Sehr geehrte Professoren, liebe Mit-Studenten, ich begrüsse sie zur präsentation meiner Diplomarbeit mit dem Titel 'highly availabe KNX networks'

zum aufbau der präsentation: ich werde zuerst kurz auf das thema gebäudeautomation und auf den KNX standard selbst eingehen.
Danach werde ich erläutern welche defizite die ursprüngliche KNX spezifikation in punkto sicherheitsaspekte aufweist, und auch die bekanntesten erweiterungen und deren ziele vorstellen.
Darauffolgend werde ich zusammenfassen warum diese erweiterungen nur bedingt ausreichen – dies ist quasi die motivitation für meine diplomarbeit, und welche ziele diese arbeit verfolgt.
abschliessend werde ich erklären wie genau mein proposal versucht diese ziele umzusetzen.

Gebäudeautomation beschäftigt sich im ursprünglichen sinn mit der steuereung und regelung von sogenannten HVAC-applikationen: HVAC steht dabei für 'heating, ventilation und air-conditioning'. Das hauptziel ist eine effiziente und zentrale verwaltung von raumparametern, und daraus folgend niedrigere verwaltungs- und energiekosten. ebenfalls nicht zu vergessen ist dass sich der raumkomfort in z.b. grossraumbüros direkt auf die produktivität der darin beschäftigten mitarbeiter auswirkt.

Wenn nun ein büro- oder fabriksgebäude mit der infrastruktur ausgestatttet ist um diese HVAC applikation steuern zu können liegt es sehr nahe diese für zusätzliche anwendungen einzusetzen – potentielle anwendungsfälle gibt es reichlich: diese reichen von raumbeleuchtungen und aufzugssystemen zu zutrittssystemen, alarmanlagen und sprinkler-und feuerschutzanlagen. Es ist allerdings schnell ersichtlich dass z.b. eine alarmanlage höhere anforderungen an die zuverlässlichkeit und integrität des zugrunde liegenden kommunikationssystems stellt als das schalten von beleuchtungselementen.. beim design von gebäudeautomationssystemene – im speziellen KNX – waren dies anforderungen welche ignoriert wurden. Dies hatte folgende gründe: erstens wurden die zu steuernden anwendungen als unkritisch eingestuft. Damit ist gemeint dass ein ausfall des systemes oder auch ein mutwilliger angriff gegen das system von aussen nur vernachlässigbare konsequenzen nach sich ziehen würde.

Ein zweites bei kabelgebunden kommunikationssystemen benutztes argument ist dass ein angreifer zuerst Pyhsikalischen zugriff auf das medium selbst erlangen musste um einen angriff ausführen zu können. Und Letztlich fehlte oft ganz einfach die rechenleistung in den verwendeten embedded geräten zum umsetzen der notwendigen kryptografischen massnahmen.

Die rechenleistung für nahtlose verschlüsselung steht allerdings heutzutage auch auf kleineren platformen zur verfügung. Weiters stellt sich heraus dass auch die beiden ersten argumente nicht auf realistischen annahmen beruhten. erstens ist es schlichtweg unmöglich bei einem automationssystem, welches über jahre wenn nicht jahrzehnte operabel sein soll, auch nur kurzfristigen zugriff auf das kabelmedium komplett auszuschliessen. Die verwendung von kabellosen medien führt dieses argument grundsätzlich ad absurdum. Zweitens ist die annahme dass die ursprünglichen HVAC anwendungen nicht-sicherheitskritisch sind zu optimistisch – simple akte des vandalismus können durchaus spürbare ausfälle nach sich ziehen.

Diese überlegungen führten letztlich zu einem paradigmenwechsel in punkto sicherheitsaspekten, welche nun immer weniger als vernachlässigbar angesehen wird.
Sicherheitsaspekte beim design von systeme zur heim- und gebäudeautomation werden dementsprechend zunehmend berücksichtigt bzw werden die systeme nachträglich erweitert. Bei KNX exisitieren mehrere sicherheitserweiterungen, welche etwas später kurz vorgestellt werden.

Ich möchte nun kurz definieren was hier unter 'sicherheitsaspekten' gemeint ist – im allgemeinen ist

damit das erreichen von 'informationssicherheit' gemeint. Diese besteht aus drei säulen: confidentiality, integrity und availability, zusammengefasst zum CIA-triad. Confidentiality soll sensitive information vor unberechtigen entities schützen. Integrity stellt sicher dass schützenswerte daten nicht von dritten böswillig modifiziert werden können. Abschliessend stellt availability sicher dass ein system zuverässlich arbeitet und die definierten services den nutzern dieser services zur verfügung stehen. Hierbei ist es allerdings wichtig, alle 3 säulen zu berücksichtigen – ein design welches nur teile des CIA triads berücksichtigt wird sehr wahrscheinlich zu einem unsicheren system führen.

---

damit komme ich zum zweiten teil meines vortrages, nämlich dem aufbau des KNX standards. KNX enstand aus drei verschiedenen standards, nämlich EIB, dem 'europaean installation bus'; EHS, 'european home systems protokoll, und batibus, und wurde von der KNX association definiert. Es handelt sich um einen offenen standard, die vollständige spezifikation kann gegen eine gebühr bezogen werden, die verwendung danach ist frei. Zusätzlich ist eine zertifizierung möglich, allerdings nicht zwingend vorgeschrieben (KNX logo, ETS eintrag). Der standard orientiert sich am ISO OSI schichtenmodell, oberstes ziel ist grösstmögliche interoperabilität und damit hersteller-unabhängigkeit. Sensoren und aktoren werden mittels sogenannter 'group adresses' logisch zusammengefasst, womit die gewünschte semantik der kontrollapplikation erreicht wird.

Ich werde die wichtigsten protokollparameter – soweit sie für meine arbeit wichtig sind – anhand des OSI modells erläutern:

Ganz oben im schichtenmodell stellt der applikation layer interfaces für die zugreifende applikation zur verfügung, z.b. das lesen oder schreiben von group values oder zur verteilung von netzwerkparamentern.

Der presentation layer, welcher eine system-unabhängige darstellung der ausgetauschten nachrichten garantieren soll, ist transparent. Um die angestrebte interoperabilität zu erreichen definiert KNX sogenannte 'data points': diese stellen die eigentlichen kontrollvariablen dar und werden zu 'data point types' mit einer definierten struktur zusammengefasst, wodurch eine einheitliche darstellung erreicht wird und der praesentation layer überflüssig wird.

Layer 5 oder 'session layer' ist ebenfalls transparent. Der Layer 4 oder transport layer stellt services für die end-zu-end verbindung zwischen quell- und zielhosts zur verfügung, wobei bei KNX unbestätigte als auch bestätigte übertragungsmodi möglich sind – für letzteren wird ein simpler handshake-mechanismus, ähnlich zu dem von TCP bekannten, verwendet.

Der wichtigste parameter des network layers ist die verwendete adresse. KNX verwendet 2 verschiedene arten von adressen: 'individual adresses' müssen eindeutig sein und adressieren ein bestimmtes gerät im netzwerk. 'group adresses' adressieren dagegen eine bestimmte gruppenvariable.

Layer 2 steuert den buszugriff und definiert verschiedene frameformate, wobei hier nur standard- und extended frames erwähnt seien.

Auf physikalischer ebene wurden verschiedene übertragungsarten definiert: erstens 'twisted pair' zur übertragung der steuerdaten über dezidierte kabelverbindungen. Powerline verwendet ein bereits vorhandenes stromnetz. Weiters besteht die möglichkeit der kabellosen kommunikation mittels 'radio frequency' sowie der austausch von daten mit IP hosts mittels KNX/IP, wobei letzteres hauptsächlich für backbone-traffic verwendet wird.

Wie bereits angemerkt definiert der ursprüngliche KNX standard keine ausreichenden sicherheitsmassnahmen. Dieses defizit wurde nachträglich durch die definition verschiedener extensions entschärft. Namentlich sind dies folgende:

- KNX data security arbeitet analog zu TLS und ermöglicht die end-zu-end-verschlüsselung zwischen KNX geräten, unabhängig vom verwendeten kommunikationsmedium
- EIBsec stellt ebenfalls verschlüsselte datenservices zur verfügung, allerdings mithilfe von dezidierten key-servern.
- KNX IP security fokussiert auf die verschlüsselung des backbone traffics und verwendet ebenfalls einen eigenen sicherheitslayer und ist damit ebenfalls mit TLS vergleichbar

Wenn also bereits erweiterungen exisitieren welche informationssicherheit erreichen sollen – wozu dann eine zusätzliche? Der sinn dieser arbeitet ist die erhöhung der verfügbarkeit für kabelgebundene KNX netzwerke, wobei sowohl schutz gegenüber transienten hardwarefehlern gegeben sein soll, als auch bei vorhandensein von böswilligen angreifern. Dies macht es notwendig das gesamte CIA triad zu berücksichtigen – die oben genannten erweiterungen fokussieren auf die komponenten confidentiality und integrity. Ein angreifer kann ein netzwerk dennoch durch kurzschliessen der kommunikationsleitungen lahmlegen. Auf die selbe fatale art und weise wirkt sich ein simpler, unvorhersehbarer kabelbruch aus.

Diese arbeit hat daher als ziel, alle 3 säulen des CIA triads in einem konzept zu vereinigen. Dazu ist es notwendig, das kommunikationsnetzwerk gegen abhören und modifikation zu schützen, aber auch durch redundanz sicherzustellen dass zumindest ein kommunikationsweg verfügbar ist.

Die vorgeschlagene lösung erreicht höhere verfügbarkeit durch duplizieren der kommunikationswege, mitsamt der entsprechend notwendigen sende- und empfangshardware, wobei alle nachrichten die über die redundanten kommunikationsleitungen gesendet werden kryptografisch abgesichert werden. Damit unterteilt die arbeit ein KNX netzwerk in einen sicheren und einen unsicheren teil. Diese 2 teile werden von sogenannten 'security gateways' minteinander verbunden. Jedes dieser gateways besitzt 3 interfaces: ein interface wird mit einem standard-KNX netzwerk verbunden und stellt damit die verbindung ins ungesicherte netzwerk dar. Die anderen 2 interfaces bilden die redundant ausgelegten und gesicherten kommunikationsleitungen. Das kleinstmögliche derartige netzwerk besteht demnach aus 2 securitygateways welche redundant über 2 netzwerke verbunden sind, wobei jedes dieser gateways zusätzlich mittels dem dritten interface mit einem normalen, ungesicherten netzwerk verbunden ist.

Beim design des protokolls wurde wert darauf gelegt dass grösstmögliche interoperabilität erreicht wird, das bedeuted dass gewöhnliche knx geräte, verbunden durch ein gesichertes netzwerk, transparent kommunizieren können. Diese Transparenz kann allerdings nur teilweise erreicht werden – gewisse in KNX definiert übertragungsarten können aufgrund der strikt definierten timingvorgaben nicht umgesetzt werden.
Ebenso wird flexibilität sichergestellt, d.h. So ist es möglich, zusätzliche standard-KNX knoten zu dem netzwerk hinzuzufügen. Dies ist sowohl hinter bereits existierenden als auch hinter neu installierten gateways möglich.

Als nächstes werde ich kurz den aufbau des vorgeschlagenen protokolls erläutern. Grob gesagt besteht dieses aus 3 unterschiedlichen phasen:

1. zuerst durchläuft jedes gateway nach dem einschalten die syncronization phase
2. vor jedem tatsächlichen datentransfer zwischen zwei gateways erfolgt die discovery phase
3. passiert der tatsächliche datenaustausch

die syncronization phase hat den zweck einen netzwerk weiten, globalen zählerwert zwischen bereits laufenden geräten und einem neu hinzukommenden gerät auszutasuchen. Der Zählerwert dient in weiterer Folge als initialisation vector verwendet und hilft so, replay attacken zu verhindern. Die syncronissierungsantworten selbst werden per MAC authentifiziert. Da hier noch nicht der globale Zählerwert bekannt ist wird die aktuelle zeit verwendet um replay-attacken zu verhindern. das bedeutet dass die systemzeit der gateways zumindest grob, also im sekundenbereich, abgeglichen sein muss.

Möchte nun ein gewöhnliches knx gerät eine nachricht an ein anderes gerät schicken, wird diese nachricht zuerst von seinem verantwortlichen gateway empfangen und gespeichert. Dieses gateway 'A' muss nun feststellen hinter welche anderen gateways die entsprechende zieladresse zu erreichen ist. Dazu werden zu allererst discovery-messages über die redundanten leitungen ausgeschickt – diese nachrichten sind symmetrisch verschlüsselt, per 'message authentication code' geschützt und enthalten zusätzliche den globalen zähler, um replay-attacken zu verhindern. Jedes empfangende gateway prüft nun ob es für diese zieladresse zuständig ist. Ist dies der fall sendet es redundant entsprechende discovery-response nachrichten zurück an den requester, wobei diese antworten analog zu den requests abgesichert sind. Zusätzlich werden dabei ebenfallls gleich die im nächsten schritt verwendeten diffie – hellman paramter mit ausgetauscht. Gateway 'A' kann nun den ursprünglich empfangen KNX frame für beide leitungen getrennt symmetrisch verschlüsseln, verpackt diese verschlüsselten frames in zwei unabhängige frames und versendet diese über beide gesicherten interfaces. Integritiät wird wiederum über einen message authentication code sichergestellt. Zusätzlich wird ein für quell – und – zieladresse dezidierter zähler gesendet, welcher wiederum gegen replay attacken schützt und zusätzlich das verwerfen der duplikate auf empfangsseite ermöglicht.

Das empfangende gateway erhält nun im idealfall 2 unterschiedliche frames, welche beide das gleiche, urspünglich von gateway 'A' empfange standard KNX frame enthalten. Nachdem per MAC geprüft wurde dass die frames nicht modifiziert wurden kann eines der 2 frames verworfen werden. Das resultierende frame wird danach entschlüsselt und an das ungesicherte interface weitergeleitet und damit dem ursprünglichen zielgerät zugestellt.

Die diffie hellman paramter für die eigentlichen datentransfers werden für jeden request-response-zyklus und für jede der 2 gesicherten leitungen neu erstellt. Das heisst den abgeleiteten key selbst kennen nur die involvierten 2 security gateways. Da die verwendeten keys nach dem transfer verworfen werden wird hier perfect forward secrecy erreicht.

Solange also sichergestellt ist dass zumindest ein kommunikationspfad verfügbar ist können entfernte standard KNX netze, verbunden durch gateways, miteinander kommunizieren. Es sei hier noch darauf hingewiesen dass der aufbau des protokolls es erlaubt, das konzept auf n statt auf 2 redundante leitungen zu generalisieren, womit noch grössere verfügbarkeit erreicht werden kann.

Abschliessend möchte ich noch kurz auf die implemtierung des prototypes eingehen, welcher gleichzeitig als evaluierung diente. Physikalisch aufgebaut wurde das versuchsnetz mittels 2 raspberryPI singleboard computer, als betriebsystem diente das debian derivat raspian. Für die businterfaces zu den gesicherten und ungesicherten kupferleitungen wurden busware USB kuppler verwendet, welche mittels einer von herren martin kögler programmierten API angesprochen wurden. Die auf den gateways für countersyncronization, schlüsselmanagement und nachrichtenaustasuch verantwortliche software wurde in C realisiert, wobei für jedes businterface dezidierte sende- und empfangsthreads implementiert wurden.

Auf kryptografischer ebene wurde auf die openSSL API zurückgegriffen. Für sämtliche

verwendeten MACs wurde SHA256 basierendes HMAC verwendet, wobei allerdings nur 4 byte des resultierenden outputs verwendet werden um den overhead im rahmen zu halten. Für die symmetrische verschlüsselung wurde 256 bit AES im counter mode verwendet, um auf paddings verzichten zu können.

Folgende Arten von keys werden verwendet: ein preshared key wird zum authentifizieren der syncronization- und discovery nachrichten verwendet. Ein 2. preshared key dient als AES key für die discovery nachrichten. Weiters wird ein eigener key von den diffie hellman parametern abgeleitet, welcher zum authentifizieren und verschlüsseln des eigentlichen datenaustausches verwendet wird, wobei hier als zugrunde liegende gruppe elliptische kurven gewählt wurden. Diese haben den vorteil dass, verglichen zum 'herkömmlichen' diffie hellman, signifikant kürzere schlüssel ein gleiches mass an sicherheit bieten.

Ausblick: alles verschlüsseln / end-zu-end verschlüsselung

verabschiedung

TODO:

derstandard.at – artikel KNX
methodik
ausblick

ANSCHAUEN

komplexitätstheorie: 'runs in polynimally bounded time'
threads