OPTIMAL SYSTEMS
A KYOCERA GROUP COMPANY

Whitepaper
Sicherheit enaio® mobile, enaio® webclient, enaio® webclient als Desktop-Anwendung

Version 4.2.1
Gültig ab: 14.10.2025
Autor: Research & Development

Änderungsübersicht und Freigabeinformationen

Version	Erstellt durch	Erstellt am	Änderung	Freigegeben durch	Freigegeben am
1.0.0	R. Espenhahn	05.02.2013		S. Hattenbach	06.02.2013
2.0.0	R. Espenhahn	12.04.2013		S. Hattenbach	12.04.2013
3.0.0	R. Espenhahn	10.12.2018		R. Espenhahn	11.12.2018
3.3.0	R. Espenhahn	15.06.2020	Anpassung auf neuen Webclient, Webclient als Desktop Anwendung und enaio mobile	R. Espenhahn	15.06.2020
3.3.2	R. Espenhahn	28.03.2022	Anpassungen an neuere Version	R. Espenhahn	28.03.2022
4.0.0	R. Espenhahn	19.08.2022	Anpassungen an neuere Version	N. Skundric	06.10.2022
4.1.0	T. Schmeißel	19.08.2024	Ergänzung Sicherheitsarchitektur	N. Skundric	22.08.2024
4.2.0	T. Schmeißel	28.11.2024	Anpassung Dokumentstruktur	T. Schmeißel	29.11.2024
4.2.1	T. Schmeißel	06.10.2025	Minimale Ergänzungen	H. Kühn	06.10.2025

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Inhaltsverzeichnis

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Einleitung

Mobile Anwendungen werden überwiegend unterwegs verwendet. Sie laufen auf kleinen, portablen Smartphones oder Tablets. Aufgrund der Größe und Portabilität sind sie jedoch leicht durch Dritte in einem unaufmerksamen Moment zu entwenden. Eine Entwendung hat oftmals nicht nur die Folge, dass das mobile Endgerät verloren ist, sondern auch die auf dem Gerät gespeicherten Daten. Der Wert sensibler persönlicher Daten oder Unternehmensdaten ist oftmals beträchtlich größer als der Wert des eigentlichen mobilen Endgeräts. Aus diesem Grund müssen mobile Anwendungen, die mit sensiblen Daten arbeiten, auch bei einem Diebstahl des mobilen Endgeräts sicherstellen, dass diese nicht in die Hände Dritter fallen.

Neben dem materiellen Diebstahl des mobilen Endgeräts existiert auch der virtuelle Diebstahl von Daten durch bösartige Apps, die aus Unkenntnis oder durch Dritte auf dem mobilen Endgerät installiert wurden. Solche Apps lesen oft sensible Daten aus, die auf dem mobilen Endgerät verarbeitet werden, und senden sie an den Hersteller der bösartigen App, der diese für seine Interessen nutzt.

Alles, was für mobile Apps gilt, gilt in gleicher Weise auch für Webanwendungen wie enaio® webclient. Auch dieser kann mobil auf dem Smartphone, Tablet oder Laptop einfach im Browser genutzt werden und es gelten somit dieselben Sicherheitsansprüche wie für mobile Apps.

Durch clientseitiges Skripting können enaio® webclient und enaio® mobile stark angepasst werden. Clientskripte werden dazu auf dem Rechner oder mobilen Endgerät des Benutzers ausgeführt. Dies zieht mehrere sicherheitsrelevante Aspekte mit sich, wie die Übertragung der Skripte und die sichere Ausführung dieser, sodass die Ausführung seitens des Benutzers oder eines Dritten nicht manipuliert werden kann.

In diesem Whitepaper vermitteln wir Sicherheitsmaßnahmen für enaio® webclient, enaio® mobile und enaio® webclient als Desktop-Anwendung. Zu Anfang stellen wir die Sicherheitsmaßnahmen von Apple iOS und Google Android für Apps vor, gehen von dort weiter zu Sicherheitsmaßnahmen von Clientskripten bis hin zu Sicherheitsmaßnahmen im Unternehmensnetzwerk und den Einsatz von Mobile-Device-Management-Lösungen.

Sicherheitsmaßnahmen von Apple iOS und Google Android

Sowohl Apple iOS als auch Google Android nutzen ein Sandboxkonzept, bei dem Apps voneinander abgeschirmt werden. Einer App ist es somit nicht möglich auf Daten anderer Apps zuzugreifen. Diese Sicherheitsmaßnahme seitens des Betriebssystems beugt bereits dem virtuellen Datendiebstahl von Apps vor, ohne dass in den installierten Apps spezielle Vorkehrungen getroffen werden müssen. Beide Betriebssysteme bieten darüber hinaus die Möglichkeit, einen PIN-Code oder eine biometrische Authentifizierung einzurichten, um das Gerät vor unbefugtem Zugriff zu schützen. Zudem können sie so konfiguriert werden, dass alle Daten auf dem Gerät automatisch gelöscht werden, wenn die Authentifizierung mehrfach fehlschlägt. Diese Funktionen sind jedoch nur effektiv, wenn der Benutzer sie aktiviert und ein sicheres Passwort wählt.

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Für die Verschlüsselung von Datenübertragungen unterstützen beide Betriebssysteme moderne Standards wie WPA3, WPA2 und VPN bei der Nutzung von WLAN-Netzen sowie 4G/5G/LTE für Mobilfunkverbindungen, wodurch die Sicherheit der Datenkommunikation gewährleistet wird. Damit bieten iOS und Android vergleichbare Sicherheitsmaßnahmen auf Betriebssystemebene.

Sicherheitsmaßnahmen von enaio® mobile

Zusätzlich zu den betriebssystemseitigen Sicherheitsmaßnahmen verfügt die App enaio® mobile von OPTIMAL SYSTEMS über weitere Sicherheitsfunktionen.

Um Daten über die App von enaio® server abrufen zu können, muss sich die App an enaio® gateway anmelden. Die Authentifizierung erfolgt hierbei über Basic Authentication (RFC 2617), die den Benutzernamen und das Passwort als Base64-kodierte Zeichenketten an enaio® gateway sendet.

Da bei dieser Art der Authentifizierung der Benutzername und das Passwort unverschlüsselt über das HTTP-Protokoll versendet werden, sollten Sie enaio® mobile mit dem gesicherten HTTPS-Protokoll an enaio® gateway anbinden. Wenn das HTTPS-Protokoll für die Übertragung eingesetzt wird, erfolgt die gesamte Datenkommunikation zwischen enaio® mobile und enaio® gateway verschlüsselt. Der Einsatz von TLS wird dringend empfohlen, da die Verschlüsselung der Datenverbindung im Mobilfunk nur bis zum jeweiligen Accesspoint erfolgt und danach die Daten unverschlüsselt weitergeleitet werden. Der Vorgänger SSL wird nicht mehr empfohlen. SSL und TLS gewährleisten eine Ende-zu-Ende-Verschlüsselung und Infrastrukturelemente in dieser Verbindung können die Daten nur sehr schwer entschlüsseln.

Die Zugangsdaten zur Authentifizierung des Benutzers an enaio® gateway werden innerhalb von enaio® mobile mittels des Cordova-Plug-ins cordova-plugin-secure-storage sicher gespeichert. Unter iOS werden die Passwörter dazu in der Schlüsselbundverwaltung gespeichert. Android hingegen besitzt betriebssystemseitig kein Äquivalent zur Schlüsselbundverwaltung. Aus diesem Grund werden die Passwörter hier mittels einer AES-Verschlüsselung gespeichert.

Ist das mobile Endgerät während der Anmeldung offline, so werden die eingegebenen Anmeldedaten gegen die zuletzt für das ausgewählte Profil hinterlegten Anmeldedaten validiert. Stimmen diese überein, so wird enaio® mobile offline aus dem Cache gestartet.

Bei erfolgreicher Authentifizierung beim Start von enaio® mobile an enaio® gateway werden von der laufenden App im Onlinefall viele empfangene Daten in einem Cache zwischengespeichert. Die Daten werden ebenfalls innerhalb der Sandbox der App abgelegt und sind durch die Sandbox vor dem Zugriff Dritter geschützt. Hierbei werden innerhalb von enaio® mobile mehrere IndexedDBs verwendet. Die Cachedaten innerhalb der IndexedDBs überdauern auch App-Neustarts und sorgen z. B. für einen schnelleren Boot ab dem zweiten Mal, da nicht alle Daten neu übertragen werden müssen, was zudem Bandbreite spart. Die Authentifizierung beim Start von enaio® mobile beugt ebenfalls einem unautorisierten Zugriff Dritter, die das Endgerät entsperrt in die Hand bekommen, auf enaio® mobile vor. Es kann zwar ein Autologin eingerichtet werden, aber dies wird für mobile Endgeräte nicht empfohlen.

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Neben dem Caching werden Favoriten- und Offlinedokumente inklusive Indexdaten offline zugänglich gemacht. Diese Funktion ermöglicht es, Dokumente und Indexdaten auch ohne bestehende Internetverbindung bei einem Kundentermin verfügbar zu haben. Wenn ein ECM-Ordner oder -Register den Favoriten hinzugefügt oder für die Offlinenutzung gekennzeichnet wird, so werden der Ordner oder das Register zusammen mit seinen Kindobjekten zur Offlinenutzung synchronisiert. Offline verfügbare Objekte werden ebenfalls in einer IndexedDB, die durch die Sandbox des Betriebssystems vor dem Zugriff anderer Apps gesichert ist, gespeichert. Die Synchronisation und das Speichern der Offlineobjekte erfolgt automatisch, sobald ein Dokument innerhalb der App als Favorit oder zur Offlinenutzung gekennzeichnet wird. Wird ein Objekt mittels enaio® mobile aus den Favoriten entfernt oder sein Offlinestatus entfernt, so entfernt enaio® mobile das Dokument direkt aus der IndexedDB. Wird ein Objekt über enaio® client zu den Favoriten hinzugefügt oder aus den Favoriten entfernt, so fügt enaio® mobile das Objekt bei der nächsten Synchronisierung hinzu oder entfernt es aus dem Offline-Cache.

Sicherheitsmaßnahmen enaio® webclient als Desktop-Anwendung

enaio® webclient als Desktop-Anwendung liegt die Open-Source-Plattform Electron zu Grunde. Electron wird vornehmlich von GitHub Inc. entwickelt. Identisch zu enaio® mobile muss sich ein Benutzer mit einem Profil an enaio® webclient als Desktop-Anwendung anmelden. Die Validierung erfolgt hierbei identisch zu enaio® mobile.

Im Gegensatz zu enaio® mobile läuft enaio® webclient als Desktop-Anwendung nicht in einer Sandbox und ist somit unter Windows mit seinen Daten nicht separiert von allen anderen Apps und Programmen. Andere Programme können somit auf alle Daten des enaio® webclient als Desktop-Anwendung zugreifen. Identisch zu enaio® mobile speichert enaio® webclient als Desktop-Anwendung alle Cachedaten in IndexedDBs. Das Datenverzeichnis liegt im Roaming-Ordner des am Betriebssystem angemeldeten Benutzers. Dort finden sich ebenfalls die Datendateien zu den IndexedDBs. Werden diese Dateien jedoch näher betrachtet, so sind sie, wie es bei fast allen Datenbanken der Fall ist, in einem proprietären Format gespeichert, um gute Lese- und Schreibgeschwindigkeit zu bieten. Recherchen unsererseits haben ergeben, dass es keine einfachen Programme gibt, um diese IndexedDBs zu lesen und somit sind die Daten innerhalb dieser Datenbankdateien hinreichend sicher. Trotzdem dieser Sachverhalt weiterhin untersucht und sollten sich daran Zweifel ergeben, werden weitere Maßnahmen zur Datensicherheit der Cachedaten von enaio® webclient als Desktop-Anwendung umgesetzt.

Identisch zu Apple iOS werden Profilpasswörter hier im systemeigenen sicheren Schlüsselbund gespeichert. enaio® webclient als Desktop-Anwendung nutzt dazu die Open-Source-Bibliothek Keytar (KT).

Sicherheit von Clientskripten

In den nachfolgenden Kapiteln widmet sich dieses Whitepaper der Thematik der Clientskripte. Es wird gezeigt, wie die Ausführung abgesichert werden kann und wieso dies notwendig ist. Zum Anfang dieser Kapitelserie soll jedoch auf einen zentralen Punkt in Bezug auf Clientskripte hingewiesen werden.

Die nachfolgend dargelegten Maßnahmen versuchen die Ausführung von Clientskripten unter enaio® mobile und enaio® webclient als Desktop-Anwendung sicher zu machen. Es sollte jedoch bekannt sein, dass alle Sicherheitsmaßnahmen, die irgendwo getroffen werden, ausschließlich dazu da sind, unbefugten Personen den Zugriff auf die Clientskripte so schwer und unattraktiv wie möglich zu machen. Werden sicherheitstechnische Hürden auf einem Computer trotzdem geknackt, so sollten im Hintergrund, auf der Serverseite, weitere Absicherungen existieren.

Alle Anfragen (Suche, Aktualisierung, Neuanlage, Löschen, etc.), bei denen mittels Clientskripten Felder als unsichtbar oder schreibgeschützt gesetzt werden bzw. sonstige Sicherheit durchgesetzt werden soll, sollten serverseitig nochmals quergeprüft werden. Dies hat nicht nur als Hintergrund, dass die Clientskripte kompromittiert werden können. Vielmehr beziehen enaio® mobile, enaio® webclient und enaio® webclient als Desktop-Anwendung neben vielen anderen Komponenten seine Daten aus öffentlichen Schnittstellen wie den enaio® Microservices. Sämtliche Anfragen, die einer der Clients an einen der Microservices sendet, können genauso gut manuell durch eine unbefugte Person im Kontext des aktuell angemeldeten Benutzers über diesen abgesetzt werden.

Neben den enaio® Microservices existieren bei enaio® viele weitere öffentlich dokumentierte Schnittstellen, über welche Daten in enaio® server manipuliert werden können. Dazu zählen z. B. enaio® appconnetor und die COM-API. Diese öffentlichen Schnittstellen existieren heutzutage auch bei einer enaio® client-Installation, da sie von dieser ebenfalls benötigt werden. Durch seine proprietäre Art bietet enaio® client einen guten Schutz vor der Manipulation der Clientskripte. Eine unbefugte Person kann jedoch auf die öffentlichen Schnittstellen ausweichen, um eine Datenmanipulation vorzunehmen. Es ist somit auch hier zwingend erforderlich, dass zur Durchsetzung der Sicherheit alleinig enaio® server mit seinem Rechtesystem und seinen Serverskripten verantwortlich sein sollte. Alleinige clientseitige Sicherheit bietet ausschließlich trügerische Sicherheit und keine reale Sicherheit.

Sicherheitsmaßnahmen für Clientskripte unternehmensintern

Clientskripte bieten die Möglichkeit viele Aspekte von enaio® mobile und enaio® webclient an die projektspezifischen Anforderungen anzupassen. Andererseits können sie auch zu einer großen Sicherheitslücke werden, wenn deren Ausführung nicht ordentlich gesichert ist oder sie falsch konzipiert wurden.

Clientskripte werden, im Browser des Benutzers ausgeführt. Dies bedeutet an jedem Arbeitsplatz eines enaio® mobile- oder enaio® webclient-Benutzers. Die Javaskripte werden von enaio®

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

appconnector als JSON abgerufen und dann im jeweiligen Browser und in enaio® mobile zur festgelegten Aktion zur Ausführung gebracht. Jeder Browser verfügt zum Testen und Entwickeln von Javaskript eine Entwicklerkonsole, die sich über die Taste F12 öffnen lässt. Über die Entwicklerkonsole eines Browsers lassen sich die Clientskripte im Klartext einsehen und auch temporär modifizieren. Weitergehend kann darüber an einem gesetzten Breakpoint neuer Javaskriptcode mit sofortiger Wirkung ausgeführt und sämtliche HTTP-Aufrufe eingesehen sowie modifiziert zur erneuten Ausführung gebracht werden. Dies sind nur drei Aspekte der Entwicklerkonsole, die sehr viele weitere Möglichkeiten der Manipulation von Webanwendungen bereitstellt.

Während die Entwicklerkonsole im Zuge der Entwicklungs- und Testphase eines Projektes als unerlässliche Hilfe benötigt wird, kann sie in der Produktivphase als Backdoor angesehen werden. Es wird deswegen empfohlen, die Entwicklerkonsole über Gruppenrichtlinien unter Windows oder Konfigurationsdateien betriebssystemübergreifend für den Großteil der Benutzer unzugänglich zu machen. Die von enaio® mobile und enaio® webclient unterstützten Browser bieten hierzu alle ein bis zwei konfigurierbare Möglichkeiten an.

Eine durchgängige Möglichkeit, die jedoch auf Microsoft-Windows-Betriebssysteme beschränkt ist, ist die der Gruppenrichtlinien. Jeder Browserhersteller bietet einen eigenen Satz an Gruppenrichtlinien an, über welche sich die Entwicklerkonsole deaktivieren lässt. Neben den Gruppenrichtlinien für Windows bieten Mozilla und Google für ihre Browser auch die Möglichkeit der Deaktivierung über eine Konfigurationsdatei an. Diese liegt an einem Ort im Dateisystem, auf den der Benutzer keine Schreibrechte hat, um sie zu ändern.

In Mozilla Firefox führt die Aktivierung der Gruppenrichtlinie *Werkzeuge für Webentwickler deaktivieren* (MOZ) dazu, dass alle Benutzer dieser Gruppenrichtlinie die Entwicklerkonsole nicht mehr öffnen können. Alternativ kann im Firefox-Unterverzeichnis `distribution` die Konfigurationsdatei `policies.json` mit folgendem JSON-Inhalt hinterlegt werden:

```
{
      "policies": {
            "DisableDeveloperTools": true
      }
}
```


Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

In den Gruppenrichtlinien für Google Chrome (GC1) muss *Festlegen, wo Entwicklertools verwendet werden können* aktiviert und mit der Option *Nutzung der Entwicklertools nicht zulassen* versehen werden. Unter Linux oder Mac OS X kann ein Konfigurationsschlüssel (GC2) gesetzt werden, um die Entwicklerkonsole im Chrome Browser unzugänglich zu machen.

Der Microsoft Edge Browser verfügt ausschließlich über die Möglichkeit der Sperrung via Gruppenrichtlinie. Die betreffende Richtlinie wird bereits von Windows mitgebracht und es ist nicht notwendig zusätzliche Richtlinienvorlagen aus einer externen Quelle zu importieren. Im Edge Browser muss die Gruppenrichtlinie *Windows Components\Microsoft Edge\Allow Developer Tools* deaktiviert werden.

Für den Safari Browser unter Mac OS X konnte im Zuge der Aktualisierung dieses Whitepapers keine Möglichkeit der Deaktivierung gefunden werden. Wir haben bei Apple angefragt und warten hier auf eine Rückmeldung. Mit der nächsten Aktualisierung hoffen wir hier ebenfalls eine Möglichkeit darlegen zu können.

Sicherheitsmaßnahmen für Clientskripte enaio® mobile

Unter Apple iOS und Google Android existiert auf einem Tablet oder Smartphone keine Entwicklerkonsole, die es zu deaktivieren gilt, um die Sicherheit der Clientskripte zu gewährleisten. Wurde enaio® mobile aus dem Apple App Store oder Google Playstore auf dem mobilen Endgerät installiert, so ist auch kein Remote Debugging über Safari unter Mac OS X oder Google Chrome möglich, da die App nicht als Debugging-Ziel zur Verfügung steht. Somit sind Clientskripte unter enaio® mobile sicher, wenn sie gesichert via HTTPS an enaio® mobile übertragen wurden.

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Sicherheitsmaßnahmen für Clientskripte externer Mitarbeiter

Arbeiten von unterwegs und zu Hause ist in der heutigen Arbeitswelt sehr weit verbreitet. Daher wird oft auch der Zugriff auf das Unternehmensnetzwerk über das Internet ermöglicht. Wenn in einem Projekt enaio® dieser Zugriff ermöglicht werden soll, sind weitere Sicherungsmaßnahmen für Clientskripte erforderlich. Im Folgenden werden drei Szenarien beleuchtet.

Szenario 1

enaio® webclient soll keinen Zugriff erhalten. Von extern und aus dem Internet darf ausschließlich mit mobilen Apps zugegriffen werden.

In diesem Fall kann der enaio® webclient-Microservice, welcher die Datei `index.html` des enaio® webclient ausliefert, mittels User-Agent-Restriktion so konfiguriert werden, dass er diese nicht ausliefert. Diese Datei bildet den Einstieg in enaio® webclient und wenn sie extern nicht verfügbar ist, kann enaio® webclient nicht aus dem Internet aufgerufen werden. Die Apps enaio® mobile für iOS und Android sowie enaio® webclient als Desktop-Anwendung benötigen diese Datei nicht, da sie diese selbst mitbringen.

Szenario 2

enaio® webclient soll Zugriff über Browser von Computern erhalten.

In diesem Fall stellt sich die Absicherung schwerer dar, als nur die Einstiegsdatei im enaio® webclient-Server zu blockieren. In dieser Konfiguration haben alle Benutzer, die die Unternehmensfirewall passieren dürfen und sich an enaio® gateway anmelden können, Zugriff auf enaio® webclient egal von welchem Computer sie zugreifen. Private Computer sind im Normalfall nicht in die Unternehmensdomain integriert, sodass auch keine Gruppenrichtlinien bei diesen angewendet werden können. Somit greifen die zuvor genannten Möglichkeiten der Deaktivierung der Entwicklerkonsole in den Browsern bei privaten Computern nicht. Es ist auch so nahezu unmöglich private Computer abzusichern, weswegen wir hier nachfolgend zwei Alternativen aufzeigen.

Die erste Alternative ist, dass sich alle externen Mitarbeiter auf einen Terminalserver von extern gesichert einwählen müssen und auf diesem dann remote arbeiten. Die Terminalserversitzung ist dann wieder unternehmensintern und die verwendeten Browser können über Gruppenrichtlinien, wie zuvor beschrieben, abgesichert werden.

Alternative zwei ist, dass das Netzwerk zwar zum Internet gesichert geöffnet ist, aber Mitarbeiter sich ausschließlich mit Unternehmenslaptops an enaio® gateway anmelden dürfen. Auf den Unternehmenslaptops wird dazu mit Clientzertifikaten gearbeitet, die auf den berechtigten Laptops im Betriebssystem hinterlegt sind. Vor enaio® gateway wird dann ein Clientzertifikatsproxy eingerichtet, der diese Verbindungen annimmt und validiert. Das heißt, wenn Mitarbeiter von zu Hause aus mit Unternehmenslaptops arbeiten, haben sie Zugriff auf enaio® webclient. Arbeiten sie mit privaten Laptops, haben sie keinen Zugriff. Da dieses Szenario recht komplex werden kann, wenden Sie sich

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

zwecks Informationen an unseren Support oder in Projekten an die Abteilung Consulting von OPTIMAL SYSTEMS.

Szenario 3

enaio® webclient soll Zugriff über Browser von mobilen Endgeräten erhalten.

enaio® webclient läuft ebenfalls auf Tablets und Smartphones im mobilen Browser. Wie zuvor beschrieben ist enaio® mobile ohne weiteres Zutun auf diesen mobilen Endgeräten als sicher zu betrachten. Mobile Browser sind hingegen nicht sicher. Unbefugte Personen können das Tablet oder Smartphone via USB oder WLAN mit einem Computer verbinden, der nicht zum Unternehmensnetzwerk gehört und somit nicht den Gruppenrichtlinien unterliegt. Durch die Aktivierung des Remotedebuggings auf dem mobilen Endgerät und die Verbindung zum Computer kann auf dem Computer remote Debugging erfolgen. In Apple Safari, Google Chrome oder Mozilla Firefox auf dem Computer kann dann die Entwicklerkonsole für den Browser auf dem mobilen Endgerät geöffnet werden und enaio® webclient, der auf dem mobilen Endgerät im Browser läuft und sich ggf. sogar über ein Clientzertifikat gegenüber enaio® gateway autorisiert, kann manipuliert werden.

Um dieses Szenario ebenfalls abzusichern, ist eine Kombination mehrerer hier vorgestellter Techniken erforderlich. Einerseits sollten nur mobile Endgeräte auf enaio® webclient zugreifen dürfen, die z. B. über ein gültiges Clientzertifikat verfügen. Weiterhin muss das Remotedebugging auf dem mobilen Endgerät unterbunden werden. Zur Unterbindung des Remotedebuggings bedarf es eines Mobile Device Managements (MDM). MDMs können Richtlinien auf mobilen Endgeräten durchsetzen wozu auch gehört, dass Remotedebugging dauerhaft deaktiviert sein muss oder soll. *Muss* bedeutet, dass der/die Browser nicht mehr starten, solange die Betriebssystemeinstellung des Remotedebuggings aktiviert ist. *Soll* bedeutet, dass die Betriebssystemeinstellung seitens des Benutzers nicht geändert werden kann und abgeschaltet ist. Auf dem Markt existieren mehrere MDM-Hersteller, die Sie diesbezüglich beraten können. MDMs sind darüber hinaus sehr für die firmeneigene IT zu empfehlen, da sie viele weitere sicherheitsrelevante Richtlinien zur Durchsetzung auf den firmeneigenen mobilen Endgeräten anbieten und darüber hinaus die Softwareverteilung auf den Geräten der Mitarbeiter steuerbar machen.

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

IT-Sicherheit für die enaio® services im Unternehmensnetzwerk

enaio® mobile und enaio® webclient kommunizieren per HTTP/HTTPS mit enaio® gateway. Hinter enaio® gateway benötigen enaio® mobile und enaio® webclient mehrere Services:

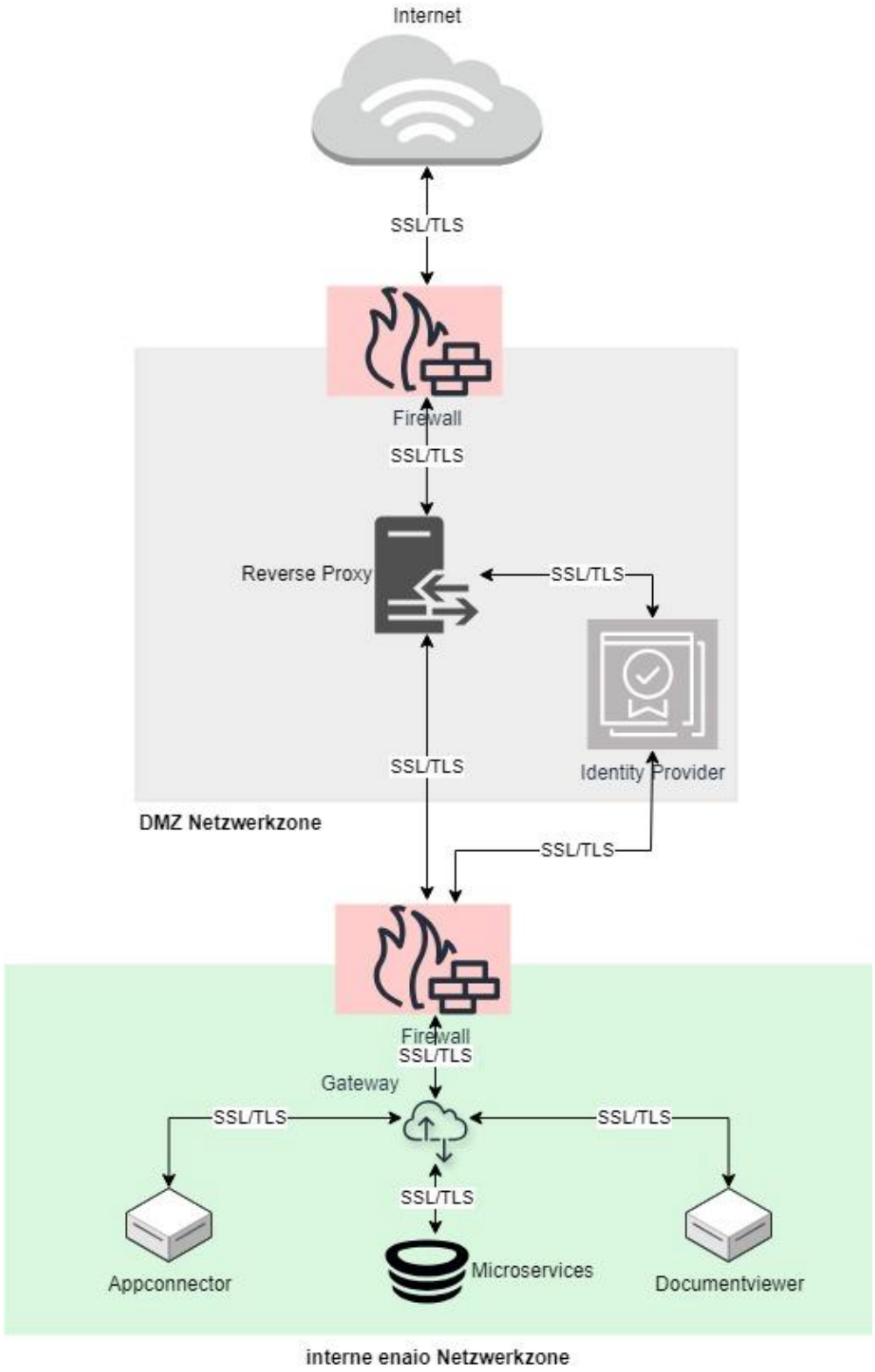
- Microservice enaio® webclient
- Microservice enaio® dms
- Microservice enaio® license
- Service enaio® appconnector
- Service enaio® documentviewer
- Microservice enaio® ems (optional)
- Microservice enaio® office365-dashlet (optional, ab Version 12.0 obsolet)
- Microservice enaio® office365-client (optional)

Um die Sicherheit der Daten durch enaio® gateway zu den einzelnen Services oder Microservices zu gewährleisten, sollten System und Netzwerk gemäß dem Stand der Technik und BSI IT-Grundschutz aufgebaut und konfiguriert sein. Um auf die jeweiligen Gegebenheiten in den unterschiedlichen Unternehmen einzugehen, wird empfohlen, einen Workshop hinsichtlich des Unternehmensnetzwerkes in Betracht zu ziehen.

In Produktivsystemen sollten die enaio® services durch eine demilitarisierte Zone (DMZ), die mit einer Firewall und einem Reverse Proxy ausgestattet ist, abgesichert werden. Dabei sollten die enaio® services und die DMZ in getrennten Netzwerksegmenten positioniert sein, um die Sicherheit zu erhöhen und das Risiko von unbefugten Zugriffen zu minimieren. Zusätzlich ist der Einsatz einer Ende-zu-Ende-Verschlüsselung zur Kommunikation zwischen allen Komponenten empfehlenswert.

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Beispielhafte schematische Netzwerkstruktur:



Einsatz von Mobile-Device-Management-Lösungen

enaio® mobile verwendet das Cordova-Plug-in cordova-plugin-emm-app-config der AppConfig Community (APC). Über dieses lassen sich standardisiert Mobile-Device-Management-Lösungen (MDM) einbinden. Als Administrator haben Sie darüber die Möglichkeit einerseits mobile Endgeräte in Ihrem Unternehmen in ihrer Verwendung zu beschränken und andererseits auch enaio® mobile ein Verbindungsprofil beim Starten mitzugeben.

Sollten Sie enaio® mobile in ein MDM-Tool einbinden, so zeigt Ihnen dieses konfigurative Felder für die Profilverwaltung an. Sie können die Serveradresse und den Benutzernamen sowie das Passwort für Ihr enaio®-System dort festlegen. Woher Sie diese Daten für das Profil laden, bleibt Ihnen überlassen. Sie können eine statische Quelle nehmen oder den Benutzer identifizieren und dynamisch seine Logindaten an enaio® mobile übermitteln. Auch können Sie festlegen, dass der Benutzer direkt angemeldet werden soll. In diesem Falle sieht der Benutzer den Profilmanager nicht und kann enaio® mobile direkt produktiv nutzen.

Während der Entwicklung haben wir mit IBM MaasS360 sowie VMWare Airwatch getestet. Es sollten jedoch auch alle anderen MDMs, deren Hersteller Mitglied der AppConfig Community ist, funktionieren.

Weitere Sicherheitsfunktionen für enaio® mobile und enaio® webclient

enaio® mobile, enaio® webclient und enaio® webclient als Desktop-Anwendung werden von OPTIMAL SYSTEMS ständig weiterentwickelt. Im Zuge der Weiterentwicklung nehmen wir gerne Ihre Wünsche entgegen und prüfen, gerne auch mit Ihnen zusammen, ob und wie diese in kommenden Versionen umgesetzt werden sollen. Wenden Sie sich dazu bitte an unser Produktmanagement pm@optimal-systems.de. Sollten Sie Fragen zur Sicherheit haben, so wenden Sie sich bitte an unseren Support unter support@optimal-systems.de.

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de

Verweise

[APL] Apple – Empfohlene Einstellungen für WLAN-Router und Zugangspunkte: https://support.apple.com/de-de/HT202068 (Abruf am 14.11.2024).

[KT] Projekt Keytar zur sicheren Speicherung der Profildaten unter enaio® webclient als Dektop-Anwendung:
https://github.com/atom/node-keytar (Abruf am 14.11.2024).

[MOZ] Gruppenrichtlinienkonfiguration zum Verbot der Entwicklertools in Mozilla Firefox:
https://github.com/mozilla/policy-templates/releases (Abruf am 14.11.2024).

[GC1] Gruppenrichtlinienkonfiguration zum Verbot der Entwicklertools in Google Chrome:
https://www.chromium.org/administrators/policy-templates (Abruf am 14.11.2024).

[GC2] Konfigurationsschlüssel zum Verbot der Entwicklertools in Google Chrom ohne Gruppenrichtlinien: http://dev.chromium.org/administrators/policy-list-3#DeveloperToolsAvailability (Abruf am 14.11.2024).

[MSE] Gruppenrichtlinienkonfiguration zum Verbot der Entwicklertools in Microsoft Edge:
https://www.tenforums.com/tutorials/106201-enable-disable-microsoft-edge-developer-tools-windows-10-a.html#option2 (Abruf am 14.11.2024).

[APC] AppConfigCommunity
https://www.appconfig.org (Abruf am 14.11.2024).

Cicerostraße 26 · 10709 Berlin
Tel.: +49 30 895708-0 · Fax: +49 30 895708-888
E-Mail: kontakt@optimal-systems.de
Internet: www.optimal-systems.de