«Passwort: Wahlen» – der technische Hintergrund und das Glossar zur Recherche

Welche Hacks bei der elektronischen Übermittlung von Wahlergebnissen möglich sind und wie die Kantone und Softwarefirmen die Schwachstellen beheben wollen.

Von Adrienne Fichter, 25.09.2020

Wenn in einem Kanton eine Wahl oder eine Abstimmung durchgeführt wird, kommt sogenannte Ergebnisermittlungssoftware zum Einsatz. Diese Software dient in den Wahllokalen dazu, die ausgezählte Stimmenzahl in ein zentrales System einzugeben, welches dann das Gesamtergebnis berechnet.

In unserer Recherche «Passwort: ‹Wahlen›» haben wir gemeinsam mit den beiden IT-Security-Forschern Melchior Limacher und Christian Killer untersucht, wie sicher diese Software in den verschiedenen Kantonen ist. In diesem Text listen wir auf, welche Schwachstellen wir gefunden haben, und führen die Stellungnahmen der Kantone und Firmen an.

Vorab: Limacher und Killer haben nicht alle 26 Kantonssysteme auf Basis der Staatsschreiber-Umfrage analysieren können. Sie suchten nach öffentlich verfügbaren Schnittstellen und Angaben und wurden bei insgesamt 15 Kantonen fündig.

Heisst das nun, dass die anderen 11 Kantone robustere IT-Systeme im Einsatz haben? Nein. Oder eher: Wir wissen es nicht.

Anders als beim E-Voting gibt es bei der Ergebnisermittlungssoftware keine Pflicht zur Offenlegung des Quellcodes. Die eingesetzten Wahlapplikationen bleiben eine Blackbox. Deshalb können keine vollständig gesicherten Aussagen gemacht werden.

In den folgenden Abschnitten finden Sie eine Übersicht, welcher Kanton welche Software einsetzt und wie sicher diese ist. Die Anbieter sind in alphabetischer Reihenfolge aufgeführt. Ein Glossar mit Erläuterungen zu den Fachbegriffen finden Sie am Ende des Textes.

Die verschiedenen Systeme und Stellungnahmen der Anbieter:

Abraxas

Die Firma Abraxas bietet ein Ergebnisermittlungssystem mit dem Namen «Wabsti» an. Die Lösung wird in **Zürich, St. Gallen** und im **Thurgau** verwendet. Für die Beurteilung der IT-Sicherheit von «Wabsti» lagen gemäss Limacher und Killer zu wenig Informationen vor. Auf der Website des Unternehmens steht lediglich, dass «sämtliche Daten auf unserem zentralen Abraxas-Rechnersystem» gespeichert sind. Wahldaten bei einer Firma zu speichern, finden Limacher und Killer per se heikel, da es sich um kritische Infrastruktur handelt, die in der Hoheit der Kantone liegen muss. Die Firma Abraxas bringt eine neue Produktelinie «Voting Ausmittlung» als Nachfolgeprodukt für «Wabsti» auf den Markt und beteiligt sich nach eigenen Angaben bei der Ausschreibung des Kantons St. Gallen.

Bedag, SyGev

Limacher und Killer haben sich auch die Systeme von **Fribourg** und **Neuenburg** (beide SyGev) angeschaut und dabei keine nennenswerten Mängel gefunden. «Allerdings lagen keine detaillierten Informationen zu den Systemen vor, sondern es konnte quasi nur die ‹Fassade› im Internet begutachtet werden», sagt IT-Security-Forscher Melchior Limacher. Bei der Software des Kantons **Bern** fanden die Forscher dieselbe Schwachstelle wie beim Kanton Wallis (siehe weiter unten): Das System wechselt von einer verschlüsselten auf eine unverschlüsselte Verbindung, dies würde einer Angreiferin erlauben, sich in den Datenverkehr einzuklinken. Der Anbieter für den Kanton Bern – die Firma Bedag – hat diese Lücke im Verlauf des Jahres von sich aus behoben.

Sesam

Die Firma Sesam bietet eine Software namens «Sesam Wahlen» an. Zu ihren Kunden zählen **Baselland, Basel-Stadt, Graubünden, Uri, Glarus, Luzern, Nidwalden, Obwalden** und auch der Kanton **Schaffhausen.** Limacher und Killer fanden heraus, dass beim Einsatz relativ einfach sogenannte Insiderattacken möglich sind – also Angriffe durch jemanden, der sich innerhalb des Verwaltungsnetzes befindet oder sich hineinhackt. Denn trotz vielfältiger Rechteverwaltung innerhalb der Anwendung «Sesam Wahlen» wird gerade mal ein einziges Datenbankbenutzerkonto verwendet. Mit diesem Konto erhält der Vertreter der Wahlbehörde umfassende Administrationsbefugnisse. Wer im Besitz dieses Passworts ist, kann sich ungehindert Zugang zur ganzen Datenbank verschaffen und nach Belieben Wahlergebnisse verändern. Dies macht «Sesam Wahlen» anfällig für Manipulationen durch Insider.

Zur Erinnerung: Ein ähnliches Manipulationsszenario fand die Forschergruppe rund um die Hackerin Sarah Jamie Lewis 2019 bei der E-Voting-Software der Post – was letzten Endes zum Abschuss des Projekts führte.

Auch hat Sesam die Installationsanleitung für ihre Software im Netz veröffentlicht – mit dem simpelsten Passwort, das man sich vorstellen kann: «Wahlen». Eine explizite Aufforderung zur Änderung des Passworts fehlt in der Dokumentation.

Sesam-CEO Reinhard Semlitsch bestätigt die Befunde. Er sagt: «Die aufgeführte User-Passwort-Kombination ist initial und sollte von den Kunden angepasst werden.» Seiner Ansicht nach müssen Kantone Insiderattacken

mit eigenen Präventivmassnahmen vorbeugen. «Dass jemand mit direktem Zugriff auf die Datenbank dort Daten verändern kann, ist unbestritten. Wir und unsere Kunden sind uns dessen bewusst. Aus unserer Sicht haben die Kantone ausreichend gute Sicherheitskonzepte, um unbefugten und direkten Zugriff auf die Datenbank praktisch auszuschliessen.»

In Schaffhausen, ebenfalls eine Sesam-Kundin, finden dieses Wochenende – am 27. September – Wahlen statt. Solche Insiderattacken sind glücklicherweise hier praktisch ausgeschlossen, da die für die Ergebnisermittlung verwendeten Rechner der Kantonsverwaltung nicht ans Netz angeschlossen sind und die Berechtigungskonzepte für den Datenbankbenutzer sehr strikt sind, wie Christian Ritzmann, stellvertretender Staatsschreiber des Kantons Schaffhausen, auf Anfrage der Republik ausführt.

Sitrox

Das Ergebnisermittlungssystem der Firma Sitrox heisst «VeWork». Die Kantone **Aargau** und **Zug** arbeiten damit. **Solothurn** ist eine neue Kundin von Sitrox und wird am 27. September mit «VeWork» Abstimmungsergebnisse ermitteln. Auch hier haben die Forscher gemeinsam mit der Republik einige Mängel ausgemacht: Sie fanden bis März 2020 schwache Verschlüsselungsalgorithmen wie RC4, das veraltete Protokoll SSLv3 sowie das Fehlen von HSTS und von etablierten Standardmechanismen zum Schutz vor «Session-Hijacking».

Limacher und Killer monieren, dass diese Versäumnisse alles andere als Best Practice seien. Ausserdem operiert Sitrox mit einem veralteten Applikationsserver der niederländischen Firma Phusion (Versionsnummer 5.0.24). Phusion fordert ihre Kundinnen zu einem Update auf, weil sie ihr altes Produkt nur noch bei spezieller Vereinbarung unterstützt.

Die Firma Sitrox war zwar einer der wenigen Anbieter, die auf die Kontaktaufnahme der IT-Security-Forscher Limacher und Killer reagierten, und nahm seither einige Updates vor. Sie stuft die Befunde jedoch als irrelevant ein und liefert für jede Entscheidung detaillierte Erklärungen. Man kompensiere die beschriebenen Angriffsszenarien mit anderen Sicherheitsmechanismen, etwa dem Blockieren von externen IP-Zugriffen, sagt Christian Singer, Chief Technology Officer der Firma. «Die fehlenden CSP, Secure-Flags der Session-Cookies und HSTS setzt Sitrox bei VeWorks nicht ein, weil die anderen Techniken (IP-Pinning, IP-Shielding) nach unserer fachlichen Beurteilung wirksamer sind.»

Nur: Bei CSP, Secure-Flags und HSTS handelt es sich um anerkannte Sicherheitsmassnahmen, die nicht mit einem IP-Whitelisting ersetzt werden können. «Unabhängig davon ist in der IT-Sicherheit eine sogenannte Verteidigung in der Tiefe eine gute Praxis. Verteidigung in der Tiefe bedeutet, dass man sich nicht auf eine einzelne Verteidigungslinie verlässt, sondern mehrere solcher Linien einsetzt. Dies gilt insbesondere für günstige und einfache Massnahmen wie HSTS und Secure-Flags», sagt der IT-Security-Forscher Melchior Limacher.

Ein IP-Whitelisting würde beispielsweise keinen Schutz bieten gegen Angriffe aus internen Netzwerken. Die Sicherheitseigenschaften eines IP-Whitelistings hängen ausserdem massgeblich davon ab, wie ein solches implementiert wird. Ein IP-Whitelisting auf Netzwerkebene nützt ungleich mehr als ein IP-Whitelisting auf Applikationsebene. Beste Industriepraxis wäre es ausserdem, administrative Schnittstellen grundsätzlich gar nicht erst im Internet zu exponieren – egal, ob mit oder ohne IP-Whitelisting.

Dass Sitrox nicht auf die neue Version des niederländischen Phusion Passenger umsteigen möchte, begründet die Firma damit, dass die von ihr benutzten Teile stabil und in ihrem Kontext sicher seien. Würde man unnötige Upgrades von einzelnen Komponenten durchführen, so könnte die installierte Software wohl nicht mehr funktionieren. «Wenn solche Dinge angepasst werden, kann es zu Inkompatibilitäten mit der Kundeninfrastruktur kommen», so Singer. «Das müssen wir berücksichtigen.»

Sitrox hat im Verlauf dieses Jahres den unsicheren RC4-Algorithmus entfernt. Auch wurde das verwundbare Protokoll SSLv3 ersetzt. Beide Massnahmen seien unabhängig von den beiden IT-Security-Forschern ergriffen worden, betont der CTO von Sitrox.

Votel, Wallis

Der Kanton **Wallis** setzt ebenfalls eine angriffsanfällige Software ein: «Votel», die von der Firma Deeprod SA stammt. Die Hauptkritik von Limacher und Killer bei «Votel» lautet: Bei der Kommunikation zwischen Browser und Server wird von einer verschlüsselten zu einer unverschlüsselten Verbindung gewechselt. Für kriminelle Hacker ist es ein Leichtes, hier sogenannte Man-in-the-Middle-Attacken durchzuführen, indem sie sich in den Netzverkehr einschleusen und die Session-Cookies oder Passwörter auslesen.

Das Wallis hat auf den Hinweis der Republik reagiert. Patrick Siggen, Chief Information Security Officer des Kantons, hat die Schwachstelle eingeräumt. Man habe diese beim letzten Audit vom Oktober 2019 selbst bemerkt und werde bis zu den Abstimmungen im November Korrekturen vornehmen.

Doch weshalb wurde über ein Jahr lang nichts getan?

Man habe diese Schwachstelle angesichts «vieler weiterer Sicherheitsmassnahmen und Kontrollprozesse nicht als kritisch für den reibungslosen Ablauf der Abstimmungen im September angesehen», sagt Siggen. «Votel» werde bei den Gemeinde- und Generalratswahlen von Oktober und November 2020 nicht eingesetzt.

Votel, Tessin

Die Eigenentwicklung des Kantons **Tessin** enthält ähnliche Schwächen wie «VeWork» (siehe unter Sitrox): So wird bei der Web-Applikation «Votel» (sie heisst gleich wie jene vom Wallis, ob es sich auch um dieselbe Herstellerfirma handelt, wissen wir nicht) ebenfalls mit einem nicht mehr zeitgemässen Verschlüsselungsalgorithmus (3DES) und fehlerhaft konfigurierten HSTS-Einstellungen operiert. Beim Tessiner «Votel» scheint es sich um ein System zu handeln, das womöglich bereits vor Jahrzehnten eingeführt und seither vermutlich sukzessive modernisiert worden ist. Die gefundenen PHP3-Dateien deuten jedenfalls auf eine Eigenentwicklung aus den Neunzigerjahren hin (mehr Informationen zu PHP weiter unten).

Die Tessiner Staatskanzlei nahm zu den Vorwürfen Stellung: «Das fragliche Computersystem, das intern entwickelt wurde, gilt als sicher und wird ständig aktualisiert. Einige der von Ihnen erwähnten Protokolle sind inzwischen, wie bereits vor einiger Zeit geplant, ausser Betrieb genommen worden.»

Doch ein Test der Republik von dieser Woche ergab: Alle gefundenen Schwachstellen (HSTS, 3DES) sind immer noch aktiv. Die Frage nach dem Hersteller der Software sowie nach dem Einsatz von PHP3 liess die Tessiner Staatskanzlei unbeantwortet.

Nun zum kleinen Glossar mit den wichtigsten Fachbegriffen:

Man-in-the-Middle-Attacken

Bei einem Man-in-the-Middle-Angriff (MITM-Angriff) schaltet sich eine Angreiferin in die Kommunikation zwischen zwei Rechnern ein (also beispielsweise zwischen Browser und Webserver). Damit erlangt sie die vollständige Kontrolle über den Datenaustausch und kann diesen überall mitverfolgen und manipulieren. Den beiden Rechnern bleibt dieser Angriff verborgen, da beide davon ausgehen, jeweils direkt mit der Gegenseite zu kommunizieren. Praktisch lassen sich solche Attacken zum Beispiel über manipulierte WLAN-Hotspots ausführen (nicht jedes WLAN, welches «Starbucks» heisst, gehört auch wirklich zur Kaffeehauskette) oder indem man als Angreifer kurzzeitig den Domain Name Service austrickst und einen Zugriff auf eine Website wie etwa Republik.ch über den eigenen Rechner leitet.

Um solchen Attacken vorzubeugen, werden Daten zwischen Browser und Webserver heute meist mittels eines Public/Private-Key-Verfahrens verschlüsselt. Die Echtheit der Schlüssel wird von einer unabhängigen und vertrauenswürdigen Zertifizierungsstelle bestätigt. Dies bedingt aber, dass Webserver korrekt konfiguriert sind und Zugriffe über eine unverschlüsselte Verbindung ablehnen. Auch die im E-Banking eingesetzte Bestätigung von Zahlungsaufträgen via SMS oder Zusatzgeräte, welche auf nur der Bank und der Benutzerin bekannten Daten basiert, ist ein Mittel, um MITM-Attacken zu verhindern.

Insiderattacken

Insiderattacken sind ein breiter Begriff für sämtliche Arten von Angriffen, bei denen es sich ein Hacker zunutze macht, dass vorhandene Schutzmechanismen nur gegen die Aussenwelt greifen und internen Benutzerinnen/Systemen keine Hindernisse in den Weg legen. So lässt sich der Zugriff auf kantonale Web- oder Datenbankserver zum Beispiel auf IP-Adressen des kantonalen Netzes einschränken. Dieser Schutz greift dann nicht, wenn etwa ein Hacker es schafft, einen der im kantonalen Netz vorhandenen Rechner zu übernehmen, oder wenn eine Mitarbeiterin des Kantons selbst die Angreiferin ist. Um eine klassische Insiderattacke zu veranschaulichen, ein konkretes Beispiel: die Alterskontrolle beim Eingang eines Nachtclubs. Der Club kontrolliert die Volljährigkeit der Gäste jeweils nur beim Eingang, jedoch nicht an der Bar, bei der Bestellung eines Drinks. Sollten sich Minderjährige durch den Lieferanteneingang oder das WC-Fenster einschleichen, gelangen sie problemlos an harte Drinks. Die Kontrollmechanismen des Nachtclubs sind in diesem Fall nicht strikt genug.

RC4-Cipher-Algorithmus

RC4 ist ein 1987 entwickelter Verschlüsselungsalgorithmus, der auf der fortlaufenden Verschlüsselung der Originaldaten mit einer algorithmisch

erzeugten Zufallsfolge basiert. Er ist sehr einfach und effizient mit Hard- oder Software implementierbar und fand daher schnell eine weite Verbreitung.

Seit 2001 sind erfolgreiche Angriffe gegen RC4 bekannt, also Wege, wie sich RC4-verschlüsselte Daten unter gewissen Umständen entschlüsseln lassen. Die ersten Angriffsmethoden konnten durch Anpassung der Zufallsfolgenermittlung abgewehrt werden, aber mit seit 2013 bekannten Methoden lässt sich RC4 auf modernen Rechnern innert nützlicher Frist knacken. Es hält sich ausserdem hartnäckig das Gerücht, dass die NSA in der Lage sei, RC4 in Echtzeit zu brechen. Aus diesen Gründen ist der Einsatz von RC4 für HTTPS seit 2015 nicht mehr zulässig, er wird aber von einzelnen Webservern zur Wahrung der Kompatibilität mit älteren Browsern weiterhin unterstützt.

TLS/SSL

TLS (Transport Layer Security) ist ein Verschlüsselungsprotokoll zur sicheren Datenübertragung im Internet, es kommt insbesondere bei sämtlichen Zugriffen via HTTPS zum Einsatz. Es besteht aus zwei Teilen, dem Handshake und der eigentlichen Datenübertragung. Während des Handshakes einigen sich Browser und Server auf einen von beiden verstandenen Verschlüsselungsalgorithmus und tauschen die für die Kommunikation notwendigen Schlüsselinformationen aus. Mit diesen Informationen wird anschliessend die Datenübertragung gesichert.

TLS (bzw. sein Vorgänger SSL) ist seit 1994 im Einsatz und wurde wiederholt angepasst, um die Sicherheit zu erhöhen und aufgetretene Schwachstellen zu beheben. So sollen zum Beispiel alte Versionen, welche den als unsicher bekannten Algorithmus RC4 (siehe oben) erlauben, seit Mai 2015 nicht mehr eingesetzt werden. Aktuell ist TLS 1.3 im Einsatz, Unterstützung für ältere Versionen wurde durch die grossen Browseranbieter abgekündigt. Die Version SSLv3 ist stark veraltet und besitzt schwerwiegende bekannte Schwachstellen.

HSTS

HTTP Strict Transport Security (HSTS) schützt verschlüsselte HTTPS-Verbindungen gegen gezieltes oder versehentliches Downgraden auf HTTP (und damit auch gegen Session-Hijacking, siehe unten). Das Verfahren braucht aktuelle Software sowohl auf Browser- wie auch auf Webserver-Seite.

Bei der Verwendung von HSTS fordert der Webserver den Browser auf, Zugriffe auf die Domain des Webservers auch dann über HTTPS zu leiten, wenn die URL selbst nur HTTP vorgibt. Dies stellt sicher, dass ein alter, noch mit HTTP markierter Link nicht dazu führt, dass Daten wie das Session-Cookie oder andere kritische Daten unverschlüsselt übertragen werden. Falls der Aufbau einer HTTPS-Verbindung, zum Beispiel aufgrund eines ungültigen Server-Zertifikats, nicht möglich ist, wird die Verbindung gar nicht erst aufgebaut.

Session-Hijacking

Webserver haben prinzipiell keine Möglichkeit zu erkennen, ob verschiedene Seitenzugriffe von derselben Benutzerin oder von unterschiedlichen Benutzern stammen. Daher schicken sie bei einem Log-in jeweils eine Kennung in Form eines Session-Cookies an den Browser. Dieses Cookie muss

der Browser dann bei jedem weiteren Seitenzugriff jeweils mitschicken. Wenn es eine Angreiferin nun schafft, das Session-Cookie aus der Kommunikation zwischen Browser und Webserver herauszukopieren (zum Beispiel durch das Mitlesen des gesamten Netzwerkverkehrs im Falle von unverschlüsselten Verbindungen, durch das Entschlüsseln von schlecht gesicherten Verbindungen oder über eine vom Benutzer gutgläubig installierte Browser-Erweiterung), dann kann sie sich gegenüber dem Browser als eingeloggte Benutzerin ausgeben und die Webapplikation mit denselben Rechten wie diese nutzen.

Um dies zu verhindern, kann der Server beim Definieren des Session-Cookies Zusatzinformationen an den Browser mitgeben, um diesen anzuhalten, das Session-Cookie nur über verschlüsselte Verbindungen an den Server zu schicken, oder um den Zugriff auf den Cookie-Wert aus Browser-Extensions beziehungsweise aus JavaScript generell zu verbieten.

PHP

PHP ist eine Skriptsprache, die hauptsächlich zur Entwicklung von dynamischen Websites verwendet wird. Eine erste Version wurde 1995 publiziert, sie wurde seither ständig erweitert und ausgebaut, aktuell ist PHP7. Die beim Kanton Tessin erwähnte Version PHP3 wurde bereits 2000 durch PHP4 abgelöst und ist aus heutiger Sicht veraltet und fehleranfällig. Die Vorteile von PHP liegen insbesondere darin, dass damit einfache Websites auch ohne grosse Programmierkenntnisse und unter Verwendung von Open-Source-Bibliotheken entwickelt werden können. Für grössere und sicherheitskritische Projekte ist PHP eher ungeeignet, da die Sprache gegen typische Programmierfehler relativ tolerant ist und diese dann allenfalls im realen Einsatz für Angriffe ausgenutzt werden können.