutzprofil BSI-CC-PP-0121 an.

Das Protokoll soll nicht primär die Meinung der Anwesenden widerspiegeln, sondern aufzeigen, welche Sachverhalte besprochen wurden.

4 Funktionsweise

Gemäß dem BSI Schutzprofil BSI-CC-PP-0037-2008 müssen Wahlverzeichnisserver, Validator und Wahlurnenserver physisch getrennt sein. Der Wahlverzeichnisserver und Validator sollen die Wahlberechtigungen prüfen, der Urnenserver soll die eigentlichen Stimmen annehmen.

Um den Zugang zur Wahlplattform zu kontrollieren, erfasst die Universität vorher die Wahlberechtigungen auf eigener Infrastruktur. Zu Beginn des Wahlzeitraums werden dann Token übermittelt, die auf Seiten von Polyas freigeschaltet sind und den Zugang zur Wahlplattform erlauben. Die Vermittlung findet über einen sog. SecureLink statt. Dieser Zugangslink wird nach IDM-Login im Self-Service Portal angezeigt. In dem SecureLink ist ein Zugangstoken enthalten.

Um die Wahlberechtigung zu prüfen, wird ein vorab eingetragener Token an den Wahlverzeichnisserver geschickt. Wird dieser als gültig akzeptiert, wird dieser an den Urnenserver geschickt, um die wahlberechtigte Person freizuschalten. Hierbei und bei anderen Kommunikationen zwischen den Komponenten kommt ein selbst implementiertes mTLS-ähnliches-Protokoll innerhalb der TLS-verschlüsselten Kommunikation zum Einsatz. Serverseitig werden BouncyCastle-Primitiven für die Implementierung verwendet.

Sollten die Server die Daten nicht wie vorgeschrieben löschen, wären aus den kombinierten Datensätzen Rückschlüsse auf Personen und deren Wahlverhalten möglich.

Der Token dient jeweils zur Stimmenabgabe für alle Stimmzettel. Wenn es sehr kleine Wählendengruppen gibt, ist es daher möglich, dass die Kombination von Stimmenabgaben auf die konkreten Wählenden schließen lässt. Wenn eine Wähler:in beispielsweise für Gremien wahlberechtigt ist, für welche es sehr wenige Wahlberechtigte gibt, ist eine Deanonymisierung möglich, auch wenn insgesamt über alle Wahlen hinweg sehr viele Stimmen abgegeben wurden. Ein besonderes Risiko ist, dass damit auch die Stimmen der Personen kompromittiert sind, die für die anderen Gremien abgegeben wurden. Gegenüber der Wahlleitung werden diese Daten im Export nicht offengelegt, eine Deanonymisierung könnte in solchen Fällen aber aus den Datenbankeinträgen heraus möglich sein. Polyas schlägt hierzu vor, mehrere Wahlen aufzusetzen, um diese Fälle abzufangen. In einer separaten Wahl würden betreffende Wählende dann einen anderen Token erhalten, der sich nicht zuordnen lässt.

Bei der Auszählung werden einzelne Dateien der Server mithilfe eines Hashverfahren auf Konsistenz geprüft. Dieser Ablauf und ein anderer Algorithmus in der Version CORE 3 wurden teilweise vereinfacht als „Blockchain“ bezeichnet. Die Vertretenden von Polyas haben klargestellt, dass es sich um keine Blockchain handelt.

5 Security Management

Der vom BSI kontrollierte Zertifizierungsprozess gestaltet sich sehr zeitaufwendig. Der Zeitaufwand kann aber ein zeitnahes Reagieren auf kritische Sicherheitslücken erschweren. Nach Zertifizierung werden Abhängigkeiten nur selektiv aktualisiert, wenn Polyas eine konkrete, das Produkt betreffende, Schwachstelle vorliegt. Die letzte Aktualisierung der Version CORE 2 betraf das Schließen einer Sicherheitslücke in der verwendeten Programmbibliothek Log4j. Die genannte Lücke wurde am 24. November 2021 entdeckt und wahrscheinlich bereits vorher als Zero-Day-Exploit genutzt. Die aktualisierte Version, die die Sicherheitslücke schließt, wurde am 11. März 2022 zertifiziert.

Möglicherweise auftretende Race Conditions werden dadurch behandelt, das ein zusätzliches „Cast“ ausgeführt wird. Hierfür wird die Stimmenauswahl nochmals bestätigt, sodass die letzte getätigte Auswahl bestätigt wird. So soll gewährleistet werden, dass nur eine Stimme abgegeben werden kann.

6 Deployment

Polyas bietet ihr Produkt nur als SaaS und nicht als On-Premise-Deployment an. Die Betreuung beim Aufsetzen sei unwirtschaftlich, da das Deployment durch das Vorhandensein mehrerer Komponenten zu komplex sei. Der Kunde habe keine Möglichkeit, nachzuvollziehen, ob tatsächlich die gewünschte Software installiert ist. Das Vertrauen in die Betriebsumgebung (Open Telekom Cloud) müsse ohnehin gegeben sein und Prozesse zur Nachvollziehbarkeit würden nicht zu nennenswert mehr Sicherheit beitragen.

Das Deployment finde als Docker-Container auf einem Kubernetes-Cluster der Open Telekom Cloud mittels Terraform statt. Die jeweiligen Umgebungen für eine Wahl würden nach Freigabe automatisiert über GitLab-CI-Pipelines deployt. Aus diesen Konfigurationen könne Polyas den Zustand des Systems zu Wahlbeginn wiederherstellen.

Wahlverzeichnisserver und Urnenserver würden über Anti-Affinity-Einstellung in Kubernetes auf verschiedenen physischen Servern deployt. Auf diese Dienste werde vom Browser der Wählenden nicht direkt zugegriffen, sondern über einen Ingress-Reverse-Proxy der Open Telekom Cloud, der auch die TLS-Verbindung zu den vorgenannten Servern terminiert. Polyas bestätigt, dass ein kompromittierter Ingress einen MitM-Angriff ermöglicht. Da der Login-Token per HTTP-POST-Request übermittelt wird und nur TLS-verschlüsselt ist, kann dieser im Ingress im Klartext abgefangen werden.

7 Fazit

Fragen zur Software bzw. zum Deployment wurden ausführlich und offen beantwortet. Der Termin war kein Code-Review und war auch nicht als ein solches geplant. Jedoch wurden durchaus konkrete Quellcode-Ausschnitte im Rahmen der Fragebeantwortung der gesamten Gruppe präsentiert.

Dieses Dokument wurde ohne Einsatz von Large Language Models erstellt.