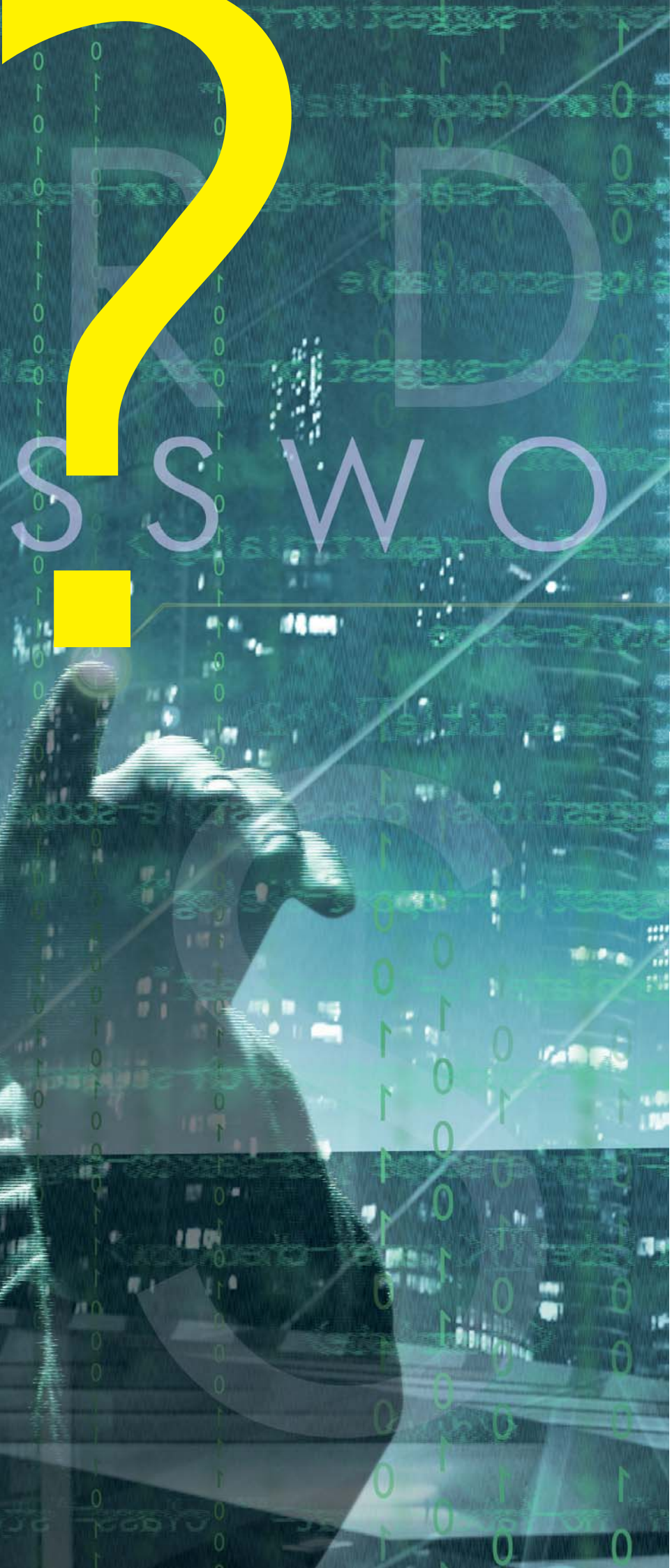
Wie sicher sind meine (Patienten-) Daten????

Das Thema Datensicherheit und Datenschutz in IT-Systemen ist komplex und für Laien abschreckend. Andererseits ist es für uns als Heilberufler (immer m/w/d) auch keine Lösung, das Thema zu ignorieren, allein schon deshalb, weil es zahlreiche rechtliche Anforderungen gibt, die wir in dieser Hinsicht beachten müssen.



Wer sich in der Berufsausübung nicht auf Papier, Stift und Schreibmaschine beschränken will, muss sich in jedem Fall mit dem Thema Datenschutz und Datensicherheit beschäftigen. Für Rechnungen z. B. ohne weitere Vorkehrungen einfach nur eine Textverarbeitung zu verwenden, ist allein schon steuerrechtlich gesehen eine ganz schlechte Lösung.

Eine steuerrechtskonforme Software zur Erstellung von Rechnungen sollte es mindestens sein. Aber dann kann ich auch ernsthaft überlegen, ob nicht eine Praxissoftware, die ja den Kalender direkt mit der Rechnungserstellung verbindet (weniger Übertragungsfehler), die komfortablere Lösung ist.

Welche Praxissoftware ist die Richtige?

Wie immer gibt es nicht die eine Lösung für jeden und alle, jede Lösung hat Vor- und Nachteile, die jeder für sich gewichten muss. Aber je besser ich die dahinter liegenden Fragen verstanden habe, um so fundierter kann ich meine Entscheidung treffen.

Es ist hilfreich, sich den eigenen Bedarf und die Kriterien - auch die langfristigen - klarzumachen, denn ein späterer Wechsel ist meist aufwändig.

Vorweg und wenig überraschend: eine 100 %ige Sicherheit gibt es bei IT-Systemen nie und nirgends. Aber natürlich unterschiedliche Meinungen darüber, was die richtige Antwort darauf ist. Je nach Blickwinkel und Eigeninteresse fallen die Antworten verschieden aus. Da bin ich nicht ausgenommen.

Das hat natürlich auch damit zu tun, dass jede Software-Entscheidung schwierige Abwägungen beinhaltet: Preis, Plattform, Funktionen, Bedienungsfreundlichkeit, Sicherheitskonzept, Pflegeaufwand, Zukunftssicherheit und mehr. Da muss einiges unter einen Hut gebracht werden. Kompromisse sind unvermeidlich.

Als Praxisbetreiber haben wir es mit unterschiedlichen Typen von Gefährdung unserer Daten zu tun. Hauptsächlich:

- Datenverlust durch einen technischen Defekt
- Datenträgerzerstörung bei mir selbst oder bei meinem Online Software-Anbieter

- „Diebstahl" von Daten durch einen realen oder elektronischen Einbruch
- Verwandlung der Daten in unleserlichen Datenmüll durch einen Angriff mit einem Verschlüsselungsprogramm („Ransomware")
- einen Brand und einiges mehr.

Das sind tatsächliche Gefährdungen, wie die folgenden Beispiele zeigen

Am 10. März 2021 zerstörte ein Großbrand zwei von vier Blöcken eines Rechenzentrums des Providers OVH in Straßburg und mit ihnen zahllose Datenträger, auf denen die Daten von Millionen Kunden oder die Daten von Kunden dieser Kunden gespeichert waren. Die Daten etlicher dieser Kunden sind aufgrund unzureichender Sicherungsstrategien für immer vollständig verloren (keine automatische Georedundanz).

2020 wurden in Finnland tausende von psychotherapeutischen Unterlagen und 2018 in Norwegen sogar Millionen von Gesundheitsdaten von Servern gestohlen, auf denen diese Daten zentral gespeichert waren.

Im November 2021 fiel ein zentraler Standort der Amazon Web Services (AWS) in den USA für Stunden weitgehend aus. Zahlreiche Dienste waren daraufhin in den USA nicht mehr erreichbar, darunter auch Netflix.

Ende 2021 wurde eine Sicherheitslücke in einer Software bekannt, die auf Hunderten von Millionen Rechnern die technischen Ereignisse protokolliert. Unglückseligerweise war das Ausnutzen dieser Lücke so einfach, dass es auch unerfahrenen Hackern leicht fiel. Fachleute sprechen von einem Daten-GAU.

Ende letzten Jahres fiel das Unternehmen hinter der Arztsoftware Medatixx einem Verschlüsselungsangriff zum Opfer.

In diesem Jahr traf es die Arztsoftware inSuite, in der die zivilgesellschaftliche Gruppe „zerforschung" angeblich „gravierende Probleme" entdeckt haben will.

Ergo: Es kann mit einer gewissen, eher niedrig anzusetzenden Wahrscheinlichkeit jeden jederzeit auf irgendeine Weise treffen. Und das sind nur die Beispiele von Großereignissen. Die vielen, kleinen privaten Desaster bekommen wir ja gar nicht mit, wenn wir nicht nah dran sind. Egal, wird schon nichts passieren ...?

Ein paar technische Grundlagen zu Computerprogrammen

Es gibt zwei grundlegend unterschiedliche Lösungswege für Software.

1. Installation einer Software auf einem Gerät direkt unter der Kontrolle des Anwenders, einem Einzelplatzrechner, einem Praxisserver oder auch Mobile Device. Ein Einzelplatzrechner ist die wohl jedem vertrauteste Form. Eine in Praxen mit mehreren Mitarbeitern sinnvollere Lösung als die Einzelplatzlösung sieht etwas anders aus.

Das Praxisprogramm läuft auf nur einem Rechner, dem Praxis-Zentralrechner (Praxisserver), der über das lokale Netzwerk per Kabelverbindung oder über WLAN mit allen Arbeitsplatz-Rechnern („Clients") verbunden ist. Die Arbeitsplatz-Rechner sind dann im wesentlichen nur noch Anzeigegeräte für die Nutzeroberfläche. Früher wurde von einem Datenträger installiert, heute wird ein Programm normalerweise von einem Server irgendwo im Internet heruntergeladen.

2. Softwarelösungen in der „Cloud" (auch als „Webanwendung" oder „Web-App" bezeichnet).

Seit inzwischen über 10 Jahren haben sich erfolgreich Softwarelösungen etabliert, bei denen bildlich gesprochen der Praxisserver von einem professionellen Anbieter in einem Rechenzentrum als Dienstleistung betrieben wird (als Auftragsverarbeiter im Sinne der DSGVO). Ich muss dann auf meinem Gerät nur einen modernen Browser wie Chrome, Safari oder Firefox (oder eine Anzeige-App) installieren und mit dem Internet verbunden sein.

Nach einer Registrierung bei einem Anbieter nutze ich dann die größtenteils auf dessen Server laufende Anwendung. Der überwiegende Teil der Datenverarbeitung und Datenspeicherung findet dort statt. Das Ganze nennt sich auch „Software-

as-a-Service" (SaaS). Die Anwender sehen auf ihren Geräten nur das (meist grafische) „User Interface" der Software. Das wird auch „Server-Client-Architektur" genannt.

Solche „serverseitigen bzw. serverbasierten" Anwendungen sind gemeint, wenn es heißt, ein Programm laufe in der „Cloud". Was natürlich nur eine Metapher ist, die Programme laufen immer noch auf Rechnern, nur sehe ich diese nicht mehr vor mir. Mittlerweile werden sogar vollständige „virtualisierte" Rechner als „Cloud-Lösung" angeboten, nicht nur einzelne Programme.

Die Vorteile liegen auf der Hand: die App ist mit funktionierender Netzverbindung immer und von überall erreichbar, läuft automatisch immer mit der aktuellen Version, und auf die Daten passen die Profis vom Anbieter auf. Der Anbieter sorgt für Wartung und die regelmäßige Sicherung von System und Daten.

Natürlich ist es keineswegs trivial, die technische Infrastruktur, die für eine Cloudlösung erforderlich ist, ununterbrochen erreichbar und sicher zu betreiben. Überwiegend klappt das gut, aber natürlich kann es auch mal zu Ausfällen von für Kunden wichtigen Systemen kommen.

Auch die verschiedenen Browser sind im zurückliegenden Jahrzehnt immer „mächtiger" geworden und haben mit jedem Update mehr Funktionen hinzubekommen.

Für dynamische Elemente in den Browsern dient die Scriptsprache JavaScript. Aber auch damit kann durchaus Schlimmeres angestellt werden, weshalb manche Menschen JavaScript (oder auch Cookies) grundsätzlich erst einmal abschalten.

JavaScript ermöglicht die Verlagerung von Teilen der Programmlogik vom Server zum Client. Einfachere Operationen werden im Browser direkt ausgeführt, was den Server entlastet und die Anwendung beschleunigt, weil die Antwort auf Nutzeraktionen beschleunigt wird.

Und so gibt es seit einiger Zeit auch Webanwendungen, die fast vollständig im Nutzergerät (Client) arbeiten und bei denen der Webserver nur die Programmdateien zum Übermitteln in den Browser bereithält.

Man könnte sagen, dass diese Form einer Web App die Eigenschaften von installierten Programmen und Online Programmen miteinander kreuzt: Das Anwendungsprogramm und meine Anwendungsdaten liegen auf einem entfernten Server. Bin ich „authentifiziert", lädt mein Browser die Anwendung wie eine Webseite zusammen mit meinen Daten herunter und läuft wie eine installierte App komplett in meinem Gerät. Diese Technik wird als "clientseitige oder clientbasierte Programmausführung" bezeichnet (im Gegensatz zur oben beschriebenen serverseitigen). So weit zu den Grundlagen.

Zum richtigen Verständnis zur Beurteilung der Frage, wann Daten „sicher" sind, gehört seit Jahren ein weiteres Thema: die Verschlüsselung.

Datenverschlüsselung

Bei einer normalen serverseitigen Anwendung werden die Daten zum einen bei der Übertragung zwischen Browser und Webserver des Diensteanbieters („Data-in-Transit") automatisch verschlüsselt, wenn dieses Feature vom Anbieter für seinen Webserver aktiviert ist. Das wird auch als „Transportverschlüsselung" bezeichnet. Bildlich gesehen wird ein Tunnel zwischen Browser und Webserver eingerichtet, der gegen Spähversuche von außen geschützt ist (zu erkennen auch am „https://" in der URL-Adresszeile). Es handelt sich im allgemeinen Verständnis um eine technisch implementierte „Punkt-zu-Punkt-Verschlüsselung".

Am Beispiel einer Mail (etwas verkürzt): Ich öffne die Nutzeroberfläche eines Mailanbieters im Browser und schreibe eine Mail an einen Freund. Zu Beginn der Sitzung einigen sich mein Browser und Webserver meines Mailanbieters auf einen Schlüssel für meine Datenübertragung (die erste „Punkt-zu-Punkt-Verschlüsselung"). Klicke ich auf „Senden" schickt mein Mailanbieter die Nachricht erneut verschlüsselt zum Mailanbieter meines Freundes (die zweite „Punkt-zu-Punkt-Verschlüsselung").

Ruft mein Freund seine Mails mit seinem Browser ab, läuft dieser Vorgang ein drit-

tes Mal ab. Im Browser und im Webserver werden die Daten bei jedem einzelnen Schritt sofort entschlüsselt und weiter verarbeitet. Auf dem Weg vom „Sender" (also mir) zum „Empfänger" (meinem Freund) ist meine Nachricht nicht durchgängig verschlüsselt. Moderne Browser erwarten eine Transportverschlüsselung und warnen einen Nutzer im Falle unverschlüsselter Datenübertragung.

Nach der (Zwischen-)Speicherung oder Weiterleitung wird meine Mail wiederum verschlüsselt und auf einem Storageserver abgelegt („Data-at-Rest"). Der Mailanbieter verfügt notwendigerweise über alle für die Ver- und Entschlüsselung notwendigen Schlüssel. Der Anwender bekommt von der ganzen Sache im Grunde gar nichts mit. Sein Browser verwaltet die Schlüssel und zugrundeliegenden Sicherheitszertifikate. Bei komplexeren Web Apps als nur einer Mailsoftware ist der Verlauf im Prinzip der Gleiche, nur dass aufwändigere Verarbeitungsschritte stattfinden als nur eine Weiterversendung und Speicherung. Handelt es sich um einen Dienst nur für mich (z. B. Cloudspeicher) bin ich gleichzeitig Sender und Empfänger.

Bei einer clientseitigen Applikation (zur Erinnerung: Programm arbeitet vollständig im Browser) laufen Datenübertragung und zentrale Speicherung beim Diensteanbieter ohne weitere Maßnahmen überhaupt nicht anders. Weil hier aber die Verarbeitung aller Daten ausschließlich im Browser stattfindet, also der Diensteanbieter überhaupt keinen Zugriff auf die Daten in lesbarer, d. h. verarbeitbarer Form braucht, kann vor der Browser/Webserverbasierten Transportverschlüsselung eine zusätzliche anwendungsgetriebene „Inhaltsverschlüsselung" der Daten erfolgen, bevor sie übertragen werden.

Diese Inhaltsverschlüsselung wird vom empfangenden Webserver nicht angerührt, sondern die Daten werden inhaltsverschlüsselt in der zentralen Datenbank gespeichert. Schlüsselzugriff hat nur der Anwender, nicht der Anbieter. Daten liegen somit lesbar nur in den Nutzergeräten vor. Diese Technik wird häufig auch unter der Bezeichnung „Ende-zu-Ende-Verschlüsselung" vermarktet und ist Standard bei Messenger Apps. Einzelne Mail- und Cloudspeicheranbietern bieten es auch.

Ende-zu-Ende-verschlüsselt sind Inhalte (Nachrichten, Daten) auf ihrem Transportweg, wenn sie auf dem gesamten Strecke zwischen Sender und Empfänger verschlüsselt bleiben. Soll diese Ende-zu-Ende-Verschlüsselung auch die Server meines Anbieters einbeziehen, muss ich sicherstellen, dass dieser auf meine Schlüssel keinen Lesezugriff hat.

Nur dann sind die Daten ausschließlich in den von mir autorisierten Anwender-Endgeräten lesbar. Ich darf dann keine Webanwendungen verwenden, die eine serverseitige Verarbeitung einsetzen.

Wenn ein Programm nun schon vollständig im Browser abläuft, kann man noch einen Schritt weiter gehen und die Daten in der Browserdatendank speichern. Die App „spricht" dann nur noch mit der lokalen Browserdatenbank, die nun wiederum im Hintergrund diese Daten mit der zentralen Datenbak inhaltsverschlüsselt synchronisiert, solange eine Netzverbindung besteht. Diesen Ansatz nennt man „local-first". Müssen bei der Anwendungsinitialisierung große Datenmengen übertragen werden, verzögert das den eigentlichen Programmstart gegenüber einer Anwendung ohne lokale Speicherung.

„Local-first" wiederum lässt sich aufbohren zu einem „offline-first", also der Fähigkeit, mit einem browserbasierten Programm auch dann weiter arbeiten zu können, wenn die Netzverbindung einmal ausfällt oder gestört ist (Offlinefähigkeit). Inhaltsverschlüsselung und „local/offline-first" bestehen nebeneinander, beide basieren auf der clientseitigen Programmausführung, sind aber nicht voneinander abhängig.

Auch für die im Browser gespeicherten Daten besteht die Möglichkeit, sie verschlüsselt zu speichern, sodass auch im Nutzergerät niemand sie anschauen kann, wenn er nicht in der App eingeloggt ist. Wichtig ist, auch den Schlüssel so zu verstecken, dass nur ein authentifizierter Nutzer darauf Zugriff hat.

Serverseitige Online-Web Apps können auf die Architektur der „Local- bzw. offline-first Web App" nicht umgestellt werden und umgekehrt, ohne sie komplett neu zu programmieren. Der Programmcode für Server funktioniert im Browser nicht.

Zu den Risiken und Nebenwirkungen fragen Sie Ihren Arzt oder Apotheker

Im folgenden werden die Vor- und Nachteile der verschiedenen Programmarchitekturen unter dem Blickwinkel von Datensicherheit und -schutz verglichen.

Die Liste möglicher Gefährdungen ist umfangreich. Typisch sind

- Ausfall von Bauteilen, speziell Speichermedien
- Fehlschlag von Updates
- Zerstörung des Gerätes oder Daten durch Unfall oder Unglück (Sturz, Stromausfall, Feuer ...)
- Datenverlust durch Softwarefehler
- Bedienungsfehler
- Zugriffsverlust durch Verlust von Passwort oder digitalen Schlüsseln
- Kompromittierung von Updates mit Schadsoftware („Malware") im System des Herstellers
- elektronischer Einbruch in mein Gerät („Diebstahl" von Daten und Passwörtern, Verschlüsselung meiner Daten mit Erpressungsversuch),
- Fehler im lokalen Netzwerk
- Störungen in der Internetverbindung.

Nicht alles ist gleich wahrscheinlich, aber im Ernstfall genügt ja schon ein Ereignis, um einen in ernsthafte Schwierigkeiten zu bringen. Es gibt keine Lösung, die alles vollständig ausschließt. Jeder muss also für sich eine Abwägung vornehmen.

Ganz generell gilt: Der Zugang zur Hardware muss immer gut abgesichert sein, völlig unabhängig davon, welche der im folgenden diskutierten Softwarelösungen genutzt werden.

Ebenso wichtig ist eine große Achtsamkeit im Umgang mit Mails. Die Kompromittierung von Systemen durch Anklicken bösartiger Links oder Anhänge kann sehr üble Folgen haben. Da hilft keine der hier vorgestellten Softwarearchitekturen bei Anwendungsprogrammen.

Installationsbasierte Praxisprogramme

Als Nutzer solcher Programme habe ich die volle Verantwortung für alles, behalte aber auch die volle Kontrolle über meine Daten und bis vor kurzem grundsätzlich den Vorteil, nur einmal bezahlen zu müssen.

(Inzwischen werden aber auch in diesem Markt Abos immer üblicher.)

Um diese Kontrolle auch sicher aufrecht zu erhalten, muss man nicht zaubern können, aber es ist eben doch ein nicht zu unterschätzender regelmäßiger Aufwand. An allererster Stelle brauche ich Sicherungen, also Backups, Backups, Backups ... Ich will sagen, ich brauche eine professionelle Sicherungs-Strategie. Es genügt nicht, einfach nur ab und an meine Daten auf einen USB-Stick zu kopieren. Zu erklären, was das praktisch bedeutet, würde den Rahmen dieses Artikels sprengen.

Es gibt im Internet eine Menge guter Anleitungen für Sicherungsstrategien. Ein Beispiel findet sich auf der Webseite der "datenwache.de".

Zur professionellen Sicherungsstrategie kommt die Notwendigkeit von Systemaktualisierungen hinzu. Betriebssystem und Programme müssen stets aktuell gehalten werden, um die Funktionsfähigkeit und die Sicherheit zu gewährleisten. Auch wenn der Trend ja gerade in die andere Richtung geht, gibt es für freie Heilberufler nach wie vor die Option, einen Praxisrechner zu betreiben, der nicht ständig mit dem Internet verbunden ist. Für die heute eigentlich unumgängliche Verbindung ins Netz brauche ich dann allerdings einen zweiten Rechner, der keine sensiblen Daten enthält.

Der Ausfall der Hardware kommt nicht jeden Tag vor, ist aber besonders übel, wenn die Speichermedien betroffen sind. Spätestens dann weiß ich, wie gut meine Sicherungsstrategie wirklich war. Das gilt genauso, falls es jemandem gelungen ist, mir eine Schadsoftware unterzujubeln, die meine Daten verschlüsselt.

Aber auch in fast allen übrigen Gefährdungsfällen sehen wir, dass die Frage der Sicherungen der Dreh- und Angelpunkt ist. Sind diese stets aktuell, ist eine Systemwiederherstellung nur eine Zeitfrage und kein Desaster.

Allerdings muss man einräumen, dass eigentlich jeder Mensch Fälle kennt, in denen das gründlich schiefgegangen ist. Und Hand aufs Herz, wie zuverlässig machst du deine Datensicherungen wirklich? Wir sind Menschen, keine Automaten. Der Fehler sitzt oft vor dem Gerät.

Deshalb ist es in meinen Augen signifikant sicherer, als Speicherort für meine Daten ein inhaltsverschlüsselndes virtuelles Laufwerk eines „Cloud"-Speicher-Anbieters zu wählen, wodurch diese Daten dann automatisch im Hintergrund auf dem beim Anbieter gebuchten Speicherplatz gesichert werden. Vereint damit bilden dann zusätzliche automatisierte lokale Sicherungen auf z. B. ein Netzlaufwerk die Brandmauer gegen Totalverluste. (Das mache ich grundsätzlich für alle wichtigen Daten in meinen Systemen.) Störungen der Netzverbindung sind da nicht so relevant wie bei einer Online Software. Datensicherheit und -schutz sind bei einer solchen Lösung schon mal auf einem ganz anderen Niveau.

Der Aufwand zur Programm- und Datenpflege bei einem Praxis-Zentralrechner ist prinzipiell der Gleiche wie bei einem echten Einzelplatz-Rechner, obwohl beliebig viele Arbeitsplätze mit den Praxisprogramm verbunden sind, aber natürlich kommt die Pflege der vervielfachten Anzahl an Rechnern, des lokalen Netzwerks und das Verwalten von Zugangsberechtigungen hinzu.

Klassische Online Web App

Verwende ich eine „Cloud"-basierte All-in-One Praxis-Web App verlagern sich die Gefährdungen. Denn was auch immer ich mit meinem Gerät anstelle, die Programmdateien sind ständig verfügbar, meine Daten sind beim Anbieter der Web App gespeichert, und ich kann nach der Beseitigung eines Schadens bei mir schnell oder von einem anderen Gerät aus auch sofort wieder darauf zugreifen. Der Preis dafür besteht darin, dass ich nun aber von einer funktionierenden Netzverbindung abhängig bin. Das ist im allgemeinen nicht das Problem, manchmal aber eben doch, und es genügt ja schon eine Störung im lokalen WLAN, dass ich dann eben doch an meinen Kalender im entscheidenden Moment nicht heran komme (auch selbst schon erlebt).

Ich lege die Pflege des Programms sowie die Aufbewahrung und (verschlüsselte) Sicherung meiner Daten in die Verantwortung von Menschen, die das als Dienstleistung anbieten und denen ich dafür etwas bezahle. Damit ist die gesamte Praxis-Datenverarbeitung entkoppelt vom eigenen Gerät, von dessen „Gesundheit" alles abhängt.

All die Gefährdungen, die ich für die eigene Datenverarbeitung aufgelistet habe, bestehen natürlich auch für alle Anwendungen in der „Cloud". Ich sehe es nur nicht mehr. Die Gefahr liegt hier ein bisschen in dem Prinzip „Aus den Augen, aus dem Sinn". Die beschriebenen Vorfälle zeigen ja, dass auch unter diesen Umständen Unerfreuliches bis Dramatisches passieren kann. Meiner Meinung nach sollte auch eine Web App die Option einer zusätzlichen lokalen Datensicherung anbieten.

Denn es heißt zwar „Cloud", aber in diesem Fall erinnert nur der Begriff an eine Wolke. Real arbeiten die Programme auf Rechnern mit Prozessoren, Arbeitsspeichern und Datenträgern.

Streng genommen habe ich es im allgemeinen mit mindestens zwei Anbietern zu tun: dem, der mir die Web App vermietet, und dem, der im Hintergrund die Infrastruktur anbietet, bei dem wiederum der Web App-Anbieter Kunde ist (Marktführer sind hier die Amazon Web Services). Die Infrastrukturarchitekturen der großen Cloud-Anbieter sind mittlerweile derart komplex, dass ein einziger größerer Fehler im Gesamtsystem zur Unerreichbarkeit meiner Anwendung führen und meine Arbeit lahmlegen kann.

Die beschriebenen Daten-„Diebstähle" von außen beruhten auf Sicherheitslücken in den Systemen oder Fehlern von Mitarbeitern. Leider sind solche Sicherheitslücken und auch Fehler von Menschen letztlich unvermeidlich. Das ist die Realität. Ein Anbieter kann nicht mehr tun, als ein System nach dem Stand der Technik zu designen und zu pflegen und alle bekannt werdenden Sicherheitslücken durch Updates jeweils schnellstmöglich zu schließen.

Noch fataler sind Angriffe von „innen", also von Mitarbeitern des Anbieters, die von Berufs wegen Zugang zu den Systemen haben. Das geschieht sehr selten, ist aber schon passiert.

Von innen wie von außen besteht prinzipiell die Möglichkeit, die Web App selbst mit Schadsoftware zu kontaminieren, die dann bei der Benutzung des Programms an alle gerade aktiven Nutzer über den Browser verteilt wird. Ist sie so programmiert, dass sie über eine noch unbekannte Lücke aus dem Browser „ausbricht", ist der potenzielle Schaden über die Maßen groß.

Und schließlich gibt es noch eine weitere Art des Angriffs, die sich „Man-in-the-Middle"-Angriff nennt. Dabei gelingt es jemandem, sich irgendwo auf dem Weg zwischen dem Server und meinen Browser auf einer Zwischenstation einzuklinken, um den Datenstrom zu manipulieren. Sei es, dass Daten „ausgeleitet" werden, Schadsoftware untergeschoben oder bei clientseitigen Web Apps der Programmcode verändert wird.

Und noch ein Aspekt tritt hinzu: Der Zugriff auf eine Web App, die sensible Daten zentral verwaltet, muss gesichert und geschützt sein. Noch heute gibt es zahlreiche Dienste, die nur durch ein einziges Passwort geschützt sind bzw. einen zweiten Authentifizierungsfaktor nicht obligatorisch verlangen. Das war bis vor kurzem selbst bei Bank- und Zahlungssystemen noch so. Das hat sich glücklicherweise geändert.

Passwörter können schwach sein, gestohlen oder vergessen werden. Passwörter müssen ebenfalls beim Anbieter gespeichert werden. Und immer wieder mal werden sie dort erfolgreich abgegriffen.

Der Speicher im Browser ist dafür nicht geeignet! Ein zweiter Faktor neben dem Passwort bringt ganz klar eine zusätzliche Sicherheit.

Ganz langsam kommen auch Authentifizierungstechniken in breitere Verwendung, die auf Passwörter ganz verzichten können. Diese bieten dann einen Schlüssel auf einem externen Medium oder Gerät, welches auch dann unzugänglich bleibt, wenn mein Rechner „übernommen" wurde.

Zusammenfassung

Einerseits bietet die Verwendung einer Web App als Praxissoftware sehr alltagstaugliche Vorteile. Sie befreit mich von einigen Notwendigkeiten und Aufgaben, die installationsbasierte Lösungen unvermeidlicherweise mit sich bringen. Für mich wiegen diese so schwer, dass ich für die meisten Aufgaben Weblösungen benutze, trotz der Gefährdungen, die ich gerade skizziert habe.

Jeder sollte sich aber bewusst sein, dass man dafür eben doch eine Reihe von Risiken eintauscht, die man selbst nicht wirklich steuern kann. Hinter der Nutzeroberfläche im Endgerät befindet sich ein hochkomplexes System. Ein erfolgreicher Angriff auf dieses System betrifft schnell viele, manchmal auch alle Kunden.

Serverseitige Online Web Apps können recht einfach mit zusätzlichen anderen Onlinediensten verbunden werden. Das erweitert einerseits deren Fähigkeiten, vergrößert aber auch die Zahl potentieller Angriffsflächen.

Bei der Mehrzahl der (auch von mir selbst verwendeten) Webanwendungen muss ich damit leben, dass meine Daten im Verlaufe des Verarbeitungsprozesses in den Anbietersystemen nicht ununterbrochen verschlüsselt sein können. Und dass für die Bereiche, in denen sie verschlüsselt sind, der Anbieter Zugriff auf den Schlüssel haben muss. Ich sollte also zumindest darauf achten, dass die Rechenzentren der Anbieter in der EU, besser noch in Deutschland stehen.

Local- oder Offline-first Web Apps

An einer Stellschraube kann ich noch drehen, die die Vor- und Nachteilsverteilung bei Web Apps noch einmal verändert. Wenn die gesamte Datenverarbeitung im Gerät der Nutzer abläuft, braucht niemand außerhalb meiner Geräte lesbare Daten, sie können also inhaltsverschlüsselt werden.

Die Risiken, die mit der Notwendigkeit einer Entschlüsselung von Daten bei einer serverseitigen Web App verbunden sind, egal, wie hoch ich sie real veranschlage, sind dann minimiert. Wenn diese Inhaltsverschlüsselung allerdings passwortbasiert gebaut ist, bedeutet der Verlust des Passwortes den Verlust der Daten. Ist der Schlüssel nicht gleichzeitig das Passwort, kann er getrennt vom Passwort und verschlüsselt ebenfalls beim Diensteanbieter gespeichert werden. Der Anwender sollte dann jedoch einen Masterkey herunterladen und auf einem externen Medium speichern. Diesen Key braucht er bei Passwortverlust für die Datenwiederherstellung. Ergo: Will ich keine lesbaren Daten im Netz, habe ich wieder eine höhere Verantwortung.

Der Anbieter hat den Vorteil einer weniger komplexen Architektur. Er muss (genau wie bei serverseitigen Web Apps) einen Webserver gesichert und sauber betreiben, aber keine komplexe Anwendungsserver-Architektur. Wenn überhaupt etwas serverseitig läuft, ist es i. d. R. schlank. Die Datenspeicherungsarchitektur muss natürlich eine ebenso hohe und schnelle Verfügbarkeit sowie georedundante Sicherungen bieten wie die anderen Lösungen. Die eigentliche Programmaktivität findet dezentral auf den Geräten der Nutzer statt.

Benutze ich mehrere Geräte, ist der Schlüssel bei der verschlüsselten Speicherung beim Anbieter auf allen diesen Geräten automatisch benutzbar.

Benutzt eine local-first Web App die Browserdatenbank, um die Anwenderdaten lokal zu speichern, kann das Ganze zu einer offline-first Web App ausgebaut werden. Auf meinen Geräten liegt dann erstens immer eine weitere Kopie der Daten (bei manchen Anwendungen mit einem Button zum manuellen Export auf ein zusätzliches externes Medium).

Zweitens bedeutet Offlinefähigkeit, dass ich Störungen der Internetverbindung überbrücken kann. Wichtig ist, dass das bei Patientendaten nur zulässig ist, wenn diese lokale Speicherung ebenfalls verschlüsselt erfolgt.

Leistungsschwächere Geräte wie Tablets sind bei komplexen Anwendungen mit hohem Datenaufkommen und mehrfachen Ver- und Entschlüsselungsprozeduren durchaus mal so stark gefordert, dass es zu verzögerten Antwortzeiten bei der Verarbeitung von Daten kommt. Da ist dann leistungsstarke neuere Hardware die Voraussetzung.

Bei inhaltsverschlüsselnden clientseitigen Apps ist es schwieriger bis unmöglich, weitere Onlinedienste hinzuzufügen, ohne dafür Funktionen unverschlüsselt auf den Server zu verlegen.

Ich hoffe, ich konnte zeigen, dass jede Softwarearchitektur ihre jeweils spezifischen Merkmale aufweist, mit teils unterschiedlichen Konsequenzen für den Workflow und die damit verbundenen Datensicherheitsrisiken. Ich muss als Anwender entscheiden, welche davon meinem Umgang mit IT und meiner Einschätzung der jeweiligen Risiken entsprechen. Bedienungskomfort und Sicherheit stehen in einigen Aspekten in einem unvermeidlichen Spannungsverhältnis.

Schlussbemerkungen

1. Sei in jedem Falle freundlich zu deiner Hardware, halte Betriebssystem und Software aktuell. Es ist immer gut, auch ein funktionsfähiges Ersatzgerät in der Hinterhand zu haben. Ganz egal, mit welcher Softwarelösung du arbeitest.

2. Misstraue den Mailanhängen und -links! Die Trickser und Täuscher werden immer schlauer!

3. Arbeitest du (überwiegend) mit selbst installierter Software hast du die vollständige Kontrolle, aber auch die volle Verantwortung für Benutzbarkeit und Datensicherheit. Ein Zugriff auf deine sensiblen Daten könnte nur über dein Gerät erfolgen. Regelmäßige System- und Datensicherungen sind Pflicht!

4. Verwendest du (auch) „gemietete" Software, also Online-Anwendungen, kannst du einen bedeutenden Teil der Verantwortung für die Software-Aktualität und die Datensicherheit und –sicherung an Profis auslagern.

5. Bei Online-Anwendungen, die serverseitig arbeiten, musst du akzeptieren, dass es deine Daten auch außerhalb deines eigenen Gerätes verarbeitet werden. Dafür kann dein Anbieter aber auch beliebig viele weitere Onlinedienste dranschrauben.

6. Bei serverseitigen und „local-first" Online-Anwendungen ohne lokale Datenspeicherung geht nichts ohne Netzverbindung. Ein Mobilfunk-Fallback ist Pflicht.

7. Bei vollständig „local- und offline-first" Online-Anwendungen mit Inhaltsverschlüsselung gibt es Datenverarbeitung nur in deinem Gerät, niemals auf Servern im Netz, vergleichbar mit installierten Programmen. Dafür ist es schwieriger bis unmöglich, weitere Onlinedienste hinzuzufügen, ohne eben doch Teile der Anwendung unverschlüsselt auf den Server zu verlegen.

8. Du musst nun aber für den Schlüssel die Verantwortung übernehmen. Nicht gleich für die ganze Datensicherheit und Datensicherung, aber eben doch für etwas wesentliches.

Dr. rer. nat. Thomas Naujokat
Dipl.-Psych., Dipl.-Biol., Heilpraktiker für Psychotherapie in eigener Praxis
thomas@naujokat.de
https://tuuva.systems