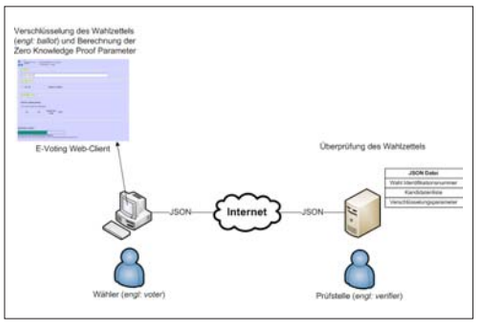
Paolo
Dorigo

Sonam
Samkang

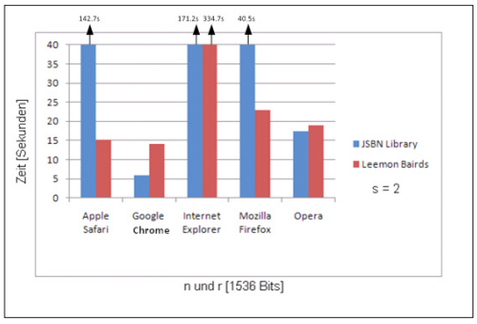
Diplomanden	Paolo Dorigo, Sonam Samkang
Examinator	Prof. Dr. Andreas Steffen
Experte	Dr. Ralf Hauser, PrivaSphere AG, Zürich ZH
Themengebiet	Internet-Technologien und -Anwendungen

E-Voting Web-Client mit JavaScript

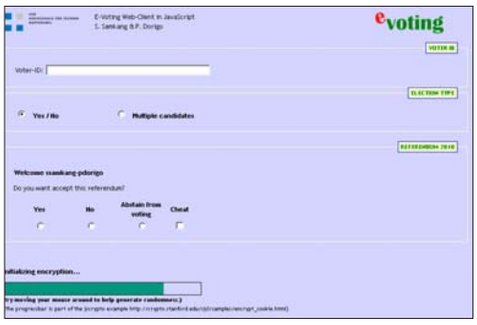
13 Ein sicheres, stabiles und transparentes Wahlsystem



Kommunikation zwischen Client und Server



Browser-Performance-Vergleich



Generierung von Zufallszahlen mittels Einsammlung von Mausbewegungssequenzen

Ausgangslage: Ein E-Voting-System muss speziellen Anforderungen genügen. Einerseits muss der Staat sicherstellen, dass jede stimmberechtigte Person nur eine einzige Stimme abgibt, andererseits muss sich der Stimmbürger darauf verlassen können, dass seine Wahl berücksichtigt und anonym ausgewertet wird. Dieses Projekt befasst sich mit der Aufgabe, ein solch sicheres, stabiles und transparentes Wahlsystem anzubieten. Das Ziel der Arbeit war, einen Web Client zu entwickeln, der mit Hilfe einer Big-Integer-Library und JavaScript (JS)-Funktionen das für E-Voting geeignete Damgård-Jurik-Kryptosystem (DJK) implementiert. Weiter galt es, die Verfügbarkeit von echten Zufallszahlen in JavaScript abzuklären. Die generierten Datensätze waren in der JS Object Notation (JSON) an den Server weiterzuleiten. In einem optionalen Teil der Arbeit sollte die Wohlgeformtheit der verschlüsselten Stimmzettel mittels eines Zero-Knowledge Proofs (ZKP) nachgewiesen werden können.

Vorgehen/Technologien: Die clientseitige Verschlüsselung der Wahlzettel mit JavaScript erlaubt es jedermann, den lesbaren JS-Code auf Korrektheit zu überprüfen. Der verwendete asymmetrische DJK-Verschlüsselungsalgorithmus hat die homomorphe Eigenschaft, dass die Multiplikation chiffrierter Werte der verschlüsselten Summe der entsprechenden Klartexte entspricht. Da nur das aufkumulierte Schlussresultat der Abstimmung entschlüsselt wird, ist dadurch die Anonymität der einzelnen Stimmen gewährleistet. Um die Gültigkeit eines verschlüsselten Wahlzettels zu überprüfen, wird ein ZKP-Protokoll verwendet. Da JS nur mit Zahlen kleiner als 53 Bit rechnen kann, ist für die Berechnungen eine JavaScript-Big-Integer-Library notwendig. Es wurden verschiedene Bibliotheken getestet und im Detail ausgewertet. Ebenfalls wurde die Möglichkeit abgeklärt, echte Zufallszahlen plattformunabhängig zu generieren.

Ergebnis: Die Wahl der Big-Integer-Library ist auf jsbn.js der Stanford University gefallen und für die Generierung von Zufallszahlen auf jscrypto.js, welche durch die Auswertung von Mausbewegungen Entropie gewinnt. Der implementierte JS-Client verschlüsselt erfolgreich Stimmzettel mittels des DJK und leitet das Chiffrat mit zusätzlichen, für den ZKP notwendigen Parametern im JSON Format an den Server weiter, der die empfangenen Daten auf ihre Wohlgeformtheit prüft. Bei der Verwendung eines 1024-Bit-RSA-Moduls benötigt der schnelle Google-Chrome-Browser nur 1,7 s für die DJK-Verschlüsselung und zusätzlich 8,8 s für die Generierung des ZKP.