BACnet/IT – Gebäudeautomation in Zeiten des Internets

Über die letzten Jahrzehnte hat Datenkommunikation in verschiedenen Anwendungsbereichen Einzug gehalten und zu jeweils eigenen Lösungen geführt. So haben die Gebäudeautomations-Ingenieure für ihre Zwecke verschiedene Datennetzwerkstandards entwickelt, die sich in der Industrie und in bestehenden Gebäuden etabliert haben. Zugleich hat sich aus verschiedenen Anwendungen wie der Datenverarbeitung und der Büroinformation die bekannte Internet-Technologie entwickelt. Diese ist mittlerweile so verbreitet, dass eine Sogwirkung einsetzt: Dank hoher Stückzahlen, Marktmacht, Bekanntheit usw. wird es immer interessanter die Internet-Standards anstelle eigener, anwendungsspezifischer zu nutzen. Wie man zwei etablierte Standards – BACnet aus der Gebäudeautomation und die IT-Netze der Gebäude-Nutzer – zusammenbringen kann, ohne dass sich eine der beiden Welten komplett der anderen unterwirft, haben wir in einem KTI-geförderten Projekt zur Konvergenz der Gebäudeautomation und IT-Welt untersucht.

Wolfgang Weck | wolfgang.weck@fhnw.ch

Digitalisierung, *Industrie 4.0* und *Internet of Things* (IoT) sind aktuell vielgebrauchte Schlagworte, mit denen Fortschritt aber auch Handlungs- und vor allem Forschungsbedarf konnotiert sind. Die Herausforderung dabei ist häufig der Brückenschlag zwischen der Internet-Technologie der Informatik und anderen Ingenieursdisziplinen. Das *Internet der Dinge* meint ja im Kern nichts anderes als in ihrer Art meist bereits existierende technische Geräte – oft Sensoren oder Aktoren – neu über das Kommunikationsmedium Internet so zu verbinden, dass sie direkt Daten und Kommandos austauschen können, aber auch für Datenanalysedienste oder optimierende Steuerungen erreichbar werden. Dafür stehen dann wiederum die ebenfalls aktuellen „Smart"-Schlagworte wie *Smart City*, *Smart Campus*, *Smart Grid* usw.

Vor diesem Hintergrund haben wir uns in einem kürzlich abgeschlossenen KTI-Projekt[1] zusammen mit der *Siemens Building Technologies Division* in Zug mit der Konvergenz von Technik und Methodik der Informatik mit solchen aus der Gebäudeautomation beschäftigt. Dabei ging es um den Betrieb grosser Gebäude u.a. mit komplexen Lüftungs-, Heizungs- und Klimatisierungsanlagen; nicht zu verwechseln mit den seit einiger Zeit auf den Markt drängen Produkten der Heim-Automation – also Anwendungen in Wohnungen und Einfamilienhäusern hauptsächlich zur Steuerung von Licht, Beschattung und Unterhaltungselektronik.

Hier berichten wir über diese Entwicklung hin zu einer gemeinsamen technischen Basis für die zwei etablierten Ingenieursdisziplinen Gebäudeautomation und Informatik. Im Sinne eines Brückenschlags zwischen diesen Disziplinen setzt dieser Bericht keine vertieften Kenntnisse voraus – weder in Gebäudeautomation noch in Informatik.

Kurze Geschichte digitaler Datenkommunikation

Die zuvor erwähnten neuen Produkte für Heimanwendungen mögen den Eindruck vermitteln, dass Digitalisierung und IoT für die Gebäudeautomation eine ganz neue Entwicklung seien. Dass dies nicht der Fall ist, zeigt Abbildung 1 in der die zeitliche Entwicklungsgeschichte zweier Datenkommunikations-Standards gegenübergestellt ist. Die Geschichte dieser beiden Datenkommunikations-Standards beginnt bereits Ende der 80er Jahre, nachdem die Entwicklung von Elektronik und Mikroprozessoren dafür gesorgt hatte, dass Werkzeuge zur Verarbeitung digitaler Daten und Signale grundsätzlich zur Verfügung standen.

Verschiedene Nutzergruppen machten sich daran, für ihre Zwecke nutzbare Anwendungen auf dieser Basis zu kreieren. So entstanden im Wesentlichen parallel und voneinander unabhängig Datenkommunikationsstandards z.B. in der Gebäudeautomation (im Beispiel *BACnet* [BACnet], seit 2003 ISO-Standard [BACISO]) und für die Verteilung elektronischer Dokumente zunächst in der

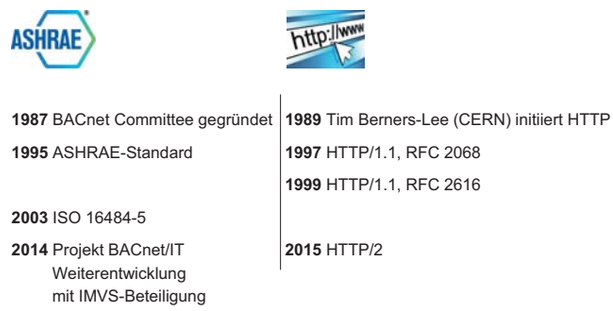

Abbildung 1: Zeitliche Entwicklung zweier digitaler Daten-Kommunikationsstandards im Vergleich: BACnet (Gebäudeautomation) und HTTP (Internet)

1 KTI-Projekt: Convergeence of Building Automation and IT World, KTI-Nr. 16841.1 PFES-ES

Wissenschaft und später in der Büro-Automation (sog. *IT-Netze*). Erst über die folgenden Jahre bis Dekaden entwickelte sich aus Letzterem rund um die Protokolle TCP, HTTP usw. sowie die dafür genutzten Kabel- und Funkstandards so viel Dynamik und Marktmacht, dass sich daraus ein praktisch flächendeckend vorhandener Service etablierte – das *Internet* aus Konsumentensicht. Den Zugang dazu betrachtet man heute vielfach als ähnlich selbstverständlich vorhanden wie den zu elektrischer Energie und trinkbarem Wasser.

Die Datenkommunikationsstandards der Automatisierung, wie das in Abbildung 1 gezeigte BACnet, umfassten zunächst separate Entwicklungen auf allen relevanten Ebenen von der Verkabelung bis zu den anwendungsspezifischen Datenformaten für Sensor-Messwerte und Steuerbefehle [Bus97]. Von der Dynamik der Entwicklungen der IT-Netze blieben sie mehr oder weniger unberührt. Einerseits bleiben bestehende Installationen in Gebäuden lange Zeit erhalten, da es aufwändig aber wenig nutzbringend wäre sie zu ersetzen. Andererseits gab es wenig Druck zu Veränderungen auf die Gebäudeautomation. Die bestehenden Standards konnten gut weiterentwickelt werden und teilweise auch auf industriell breit gefertigte Standardhardware übertragen werden. So sind die Kabelstandards neuerer BACnet-Anlagen identisch mit jenen der IT-Netze, die Datenübertragungsformate aber weiterhin unverändert, was die Integration neuer Geräte in bestehende Gebäude – z.B. beim Ersatz nach Defekt – möglich macht.

Die Digitalisierung hat sich also in der Gebäudeautomation und der Informatik parallel entwickelt. Dies ermöglichte auch, unterschiedliche Anforderungen zu unterstützen. Während es beim weltweit offenen Internet wichtig wurde, die eigenen lokalen IT-Netze (LAN oder Intranet) und Geräte vor Missbrauch durch Datenkommunikation von aussen zu sichern, war es für die physikalisch ohnehin von der Aussenwelt getrennten Gebäudeautomationsnetze vor allem wichtig, innerhalb des Gebäudes Daten direkt und schnell – das heisst innerhalb enger Zeitfenster – zu übertragen.

Die Dichte von Geräten wird dabei im Internet der Dinge sehr hoch. Betrachten wir dies exemplarisch am Beispiel des FHNW-Standorts Windisch, wo sich auch die Hochschule für Technik mit dem IMVS befindet. Hier wird zwar nicht BACnet verwendet, sondern mit KNX ein anderer Kommunikationsstandard der Gebäudeautomation [KNX]. Zwar ist KNX – wie BACnet – ein anwendungsbezogener Kommunikationsstandard und das Netz ist a priori nicht direkt mit dem Internet verbunden. Die Perspektive des IoT ist aber durchaus, das zu ändern. Es ist auch heute schon vergleichsweise einfach, einen kleinen preisgünstigen Computer (z.B. einen Raspberry Pi [RasPi]) gleichzeitig mit dem KNX-Bus und dem Internet zu verbinden und so einen offenen Zugang zur Gebäudeautomation zu schaffen.

Am KNX-Bus des Standorts Windisch der FHNW sind ca. 13 000 Sensoren und Aktoren angeschlossen. Zum Vergleich: Die Anzahl von Computern und sonstigen Geräten (Server, Drucker, etc.), die mit dem IT-Netz verbunden sind, beträgt an der FHNW in Windisch ca. 3500, also weniger als ein Drittel. Erst für alle Standorte der FHNW zusammengenommen kommt die Anzahl der Geräte im IT-Netz mit 12 000 in eine ähnliche Grössenordnung wie diejenige der „Dinge im Gebäudeautomationsnetz“ alleine in Windisch.

BACnet/IT – Konvergenz von Gebäudeautomation und Internet

Im Rahmen des oben erwähnten KTI-Projekts wurde auf Initiative des BACnet-Standardkomitees [BACnet] die Möglichkeiten zur Konvergenz der beiden parallelen Entwicklungen von IT-Netzen und BACnet untersucht. Kurzgefasst kann man das Ziel der Untersuchungen formulieren als „ein Netz für alles“. Das heisst, die Gebäudeautomation soll die ohnehin vorhandene Internet-Infrastruktur mitbenutzen: Nicht nur die gleichen, sondern dieselben Kabel und Services wie das IT-Netz für Büroautomation, Unterhaltungselektronik und das Internet.

Hinter diesem Ziel stecken zwei treibende Kräfte. Erstens hat sich die IT-Netzwerk-Technologie heute soweit ausgebreitet, dass entsprechende Installationen als vorhanden bzw. ohnehin notwendig vorausgesetzt werden können, analog zu denen für die Verteilung elektrischer Energie. Die nötigen Geräte samt der zum Betrieb relevanter Dienste nötigen Software werden industriell in grossen Stückzahlen preisgünstig gefertigt. Es ist also mit substanziellen Kostenreduktionen zu rechnen, wenn man bei neuen Gebäuden auf die Internet-Technologie setzen kann, anstatt eine komplette Parallelwelt aufzubauen.

Nur ein Netz anstelle von mehreren verschiedenen zu betreiben, macht es ausserdem leichter, die Datenkommunikation zuverlässiger zu machen. Investiert man in redundante Einrichtungen dieses einen Netzes, kommt das gleichzeitig allen Netznutzern zugute, kostet aber nur einmal.

Die zweite treibende Kraft ist die immer wichtigere und nutzbringendere Öffnung der Kommunikation von lokalen Anlagen, also der direkten Verbindung mit dem Internet. Sei es, dass man Cloud Services zur Erfassung grosser Datenmengen aus Gebäuden nutzen möchte, um sie zu analysieren und damit beispielsweise den Energieverbrauch eines Gebäudes optimieren zu können; sei es, dass man in die Steuerung von aussen eingreifen können möchte, um auch externe Informationsquellen wie Wetter- oder Energiepreisprognosen einfliessen zu lassen; sei es, dass man ein-

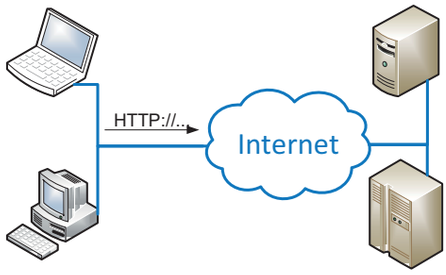

Abbildung 2: Typischer Verbindungsaufbau vom Client zum Server in IT-Netzen und im Internet

fach Daten und Steuerung über mehrere Gebäude hinweg verbinden möchte (*Smart Campus*).

Herausforderungen

In unserem Forschungsprojekt zeigten sich zwei Hauptherausforderungen, die wir hier vertieft betrachtet wollen. Diese Herausforderungen ergeben sich gerade aus den bisher unabhängigen Entwicklungen, Anforderungen und eben auch Kulturen der Gebäudeautomations-Ingenieure und der Informatiker. Kurzgefasst handelt es sich um folgende zwei Punkte:

1. Aus Sicherheitsgründen segmentierte IT-Netze behindern die Peer-to-Peer-Kommunikation im IoT. Die Herausforderung besteht darin, Verbindungen zwischen beliebigen IoT-Geräten möglich zu machen, was professionelles IT-Netzwerkmanagement unter Umständen gerade zu verhindern versucht.
2. Die für sich abgeschlossenen Netze der Gebäudeautomation sind nicht auf die Sicherheitsrisiken vorbereitet, die sich durch eine Verbindung mit dem Internet ergeben. Die Herausforderung besteht darin, jegliche Datenkommunikation eines Geräts nur mit erkennbar berechtigten Kommunikationspartnern zuzulassen, seien diese innerhalb oder ausserhalb des gleichen Gebäudes.

Sicherheit in IT-Netzen behindert IoT

IT-Netze werden meist für Client-Server-Architekturen konfiguriert (s. Abb. 2). Arbeitsplatz-Computer und mobile Geräte sind Clients, die Verbindungen zu Servern aufbauen, um deren Services zu nutzen. Im einfachsten Fall ist das eine Webseite, die vom Server als Antwort auf eine Anfrage geliefert und dann vom Browser-Programm auf dem Client dargestellt wird. Ein Client kann aber auch ein Programm sein, das einen Microservice beansprucht, der von einem Server angeboten wird. Dabei spielt es a priori keine Rolle, ob sich der Server innerhalb des gleichen IT-Netzes befindet oder ausserhalb, z.B. als sogenannter Cloud-Server bei einem entsprechenden Anbieter.

Die Kommunikation zwischen Clients und Servern ist üblicherweise verbindungsorientiert (basierend auf dem TCP-Protokoll). Nur über eine vom Client her eröffnete Verbindung kann ein Server seine Antwort schicken. Man möchte hingegen nicht, dass eine Verbindung hin zu einem der Clients eröffnet wird. Das wäre nicht konform mit dem Client-Server-Ansatz und möglicherweise der Versuch eines böswilligen Angriffs.

Das IT-Netz der FHNW beispielsweise verbindet eine grosse Menge an persönlichen Clients von Studierenden und Mitarbeitenden. Die meisten dieser Geräte werden auch ausserhalb der Hochschule mitgenommen und mit anderen Netzen verbunden. Da ist es schnell möglich, dass ein solches Gerät, das irgendwo anders einmal von Schadsoftware infiziert wurde, im FHNW-Netz versucht, andere Clients ebenfalls zu infizieren. Um das zu verunmöglichen, ist das logische Teilnetz, in dem sich alle diese Clients befinden, so konfiguriert, dass überhaupt keine Verbindungen zu diesen Teilnehmern eröffnet werden können – weder von einem anderen Gerät innerhalb desselben Teilnetzes, noch von ausserhalb.

Die IT-Server und Services eines Unternehmens lassen sich grob in zwei Gruppen teilen: Solche die man der Öffentlichkeit über das Internet anbieten möchte, wie Informationsseiten, Kundenportale, E-Mail-Empfang usw., und solche, die interne Dienste erbringen, die nur für Mitarbeitende des Unternehmens zur Verfügung stehen dürfen. Letzteres sind Datenbanken, Verwaltungssysteme und meist alles, was die eigentliche Geschäftstätigkeit ausmacht.

Der Datenkommunikationszugang von aussen wird deswegen meist zweistufig gesichert (s. Abb. 3). In einem oft als „demilitarisierte Zone“ (DMZ) bezeichneten Teilnetz sind Server erreichbar, deren Dienste im Internet öffentlich zugänglich sein sollen. Eine Firewall blockiert dabei Versuche, zu

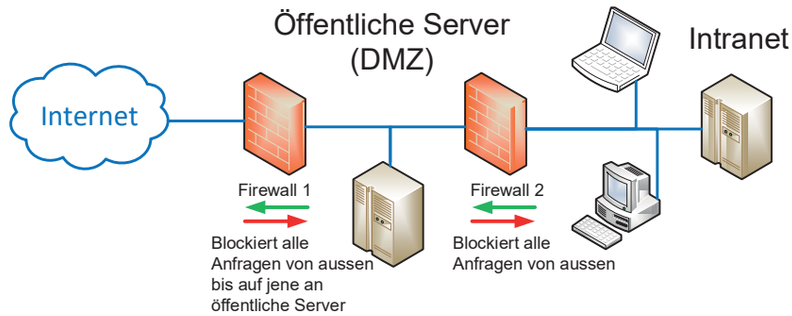

Abbildung 3: Zweistufige Sicherung des internen Netzwerks mit Firewalls

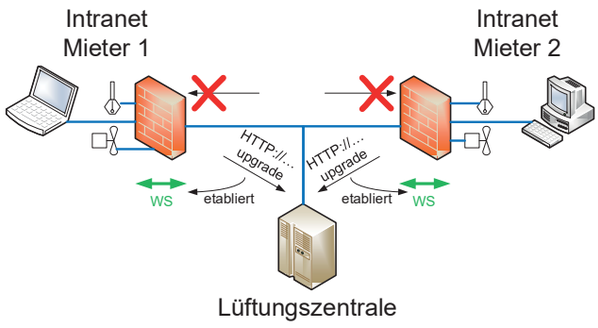

Abbildung 4: Bidirektionale Datenkommunikation auch in Mieternetze dank Websockets (WS)

diesen Maschinen unerwünschte Verbindungen zu eröffnen, also solche, die nicht direkt einem publizierten Service zugeordnet werden können. Oft wird daher nur das HTTP bzw. HTTPS-Protokoll zugelassen. Eine zweite Firewall verbietet jegliche Verbindungseröffnung zu den Geräten im internen Netzwerk. Diese dürfen – ganz gemäss Client-Server-Modell – Verbindungen aus diesem inneren Netz nach aussen öffnen, aber nicht umgekehrt.

Im Gegensatz zu Client-Server-Architekturen benutzt Gebäudeautomation traditionell ein verbindungsloses Kommunikationsmodell (wie das UDP-Protokoll). Anfragen nach Messwerten von Sensoren und Stellkommandos an Aktoren sowie Antworten und Bestätigungen werden einfach als einzelne Nachrichtenpakete im Netz verschickt. So kann beispielsweise ein Raumthermostat direkt einem Heizungsventil Kommandos zum Öffnen oder Schliessen zustellen. Die Firewalls der IT-Netze blockieren solche (UDP-)Nachrichten, die nicht über akzeptierte TCP-Verbindungen verschickt werden im Allgemeinen.

Stellen wir uns ein grosses Gebäude mit verschiedenen, sehr unterschiedlichen Mietern vor. Ein solcher Mieter ist vielleicht eine Bank, die hohe Sicherheitsstandards an ihre IT-Netze anlegt und Personal beschäftigt, das sich aktiv um deren Einhaltung kümmert, indem es entsprechende Einrichtungen wie Firewalls betreibt und aktiv konfiguriert. Ein anderer Mieter ist vielleicht eine kleine mittelständische Firma mit wenigen Mitarbeitenden, die alle einen Laptop mit Internet-Zugang benutzen wollen, aber keinerlei eigene Server vor Ort betreiben. (Entsprechende Services laufen vielleicht extern „in der Cloud“ bzw. bei einem Service Provider). Diese Firma hat dann vielleicht einen einfachen Router mit integrierter Firewall für den Internetzugang der Mitarbeiter-Laptops installiert. Verbindungsaufbauversuche von ausserhalb des (einfachen und nicht aktiv administrierten) Firmennetzes werden typischerweise so abgewiesen.

Schwierig wird es nun, wenn z.B. eine Lüftungszentrale im Gebäude Steuerbefehle an Ventile geben muss, die sich verteilt im Gebäude befinden und damit in den individuellen IT-Netzen der verschiedenen Mieter (vgl. Abb. 4). Der bei BACnet übliche Ansatz, einfach je ein Nachrichtenpaket von der Zentrale an die betroffenen Ventile zu verschicken, wird nicht mehr funktionieren, weil die Firewalls der Mieter im Wege stehen und diese Nachrichtenpakete nicht weiterleiten werden. Denkbar wäre natürlich, alle Mieter zu bitten bzw. zu verpflichten, entsprechende Sonderregeln für ihre Firewalls und Router zu konfigurieren. Die ICT-Spezialisten der Bank aus unseren obigen Beispielszenarien wären wohl grundsätzlich in der Lage dazu. Allerdings widerspricht ein solcher Wunsch meist der internen Policy der Firma. Ausserdem kann durch die schiere Anzahl an Gebäudeautomationsgeräten die Umsetzung aufwändig werden. Bei der mittelständischen Firma wiederum gibt es möglicherweise keine Person, die sich mit der IT-Technik genügend gut auskennt, um den (einfachen) Router entsprechend zu konfigurieren.

Logisches Netz aus Websocket-Verbindungen

Es wäre nun wenig hilfreich die Erfahrungen und Methoden samt existierender Programme der Gebäudeautomations-Ingenieure auf den Client-Server-Stil der Informatik zu zwingen, oder – umgekehrt – von IT-Netzen grundsätzliche die Durchlässigkeit von einzelnen Nachrichtenpaketen zu verlangen, wie sie die Gebäudeautomation verwendet. Vielmehr braucht es einen gemeinsamen Weg, mit dem einerseits existierende Konzepte der Gebäudeautomation weiterhin genutzt werden können, andererseits IT-Netze nicht speziell konfiguriert werden müssen.

Hier kommt ein Werkzeug zum Zug, das die Informatik ursprünglich entwickelt hat, um manche Grenzen der Client-Server-Architektur zu überwinden. In der ursprünglichen einfachen Form dieser Architektur muss nämlich jeder Client, der z.B. aktuelle Daten von einem Server benötigt, den Server immer wieder neu nach Veränderungen der Daten anfragen. Reaktionsschneller ist es aber, wenn der Server den Client von sich aus über solche Änderungen informieren kann. Dazu müssen Client und Server gemeinsam die am Anfang der Kommunikation etablierte Netzverbindung offen behalten, damit so der Server immer wieder neue Daten an den Client schicken kann. Dafür hat sich unter dem Namen *Websocket* ein entsprechender Standard etabliert [FM11].

Websockets sind (TCP-)Verbindungen, die sich mittels üblicher HTTP-Anfragen etablieren und nachher nahezu uneingeschränkt in beiden Richtungen nutzen lassen. So können sie auch in Netzen mit Firewalls benutzt werden, weil die Eröffnung dem Client-Server-Ansatz entspricht: Die Verbindung wird aus einem internen Netz heraus wie von einem Client bestellt.

Abbildung 4 zeigt, wie eine Lüftungszentrale des Gebäudes Ventile steuern kann, die über die internen IT-Netze der Mieter angeschlossen sind: Jedes Ventil verbindet sich einmalig mit einem Server bei der Zentrale und hält diese Verbindung offen – als Websocket. So kann die Zentrale es jederzeit für Kommandos erreichen. Wird die Verbindung aus irgendwelchen Gründen einmal unterbrochen, müssen die Kommunikationspartner dies feststellen und dann kann das Ventil eine neue Verbindung öffnen, um die bisherige zu ersetzen.

In einem Gebäude mit BACnet über IT-Netze verbinden sich also zunächst die einzelnen BACnet-Geräte untereinander indem sie ein Netz von Websockets etablieren über das sie dann im Bedarfsfall ihre Nachrichten nach bewährtem BACnet-Schema austauschen. Dass die Websockets nicht erst geöffnet werden, wenn konkret eine Nachricht übertragen werden muss dient auch dazu, Zeitverluste beim Nachrichtenversand zu vermeiden.

Dieses logische Netz aus Websocket-Verbindungen kann nach Bedarf aus einer Reihe direkter Verbindungen zwischen verschiedenen BACnet Geräten bestehen, aber auch als Stern mit einem zentralen Hub oder Broker organisiert werden. Direkte Verbindungen sind vor allem bei sehr zeitkritischen Anwendungen von Feuermelder bis Lichtschalter nützlich. Andererseits vereinfacht eine Sterntopologie die Verwaltung der Verbindungen. Möglich sind beide Ansätze.

Security – Datenkommunikation auf berechtigte Partner einschränken

Die Verwendung von Websockets unterläuft nun – zunächst ja beabsichtigt – die Sicherheitsmechanismen der IT-Netze. Gleichzeitig können sie Geräte der Gebäudeautomation mit dem öffentlichen Internet verbinden. Diese Mischung trägt grosses Gefahrenpotenzial in sich, denn es ist nun denkbar, dass irgendjemand von einem beliebigen Ort der Welt aus mit einzelnen Geräten eines Gebäudes Kontakt aufnehmen und Messwerte lesen oder Steuerkommandos geben kann.

Um dies zu verhindern, müssen die folgenden zwei Hauptanforderungen an die Sicherheit eines Gebäudeautomations-Netzes gestellt werden:

1. Unberechtigten darf es nicht möglich sein, Datenverkehr mitzulesen. Daraus liessen sich z.B. Erkenntnisse gewinnen, wann Räume leer stehen, um Einbrüche zu planen.
2. Geräte müssen erkennen können, ob Kommandos, die sie erhalten, aus einer Quelle stammen, die zur gleichen Anlage gehört.

Das heisst also, die Datenübertragung muss verschlüsselt werden (Kryptographie) und die Identität des Absenders einer Nachricht muss überprüfbar sein (Authentifizierung).

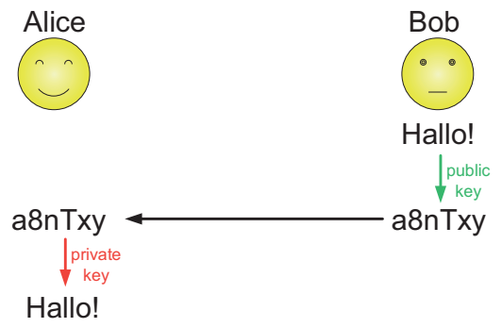

Abbildung 5: Verschlüsselte Datenübertragung beliebiger Sender (hier Bob), die nur mit privatem Schlüssel entschlüsselt werden kann

Public Key Cryptography

In der IT-Welt etablierte Lösungen für die beiden zuvor genannten Aufgaben beruhen auf asymmetrischer Public Key Cryptography. Diese arbeitet mit Paaren von kryptographischen Schlüsseln – z.B. S_1 und S_2. Jeder solcher Schlüssel ist eine mathematische Funktion, die aus einer Folge von Bytes eine andere Byte-Folge erzeugt. Schlüssel S_1 angewandt auf eine Nachricht N – also $S_1(N)$ – liefert eine verschlüsselte Form der Nachricht N.

Speziell an asymmetrischer Kryptographie ist, dass man eine Nachricht (bzw. Byte-Folge) die mit einem der beiden Schlüssel S_1 oder S_2 verschlüsselt worden ist, nur mit dem jeweils anderen Schlüssel wieder entschlüsseln kann. Es gilt also:

$S_2(S_1(N)) = N$ bzw. $S_1(S_2(N)) = N$.

Wer nur einen der beiden Schlüssel kennt, kann zwar verschlüsseln aber die gerade verschlüsselte Nachricht nicht wieder entschlüsseln.

In der Anwendung erklärt man nun einen der beiden Schlüssel als öffentlich (oder eben *public*) und den anderen als *privat*. Der private Schlüssel darf nur auf einem einzigen Gerät bekannt sein und muss dort geheim gehalten werden. Er wird also nie über das Netz verschickt. Den öffentlichen Schlüssel hingegen darf jeder und jedes beliebige Gerät kennen.

Zwei Geräte, die verschlüsselt miteinander kommunizieren sollen, können also einfach zu Beginn ganz offen ihre jeweiligen öffentlichen Schlüssel austauschen. Dieser Austausch darf auch von böswilligen Gegnern mitgehört werden, denn mit diesen Schlüsseln alleine ist keine Entschlüsselung möglich. Ist dies einmal geschehen, kann jedes Gerät den öffentlichen Schlüssel des Kommunikationspartners verwenden, um diesem Nachrichten so verschlüsselt zu schicken, dass nur er sie wieder entschlüsseln kann (s. Abb. 5).

Die Kosten dieser Public Key Cryptography fallen vor allem in Form von Rechenaufwand beim Ver- und Entschlüsseln an. Dieser Rechenaufwand ist höher, als bei einfacheren symmetrischen Verschlüsselungsverfahren, was gerade bei sehr Ressourcen-beschränkten Geräten im IoT-Bereich problematisch sein kann.

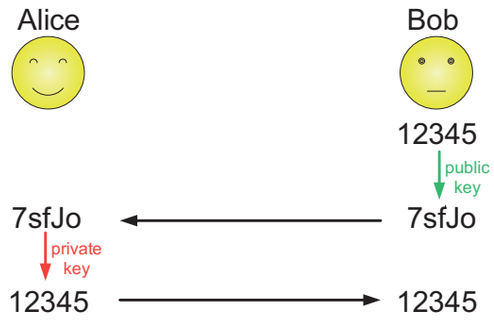

Abbildung 6: Authentifizierung als Besitzer des privaten Schlüssels zu einem öffentlichen Schlüssel mittels «Challenge Response»-Verfahrens

Symmetrische Verschlüsselung beruht auf je einem Schlüssel, den sowohl Sender als auch Empfänger einer Nachricht kennen. Treten zwei Geräte in Kontakt, müssen sie Information austauschen, die ausreicht, um diesen Schlüssel auf beiden Seiten identisch zu berechnen. Diese Information könnte aber auch ein Angreifer abhören, der dann ebenfalls den Schlüssel kennen und anschliessend abgehörte verschlüsselte Nachrichten entschlüsseln könnte.

Für verbindungsorientierte Kommunikation wie für Client-Server-Architekturen verwendet man in der Informatik eine Kombination dieser beiden Ansätze. Zunächst wird Public Key Cryptography benutzt, um die Möglichkeit zu schaffen, überhaupt sicher verschlüsselt zu kommunizieren. Dieses aufwändige Verfahren dient somit dazu – abhörsicher – einen symmetrischen Schlüssel zu vereinbaren, der anschliessend – recheneffizient – für den eigentlichen Austausch grösserer Datenmengen verwendet wird.

Ein dafür geeignetes Ablaufprotokoll ist unter Namen wie *TLS* oder *Secure Websockets* standardisiert und es existieren Implementierungen in Form von Bibliotheken, die in eigene Programme eingebunden werden können.

Authentifizierung mit X.509-Zertifikaten

Public Key Cryptography lässt sich auch als Werkzeug nutzen, um das zweite Sicherheitsproblem zu lösen. Es nutzt ja nichts, verschlüsselt zu kommunizieren, wenn ein Teilnehmer unbeabsichtigt und unwissentlich seine Nachrichten einem falschen Empfänger schickt und zur Verschlüsselung dessen öffentlichen Schlüssel benutzt. Der Sender sollte daher sicher sein, auch den richtigen Empfänger zu erreichen.

Der öffentliche Schlüssel wird dazu als Teil eines *Zertifikats* übermittelt, das die Funktion eines *Ausweises* hat. Analog zum Reisepass muss ein solches Zertifikat zwei Beweise möglich machen: Zum Ersten muss überprüfbar sein, dass der Ausweis für denjenigen ausgestellt ist, der ihn präsentiert (entsprechend dem Passfoto oder biometrischen Daten). Zum Zweiten muss geprüft werden können, dass die Angaben im Ausweis (z.B. Name oder Zugehörigkeit zu einer bestimmten Gemeinschaft) korrekt sind und der Ausweis von vertrauenswürdiger Stelle ausgestellt wurde.

Um zu überprüfen, ob ein Zertifikat tatsächlich vom Kommunikationspartner am anderen Ende einer Verbindung stammt und nicht einfach von diesem weitergeleitet worden ist, wird wieder der öffentliche Schlüssel benutzt. Allerdings geht es diesmal nicht darum, Daten abhörsicher zu übertragen, sondern zu überprüfen, ob das Gerät am anderen Ende einer Kommunikationsverbindung das Gegenstück zum öffentlichen Schlüssel, den privaten Schlüssel kennt. Wie Abbildung 6 zeigt, lässt sich dies sehr einfach nachprüfen: Man schickt eine beliebige zufällig gewählte Nachricht mit dem öffentlichen Schlüssel verschlüsselt an den Kommunikationspartner und lässt sich von diesem die entschlüsselte Nachricht zurückschicken. Die retournierte Nachricht ist nur identisch mit der ursprünglichen, wenn der Kommunikationspartner tatsächlich über den privaten Schlüssel verfügt, der zum verwendeten öffentlichen Schlüssel gehört.

Ein Zertifikat enthält eine Reihe von Angaben über den Kommunikationspartner. Beispielsweise seine Adresse, einen Namen und – für unsere BACnet-Anwendung – die Information, zu einer bestimmten Anlage zu gehören. Letztere ist ausschlaggebend, denn die Sicherheit von BACnet/IT baut darauf auf, dass nur Geräte derselben Anlage miteinander Verbindungen unterhalten dürfen.

Offensichtlich wäre es für einen Angreifer ein Leichtes, ein Schlüsselpaar (privater und öffentlicher Schlüssel) zu erzeugen und damit ein Zertifikat zu generieren, das aussagt, dass der Inhaber zu einer Anlage nach Wahl gehört. Wie sich aber niemand selbst einen anerkannten Reisepass ausstellen kann, sondern dafür einen vertrauenswürdigen Herausgeber benötigt, müssen auch Zertifikate von einer sogenannten *Certificate Authority* herausgegeben bzw. beglaubigt werden.

Diese Beglaubigung geschieht durch eine sogenannte *Signatur* des Zertifikats. Abbildung 7 zeigt das Schema dazu. Mit einem fixen Algorithmus (es gibt mehrere mögliche und das Zertifikat enthält den Namen des zu benutzenden Algorithmus) wird aus dem zu signierenden Zertifikat eine Art Prüfsumme (ein sogenannter Hashwert) berechnet. Das ist im Prinzip eine grosse Zahl, die man zu einem gegebenen Zertifikat rasch berechnen kann, für die es aber extrem schwierig ist, ein anderes nützliches Zertifikat zu erzeugen, das zur gleichen Prüfsumme führen würde. Diese Prüfsumme wird von der *Certificate Authority* mit deren privatem Schlüssel verschlüsselt und dem Zertifikat als „Signatur“ hinzugefügt.

Um die Echtheit eines solchen Zertifikats zu prüfen, kann jeder, der den öffentlichen Schlüs-

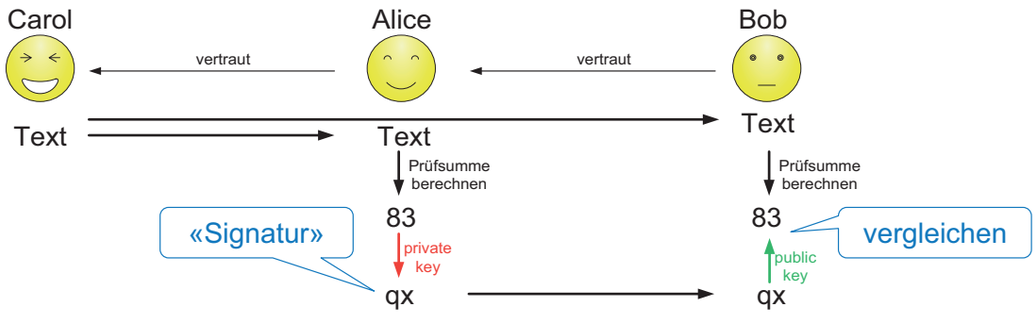

Abbildung 7: Signieren eines X.509-Zertifiakts durch Alice als vertrauenswürdige Certificate Authority: Durch Hinzufügen einer mit dem privaten Schlüssel verschlüsselten «Prüfsumme» zu einem Text von Carol gibt Alice allen ihr vertrauenden Partnern bekannt, dass sie diesem Text von Carol ebenfalls vertrauen können.

sel der *Certificate Authority* kennt, die Signatur damit entschlüsseln und mit der selbst zum Zertifikat berechneten Prüfsumme vergleichen. Sind die beiden Werte gleich, kann davon ausgegangen werden, dass das vorliegende Zertifikat mit genau diesem Inhalt von der *Certificate Authority* für vertrauenswürdig beurteilt worden ist. Mit diesem Prozess kann also Vertrauen weitergegeben werden.

Die hier erklärten Verfahren sind Grundlagentechniken der Informatik. Glücklicherweise sind die Informatiker aber über die reinen Konzepte hinausgegangen und haben z.B. Formate für Zertifikate (sogenannte X.509-Zertifikate) sowie Protokolle für den Austausch und die Überprüfung sowie für verschlüsselte Kommunikation (TLS) vereinheitlicht standardisiert und in Programmbibliotheken implementiert, die man in eigene Produkte einbinden kann.

Ergebnisse

In dem hier beschriebenen Forschungsprojekt haben wir gezeigt, dass sich aus etablierten Werkzeugen der Informatik eine sichere und genügend effiziente Datenkommunikationsgrundlage schaffen lässt, auf der etablierte Standards und Anwendungen der Gebäudeautomation aufgesetzt werden können. Aus *Secure Websockets* (der standardmässigen Kombination aus *Websockets* mit TLS) lässt sich ein logisches Netz von Verbindungen auch durch Firewalls der IT-Netze hindurch errichten.

Vorhandene BACnet-Implementierungen binden dies als neues Transportmedium (in der BACnet-Welt als *Data Link* bezeichnet) ein. Die Applikationen der Gebäudeautomation benutzen unverändert die Schnittstelle ihres BACnet-Stacks.

Im Jahr 2018 wird das BACnet-Standardisierungs-Komitee der ASHRAE einen neuen Standard-Entwurf *BACnet/SC* öffentlich zur Review aufgelegen [ASHRAE]. *SC* steht dabei für *secure connect* und beinhaltet die in diesem Artikel beschriebene sichere Datenkommunikation über *Secure Websockets*.

An diesem Konzept durften wir in den vergangenen Jahren mitarbeiten, zusammen mit Ingenieuren der *Siemens Building Technologies Devision*. Sowohl von Siemens als auch von uns ist der entstehende Standard implementiert worden, um zu zeigen, dass zwei unabhängig entstehende Implementierungen zueinander kompatibel sind. Wir haben von diesen beiden Prototypen auch nicht-funktionale Qualitätsmerkmale untersucht, allen voran das Laufzeitverhalten auch bei grösserem Nachrichtenvolumen. Die Ergebnisse dieser Arbeiten sind vom Standardisierungskomitee direkt genutzt worden, um Teillösungen zu bestätigen oder zu überarbeiten.

Wir haben aus dem Projekt Erkenntnisse gewinnen können, die sich gut auf generelle IoT-Themen auch jenseits von Gebäudeautomation verallgemeinern lassen. Je nach konkreten Anforderungsprofilen können die für BACnet erarbeiteten Lösungsstrategien direkt oder in angepasster Form übernommen werden.

Referenzen

[ASHRAE] American Society of Heating, Refrigerating and Air-Conditioning Engineers: https://www.ashrae.org/

[BACnet] BACnet – A Data Communication Protocol for Building Automation and Control Networks, Official Website of ASHRAE SSPC 135: http://www.bacnet.org/

[BACISO] Der ISO 16484-5:2017-Standard in aktueller Form: https://www.iso.org/standard/71935.html

[Bus97] Steven T. Bushby: "BACnetTM – A standard communication infrastructure for intelligent buildings", Published in Automation in Construction, Vol. 6 No. 5-6, 1997, pp. 529-540. http://www.bacnet.org/Bibliography/AIC-97/AIC1997.htm

[FM11] Fette, I. and A. Melnikov, "The WebSocket Protocol", RFC 6455, DOI 10.17487/RFC6455, December 2011. http://www.rfc-editor.org/info/rfc6455

[KNX] KNX Association: https://www.knx.org

[RasPi] Raspberry Pi: https://www.raspberrypi.org