Authentikation in Ad-hoc und Sensornetzwerken

André Weimerskirch

Horst Görtz Institut für IT-Sicherheit
Ruhr-Universität Bochum, Germany

Email: weika@crypto.rub.de

Zusammenfassung

Nahezu alle elektronischen Geräte werden heute mit einem Mikrochip ausgestattet. So ist heute schon fast in jeder Waschmaschine ein Chip eingebettet. In Zukunft ist zu erwarten, dass sich dieser Trend fortsetzt und zusätzlich eine drahtlose Kommunikationsmöglichkeit eingebaut wird, so dass schließlich ein weit verzweigtes drahtloses Netzwerk entsteht. Ein solches Netzwerk, das dezentral und selbstorganisierend ist, wird Ad-hoc Netzwerk genannt, bei leistungsschwachen Sensoren auch Sensornetzwerk.

Sicherheit in Ad-hoc Netzen muss genauso betrachtet werden wie in traditionellen Netzwerken, die Herangehensweise ist jedoch unterschiedlich. In dieser Arbeit zeigen wir diverse Ansätze zur Authentikation in Ad-hoc Netzen. Wir erläutern generelle Probleme, und gehen auf heute eingesetzte Verfahren ein, z.B. bei Bluetooth. Daraufhin erläutern wir einige von uns entwickelte Ansätze zur effizienten Authentikation in Ad-hoc Netzwerken.

Keywords: Sensornetzwerke, Ad-hoc Netzwerke, Authentikation, Bluetooth

1 Einleitung

Die meisten elektronischen Geräte werden heute mit einem Mikrochip ausgestattet. So hat heute nahezu jede Waschmaschine schon einen eingebetteten Chip integriert. In Zukunft ist zu erwarten, dass sich dieser Trend fortsetzt. Werden diese Geräte mit einer drahtlosen Kommunikationsmöglichkeit ausgestattet (z.B. Bluetooth), so könnten sie zusammen mit schon vorhandenen Computern in Form von Desktop PC's, Mobiltelefonen und elektronischen Organizern ein weit verzweigtes drahtloses Netzwerk aufspannen. Ein solches Netzwerk, das dezentral und selbstorganisierend ist, wird Ad-hoc Netzwerk

genannt. Ein Ad-hoc Netzwerk, das aus leistungsschwachen Sensoren besteht, die ihre Umgebung abtasten (z.B. Licht- oder Temperatursensoren), nennt man auch Sensornetzwerk. Ad-hoc Netze sind insbesondere sinnvoll in Szenarien, wo keine feste Infrastruktur besteht, oder wo diese zu teuer wäre. In solchen Netzwerken ist jeder Knoten auf seine Nachbarknoten angewiesen. Er bietet diesen seine Dienste an und nutzt genauso deren Dienste, z.B. bei dem Senden und der Weiterleitung von Datenpaketen. Ad-hoc Netze finden Anwendung in militärischen Szenarien, für die sie ursprünglich entwickelt wurden, aber auch in zivilen Szenarien. Beispiele für Ad-hoc Netzwerke sind folgende: (1) Notfallsituationen, in denen die Kommunikation ausgefallen ist; (2) Erweiterung traditioneller Netzwerke, um Kosten zu sparen; und (3) Errichtung temporärer Netzwerke für ein Gruppentreffen. Ad-hoc Netze werden in der Zukunft darüber hinaus Anwendung finden in fast allen Bereichen, z.B. innerhalb des Automobils und zwischen ihnen, im Home-Entertainment Bereich, in intelligenten Häusern, und im wearable Computer. Ad-hoc und Sensornetzwerke werden sogenannte pervasive (alles durchdringende) Netzwerke und ubiquitäre (allgegenwärtige) Computer ermöglichen, die unseren Alltag grundlegend verändern könnten.

Sicherheit in Ad-hoc Netzen muss genauso betrachtet werden wie in traditionellen Netzwerken, die Herangehensweise ist jedoch unterschiedlich. So sind die Geräte, die in Ad-hoc Netzen genutzt werden, weniger leistungsstark, und in vielen Szenarien gibt es keine zentralen Server, also keine zuverlässige Infrastruktur. Sicherheit wird jedoch in den meisten Fällen benötigt, angefangen vom zuverlässigen Routing über den geschützten Austausch von Dateien bis hin zu finanziellen Transaktionen, die mit Hilfe der Geräte (z.B. Handy) abgewickelt werden. Der Ausgangspunkt von Sicherheit ist die Authentikation. Nur wenn man weiß, mit wem man spricht, kann man über Verschlüsselung und Integritätssicherung nachdenken.

In dieser Arbeit zeigen wir diverse Ansätze zur Authentikation in Ad-hoc Netzen. Wir erläutern generelle Probleme, und gehen auf heute eingesetzte Verfahren ein, z.B. bei Bluetooth. Wir stellen häufige Anwendungsszenarien vor, und beschreiben mögliche Lösungen. Schließlich geben wir einen Überblick über eigene Forschungsergebnisse. Wir stellen einen neuen Ansatz zur Authentikation vor, der extrem effizient ist und auch für leistungsschwache 4-Bit CPU's (z.B. Sensoren) geeignet ist. Dabei geht es darum, dass die Geräte in der Lage sind, andere Geräte (genauer gesagt Dienstanbieter) wiederzuerkennen.

2 Sicherheit in Ad-hoc Netzwerken

Wie bereits erwähnt, ist zu erwarten, dass in der Zukunft ein Mikrochip in nahezu alle Geräte wie Kaffeemaschinen, Thermostate sowie Radiowecker eingebaut werden wird. Werden diese Geräte mit einer drahtlosen Kommu-

nikation ausgestattet, so könnte zusammen mit schon vorhandenen Rechnern wie Mobiltelefonen und PCs ein extrem weitverzweigtes drahtloses Netzwerk entstehen. Da diesem Netzwerk Geräte kontinuierlich hinzugefügt oder daraus entfernt werden können, sollte dieses Netzwerk selbstorganisierend und nicht auf eine zentrale Infrastruktur angewiesen sein. Jeder Knoten in dem Netzwerk verlässt sich dann auf seine Nachbarn, indem er diesen seine Dienste anbietet und deren Dienste beansprucht, z.B. bei der Weiterleitung und dem Senden von Datenpaketen. Solch ein Netzwerk wird Ad-hoc Netz genannt. Hierbei existiert keine einzelne Fehlerquelle oder Angriffspunkt, da es keine zentralen Server gibt. In vielen Fällen ist zu erwarten, dass stationäre Basisstationen den Zugang zum Internet ermöglichen.

2.1 Besonderheiten und Herausforderungen

Einführend soll klargestellt werden, dass sich die Sicherheit in Ad-hoc Netzen im Allgemeinen stark von der IT-Sicherheitsproblematik in Computernetzen (LAN-, Internet-, VPN-Sicherheit) unterscheidet. Die Letztere ist relativ vertraut und es stehen Lösungen wie beispielsweise Verschlüsselungssoftware, Firewalls, Intrusion Detection Systeme u.a. zur Verfügung. Wenn wir die hier vorliegende Problematik betrachten, wird deutlich, dass die genannten Lösungen aus der Welt der Computernetze zum großen Teil nicht übertragbar sind.

Ressourcenbeschränkung: Viele der zu schützenden Systeme werden mit vergleichbar schwachen eingebetteten Prozessoren, insb. 8 oder 16 Bit Mikrocontroller ausgestattet. Während auf einem Desktop PC die eingesetzte Kryptographie bezüglich Rechenintensität inzwischen nahezu irrelevant ist, ist die Lage bei eingebetteten Systemen, wie sie in Ad-hoc Netzen genutzt werden, anders. Hier ist es wünschenswert, größtenteils symmetrische Primitive einzusetzen, die etwa um einen Faktor 1000 schneller sind als asymmetrische kryptographische Primitive (z.B. digitale Signaturen). Trotzdem sind häufig asymmetrische Algorithmen erforderlich, die extrem arithmetikintensiv sind (z.B. Berechnungen mit 1024 Bit Operanden). Daher muss der Einsatz asymmetrischer Algorithmen minimiert werden, und ihre Implementierung sehr sorgfältig optimiert werden. Eines unserer Hauptziele ist es somit, eine quasi asymmetrische Funktionalität mit (fast nur) symmetrischen Methoden zu erreichen.

Physikalische Attacken: Eine zentrale Komponente für die Absicherung einer IT-Anwendung sind kryptographische Algorithmen. Sowohl symmetrische als auch asymmetrische Verfahren basieren darauf, dass die zu schützende Einheit (z.B. der Sensor) einen *geheimen* kryptographischen Schlüssel besitzt, der durch Angreifer nicht ausgelesen werden kann. Da jedoch insbesondere Sensoren durch potentielle Angreifer offen zugänglich sind und nur ein Mindestmaß an Abwehrmaßnahme bereitstellen können[1],

[1] Sensoren sollen möglichst preisgünstig sein, da sie in großer Anzahl verbreitet werden.

besteht die Gefahr, dass diese durch Seitenkanalangriffe in den Besitz des Schlüssel gelangen, und damit Teile manipulieren und klonen können. Seitenkanalattacken nutzen Informationen über den Verlauf des Stromverbrauchs oder des Zeitverhaltens von kryptographischen Algorithmen aus, um den Schlüssel zu rekonstruieren. Diese Attacken wurden gegen Ende der 90er Jahre das erste Mal vorgeschlagen, und es existieren zur Zeit eine Vielzahl von Gegenmaßnahmen einerseits und verbesserten Attacken andererseits. Viele der Ergebnisse in diesem Bereich wurden in der CHES Konferenz dargestellt [KP99, KP00, KNP01, KKP02]. Verwandt mit Seitenkanalattacken sind Angriffe, die durch Methoden des Reverse Engineering versuchen in den Besitz von geheimen kryptographischen Schlüsseln zu gelangen. Hierzu gehört beispielsweise das Auslesen von Speicherzellen im Prozessor oder in integrierten Schaltungen. Entsprechende Gegenmaßnahmen fallen in den Bereich des "Tamper Resistance". Fallbeispiele zu diesem Thema und den damit verbundenen Schwierigkeiten sind in [And01] zu finden.

Beschränkte Wartungsmöglichkeiten: In einigen Fällen wird es sehr schwer sein, bekannt gewordene Sicherheitsprobleme mit nachträglichen Änderungen zu verhindern. Dies ist leider aber der Alltag in der IT-Sicherheit in konventionellen Computeranwendungen: Nachdem eine neue Lücke bekannt geworden ist, werden beispielsweise Software-Patches installiert, oder der Virusscanner erhält Signaturen neuer Viren. Insbesondere für Sensornetzwerke wird solch ein patchen nicht möglich sein, insbesondere wenn Sicherheitsfunktionen in Hardware realisiert werden. Dies unterstreicht die Bedeutung eines einwandfreien Security-Engineerings in der Entwurfsphase, um spätere Änderungen so klein wie möglich zu halten.

Anwendungen: Typische Beispiele von Ad-hoc Netzen sind Bluetooth, HiperLAN2, und eingeschränkt auch WLAN (802.11). Ad-hoc Netze wurden ursprünglich in den 70'er Jahren im Rahmen militärischer Forschung (DARPA) entwickelt. Daher werden Anwendungen auch häufig in militärischen Bereichen oder in Katastrophengebieten gesehen, z.B. bei der Kommunikation und Überwachung von Soldaten, oder dem Aufbau eines ausgefallenen Telefonnetzwerks. Weitere Beispiele sind ein Netz mobiler Telefone mit Hilfe von Bluetooth, um kostenlos telefonieren zu können, und Sensornetzwerke. Einen ersten Eindruck winziger Mikrochips, die als Sensoren wie Temperatur- oder Lichtfühler in solch einem Sensornetzwerk fungieren können, geben die Smartdust Geräte, die an der Berkeley University entwickelt werden. Schon nahezu alltagstauglich sind sogenannte Radio Frequency Identification (RFID) Etiketten. Dies sind kleinste passive Elemente, die einen kurzen Bitstring speichern können. Es ist vorstellbar, dass RFID Etiketten bald die Barcodes ablösen werden. Dann müsste an der Supermarktkasse nicht mehr jedes Produkt aus dem Einkaufswagen genommen werden, um den Endpreis festzustellen, sondern alle Waren könnten gleich-

Daher werden physikalische Schutzmaßnahmen gering gehalten.

zeitig erfasst werden, wenn der Einkaufswagen durch die Kasse geschoben wird. Diese Technologie kann zu riesigen Einsparungen in allen Bereichen der Logistik führen. Weiterhin ist vorstellbar, dass Tablettenpackungen ein RFID Etikett beinhalten. Dadurch kann der Anwender automatisch vor Nebenwirkungen gewarnt werden, die durch die Einnahme verschiedener nichtverträglicher Tabletten auftreten könnten.

Sicherheit: Wegen der Beschränkungen der mobilen Geräte, insbesondere bezüglich Rechenleistung, Speicherkapazität und Batterieleistung, müssen die Sicherheitsanforderungen in Ad-hoc Netzen anders betrachtet werden als in statischen Netzwerken. So ist es zum Beispiel nicht möglich, jedes Datenpaket zum Zweck des sicheren Routings zu signieren, um den Ursprung sicherzustellen. Dies scheitert aufgrund der beschränkten Rechenleistung und der Notwendigkeit einer Public-Key Infrastruktur (PKI), die sich im Grundsatz nicht mit dem dezentralen Charakter eines Ad-hoc Netzes verträgt. Das sichere Routing von Datenpaketen ist ein wichtiges, aber bisher nicht grundsätzlich gelöstes Forschungsthema. Da in einem Ad-hoc Netzwerk jeder Knoten potentiell ein Router ist, müssen sichere und robuste Verfahren eingesetzt werden. Mögliche Lösungsansätze werden z.B. in [BH01, LPW03, WW03a] behandelt. Aufgrund der mangelnden physikalischen Sicherheit ist es in Ad-hoc Netzen zudem wichtig, dass einige bösartige Knoten die Funktionsweise des Netzes nicht gefährden.

Die Gefahren aufgrund mangelnder Sicherheit werden insbesondere in Szenarien sichtbar, in denen Alltagsgegenstände mit Sensoren ausgestattet sind. So können die oben erwähnten RFID Etiketten nicht nur Logistikabläufe beschleunigen, sondern sie können auch zur Erstellung von Kundenprofilen genutzt werden. Sie können aber auch eine reale Gefahr bedeuten. Es gibt Überlegungen, winzige RFID Etiketten in Geldscheinen einzubetten, um diese fälschungssicher zu gestalten. Dies bedeutet allerdings auch, dass ein Taschendieb mit einem einfachen Gerät sofort feststellen kann, wer große Geldmengen bei sich trägt. Eine mangelnde Sicherheit gefährdet also die Privatsphäre, und kann auch zu weiterem Schaden führen [Sta02]

2.2 Kommerziell verfügbare Lösungen

Heutige verfügbare Lösungen für Sicherheit in Ad-hoc und Sensornetzen sind klassische Lösungen, die in ähnlicher Form schon in herkömmlichen Netzen genutzt wurden. Im nachfolgenden wollen wir einige dieser Lösungen vorstellen.

Bluetooth: Bluetooth bietet verschiedene Sicherheitsmodi an. Es wird vor allem die Sicherheit auf dem Link Layer und auf der Applikationsebene unterschieden. Die Sicherheit auf dem Link Layer basiert letztlich auf einem gemeinsamen Schlüssel, der mit Hilfe einer Tastatur den beiden Geräten mitgeteilt wird. Dazu wird paarweise in beide Geräte dieselbe PIN eingegeben, und daraus ein Schlüssel für die Authentifikation und Verschlüsselung

abgeleitet. Dies funktioniert sicherlich in den meisten Anwendungen in der Unterhaltungs- und privaten Kommunikationsindustrie, z.B. zur Absicherung des Kanals zwischen Mobiltelefon und Bluetooth Headset, wird jedoch unübersichtlich bei einer größeren Anzahl an Geräten, die miteinander kommunizieren wollen. Dazu müssten bei n Geräten jeweils $n-1$ PIN Nummern eingegeben werden, insgesamt also $n(n-1)$ PINs. Aufgrund des mangelnden Komforts werden in der Praxis daher häufig kurze PIN Nummern verwendet, die dann leicht über ein vollständiges Ausprobieren herausgefunden werden können. Geräte, die keine Möglichkeit zur Eingabe einer PIN mitbringen, sind meist mit einer Standard PIN voreingestellt, was natürlich Sicherheitsprobleme aufwirft. Zudem sind einige akademische Angriffe auf die Verschlüsselung bekannt geworden, die jedoch in der Praxis bisher keine Anwendung finden, da sie für die zur Verfügung stehenden Ressourcen immer noch zu aufwendig sind [FL01]. Die Sicherheit auf Applikationsebene bietet einen beliebigen Sicherheitslevel, ist allerdings auch rechenintensiver. Wir gehen darauf unten näher ein.

WLAN: WLAN (802.11b) wurde ursprünglich als Erweiterung eines LAN geplant, und nicht zum Aufbau eines Ad-hoc Netzes. Trotzdem ist es möglich, WLAN im Ad-hoc Modus zu betreiben. WLAN bietet genauso wie Bluetooth eine Sicherheit auf der Ebene des Link Layer an, die durch den WEP Mechanismus implementiert ist. Hier wird ein 104 Bit langer Schlüssel eingegeben, mit dessen Hilfe alle Nachrichten verschlüsselt und gegen Manipulation gesichert werden. Allerdings können nur 4 Schlüssel vorgegeben werden, so dass ein paarweiser Schlüsselaustausch nur bedingt möglich ist. Seit der Einführung von WLAN sind jedoch zahlreiche Sicherheitsprobleme im WEP Algorithmus bekannt geworden, so dass schon heute eine verschlüsselte Verbindung in einem kurzen Zeitraum gebrochen werden kann. Abhilfe schafft auch hier nur die Sicherheit auf Applikationsebene.

Mica Motes und TinyOS: Die an der Berkeley Universität entwickelten Mica Motes bieten schon heute das einfache Betriebssystem TinyOS mit integrierter Sicherheit an. Die Mica Motes sind kleine Sensoren, die nur eine geringe Rechenleistung und wenig Speicher haben, und mit herkömmlichen Batterien laufen. Hier wird eine 8-Bit CPU mit 8 MIPS eingesetzt, die insgesamt 128 KB Flash Speicher und 4 KB SRAM aufweist [Cro04]. Die Sicherheit von TinyOS basiert auf einem einzigen gemeinsamen symmetrischen Schlüssel, der zur Verschlüsselung und Authentikation genutzt wird. Dies bedeutet natürlich auch, dass das komplette Netzwerk gebrochen ist, wenn der Schlüssel eines Sensors ausgelesen werden kann. Da Sensoren wie schon oben erwähnt nur wenig gegen physikalische Attacken geschützt sind, ist dies bei einem entsprechenden Aufwand zu erwarten. Trotzdem bieten die Mica Motes schon heute ein funktionierendes Sicherheitskonzept für kleinste Computer an.

Anwendungsebene: Auf Anwendungsebene stehen zahlreiche Möglichkeiten zur Verfügung, die aus der PC Umgebung kommen. Als konkretes Bei-

spiel sei hier nur PGP für Organizer wie den Palm genannt. Auf modernen PDAs laufen heute schon Linux Betriebssysteme, und somit stehen auch sichere Kommunikationskanäle über SSL bereit. Daraus ergibt sich, dass ähnliche Probleme wie in traditionellen Netzwerken bestehen. Es treten jedoch Unterschiede auf, da im Gegensatz zu herkömmlichen Netzen keine zentralen Server bereitstehen, die als vertrauensvolle Authorität fungieren. So können sich z.B. eine handvoll Leute auf einer Konferenz bei einem Schlüsseltausch nicht sicher sein, dass sie wirklich den richtigen Schlüssel benutzen, oder den Schlüssel eines Angreifers in der Mitte (man-in-the-middle Attacke). Es ist jedoch festzustellen, dass Ansätze auf Anwendungsebene nur leistungsstarken Geräten zur Verfügung stehen, und daher für leistungsschwache Geräte wie Sensoren keine Lösung darstellen.

2.3 Lösungsansätze

Wir betrachten nachfolgend mögliche Lösungsansätze, indem wir eine Kategorisierung der möglichen Szenarien angeben.

Militärische Anwendungen: Es gibt nur eine Autorität, der alle Knoten untergestellt sind. Weiterhin sind Kostenfragen hier weniger bedeutend. Daher sind Lösungen mit symmetrischer Kryptographie vorstellbar, oder auch Public-key Techniken. Da es nur eine Autorität gibt, könnte der Ansatz einer verteilten PKI gewählt werden [ZH99]. Hierbei wird die Aufgabe der Erstellung und Erneuerung von Zertifikaten auf viele Knoten verteilt, so dass eine Manipulation erschwert wird.

Heimanwendungen: Auch hier gibt es nur eine Autorität, nämlich den Besitzer der Heimgeräte. Daher können symmetrische Kryptographiemethoden genutzt werden. So ist z.B. ein Schlüsselaustausch als Eingabe einer PIN möglich, wie es z.B. letztlich bei Bluetooth und WLAN geschieht. Eleganter ist die Möglichkeit des Schlüsselaustauschs durch einen physikalischen Kontakt, wie es z.B. als Resurrecting Duckling durch Stajano vorgeschlagen wird [SA99].

Meeting: Wenn sich eine kleine Personengruppe trifft, die gegenseitig auf Dienste zugreifen will, bietet sich ein passwortbasierter Schlüsselaustausch an. Dabei wird ein Gruppenpasswort verteilt, also das Vertrauen der Leute ineinander auf das Ad-hoc Netz abgebildet. Bei Gruppen ohne Vertrauensbasis funktioniert dieser Ansatz jedoch nicht.

Heterogenes Pervasives Netzwerk: Hierbei besteht das Netzwerk aus heterogenen Knoten, die von einer Vielfalt an Autoritäten betrieben werden. Falls das Ad-hoc Netzwerk eine Verbindung zum Internet hat, können Sicherheitsmechanismen wie z.B. ein Kerberos Server genutzt werden. Andernfalls ist es notwendig, ein Vertrauensnetz aufzubauen. Dabei muss jeder Knoten mit der Zeit ein Vertrauensnetz zu anderen Knoten aufbauen. Dies findet in ähnlicher Form Anwendung im Web of Trust von PGP.

Sensornetzwerke: In Sensornetzwerken ist die Benutzung asymmetri-

scher Kryptographie aufgrund der beschränkten Rechenleistung der Sensoren nahezu ausgeschlossen, jedoch ist die Gefahr physikalischer Angriffe sehr hoch, so dass eine einfache symmetrische Lösung unzureichend ist. Wir haben jedoch gezeigt, dass ein vernünftiger Sicherheitsansatz mit rein symmetrischer Kryptographie in Sensornetzwerken möglich ist, der auch auf extrem leistungsschwachen Mikrochips realisiert werden kann [WW03b]. Wir stellen diesen Ansatz weiter unten genauer vor.

Abschließend ist hier zu sagen, dass keine einheitliche Lösung existiert, die Sicherheit in Ad-hoc Netzwerken sicher stellt. Bei der Erstellung von Lösungen müssen die Anforderungen und Annahmen gründlich gewählt werden. Dies bedeutet meist, dass Ansätze größtenteils auf symmetrischer Kryptographie basieren sollten, um teure Rechenoperationen zu vermeiden. Weiterhin muss natürlich aufgrund der großen Anzahl an Knoten eines pervasiven Netzwerkes immer davon ausgegangen werden, dass ein Teil der Knoten infiltriert und manipuliert werden kann. Dies sollte jedoch keinen merkbaren Einfluß auf das Gesamtnetzwerk haben.

3 Effiziente Authentikation in Sensornetzwerken

Wir haben oben die Sicherheitsprobleme in Ad-hoc und Sensornetzwerken beschrieben, und mögliche Lösungsansätze gegeben. Wir möchten nun einen eigenen Lösungsansatz vorstellen[2]. Dazu definieren wir zuerst die *Entitäten Wiedererkennung* (oder auch Zero-Common Knowledge Authentication) als den Vorgang, eine andere Entität, mit der in der Vergangenheit eine Verbindung bestand, wiederzuerkennen. Wir schlagen eine Lösung vor, die keinerlei Voraussetzungen macht, d.h., es müssen keine Schlüssel auf einem sicheren Kanal übermittelt oder vorher ausgetauscht werden, noch müssen besondere Anforderungen an die Hardware gestellt werden, um die Sicherheit des Netzwerkes zu erhalten[3]. Dies macht insbesondere in Sensornetzwerken Sinn, da man hier aufgrund der Rechenleistung keine digitale Signaturen einsetzen kann, und ein paarweiser voreingestellter symmetrischer Schlüssel das Sensornetzwerk unflexibel werden lässt. Dies ist insbesondere problematisch, da Sensoren häufig ersetzt und hinzugefügt werden müssen.

Wir möchten unseren Ansatz hier anschaulich erklären. Es treffen sich zwei Leute in der Wüste, die sich nicht ausweisen können (bzw. die den Ausweis des anderen nicht lesen können, da sie nicht wissen, wie er aussehen muss - z.B. zwei Personen aus unterschiedlichen Ländern). Diese beiden Personen helfen sich gegenseitig in der Wüste, und gewinnen Vertrauen in-

[2]Wir haben dies ursprünglich in[WW03b] vorgeschlagen, die allerdings einen Sicherheitsfehler enthielt. Stefan Lucks, der auch die Attacke gegen die alte Version gefunden hat, hat eine korrigierte Lösung in [LWWZ04] vorgeschlagen.

[3]Natürlich wird bei einer Attacke der Schlüssel eines Gerätes geklaut, so dass alle Kanäle, die auf diesem Schlüssel beruhen, unsicher sind. Darüber hinaus ist das Netzwerk jedoch nicht gefährdet.

einander. Zu einem späteren Zeitpunkt trennen sich diese beiden. Nun ist es vorteilhaft für sie, wenn sie in der Lage sind, sich in der Zukunft wiederzuerkennen, und auf ihr vorhandenes Vertrauen ineinander aufzubauen. Menschen tun dies mit Hilfe von Biometrie, sie erkennen das Gesicht der anderen Person wieder. In einer technischen Welt sind die Geräte aber nicht geographisch nahe, sondern möglicherweise weit voneinander entfernt, d.h., sie können nicht über einen sicheren Kanal kommunizieren, um ein Wiedererkennungsmerkmal auszumachen. Wir erwarten, dass die Entitäten Wiedererkennung Anwendung findet in Peer-to-Peer Netzwerken und Sensornetzen.

Wir zeigen nun eine Lösung, wie dies technisch realisiert werden kann mit einfachsten und extrem effizienten Methoden, die auf praktisch jedem leistungsschwachen Sensor laufen können. Wie schon vorher erwähnt sind asymmetrische Algorithmen rechenintensiv, so dass wir hier nur symmetrische Primitive nutzen. Trotzdem zeigen wir hier eine asymmetrische Lösung zur besseren Anschauung, die mit Hilfe einer digitalen Signatur ermöglicht wird, wie nachfolgend dargestellt. Dabei ist SK/PK ein asymmetrisches Schlüsselpaar, wobei eine Signatur über der Nachricht m mit dem privaten Schlüssel SK als $SIG(m, SK)$ berechnet wird. Die Verifikation wird ausgeführt als $VER(SIG(m, SK), m', PK)$. Die Verifikation ist erfolgreich, falls $m = m'$, und ansonsten ist sie nicht erfolgreich.

1 : B `generates` SK/PK `at random`
2 : B `sends` PK `to` A
`Repeat Steps 3 to 5 for each authentication process`
3 : A `sends challenge` r `to` B
4 : B `sends authenticated` $S := SIG(r, SK)$ `to` A
5 : A `checks` $VER(S, r, PK)$
`If 'valid',` A *`accepts`*`, otherwise she` *`rejects`*

Nun präsentieren wir unser effizientes Protokoll, das Entitäten Wiedererkennung gewährleistet. Dazu wählt sich Alice zufällig einen Wert a_0, und berechnet $a_1 := h(a_0), a_2 := h(a_1), \ldots, a_n := h(a_{n-1})$, wobei h eine Hashfunktion ist. Solch eine Kette nennen wir Hash-Kette, auch bekannt als Lamport's Hash-Kette. Bob wählt genauso b_0 und generiert Werte b_i für $i \leq n$. Die Werte a_n und b_n sind die letzten Elemente der Hash-Kette. Diese werden als öffentlicher Schlüssel zwischen Alice und Bob ausgetauscht. Sei nun m eine beliebige Nachricht, dann ist $MAC(m, k)$ der Message Authentication Code der Nachricht mit dem Schlüssel k, d.h. Wert, der die Nachricht m gegen Manipulation schützt, und die Authentikation des Senders der Nachricht gewährleistet.

Die Hauptidee unseres Protokolls ist nun folgendermaßen. Alice und Bob tauschen zuerst ihre öffentlichen Schlüssel a_n und b_n aus. Dieser Austausch kann von Mallory (dem Angreifer) abgehört werden, ohne die Sicherheit zu gefährden. Jetzt schickt Alice eine Nachricht, und authentifiziert diese mit einem Schlüssel a_i mit $i < n$. Aufgrund der Einwegeigenschaft einer Hash-

funktion kann nur Alice diesen Wert wissen. Daraufhin beweist Alice, dass sie den Wert a_i kennt. Dieser Vorgang kann beliebig oft wiederholt werden. Das Protokoll funktioniert wie in Tabelle 1 dargestellt. Dabei bezeichnet $\mathcal{P}(.)$ das Speichern einer Information, und $\stackrel{?}{=}$ einen Vergleich. Falls dieser Vergleich fehlschlägt, wird das Protokoll gestoppt, und kann später an derselben Stelle weitergeführt werden. Die Schritte $1-2$ werden nur einmal bei der Initialisierung ("dem Kennenlernen") des Paars Alice und Bob ausgeführt, wohingegen die Schritte $3-7$ beliebig oft für jede nachfolgende Authentikation durchgeführt werden.

Transmitting:	**Processing:**
`For key exchange, do only once Steps 1-2:`	
1. $A \rightarrow B : a_n$	$B : \mathcal{P}(a_n)$
2. $B \rightarrow A : b_n$	$A : \mathcal{P}(b_n)$
`For each authentication process, repeat Steps 3-8:`	
3.	A knows (b_{i+1})
	B knows (a_{i+1})
4. $A \rightarrow B : MAC(m, a_i)$	
5. $B \rightarrow A : b_i$	$A : h(b_i) \stackrel{?}{=} b_{i+1}$
6. $A \rightarrow B : a_i$	$B : h(a_i) \stackrel{?}{=} a_{i+1}, MAC(m, a_i) \stackrel{?}{=}' valid'$
7.	$A : \mathcal{P}(b_i)$
	$B : \mathcal{P}(a_i)$

Tabelle 1: Zero Common-Knowledge message authentication.

Der Austausch der öffentlichen Schlüssel a_n und b_n kann z.B. in Ad-hoc Netzen als Teil einer Dienstanfrage und Beantwortung geschehen. Alice bittet Bob z.B., ein Paket weiterzuleiten, und hängt dabei ihren öffentlichen Schlüssel a_n an das Paket. Bob entfernt diesen, hängt seinen Schlüssel b_n an das Paket, und leitet es weiter. Wird als Übertragungsmedium z.B. Bluetooth benutzt, wird Alice sehen, dass Bob ihr Paket weitergeleitet hat, und sie erhält Bob's öffentlichen Schlüssel. Natürlich kann Mallory genau dasselbe anstelle von Bob machen. Alice ist dies aber gleichgültig, für sie zählt nur, dass ihr Paket weitergeleitet wurde. Im nächsten Schritt gibt es also eine Beziehung zwischen dem Paar Alice und Bob, die einen ersten Schritt zum Aufbau einer Vertrauensbasis gemacht haben.

Dieses Protokoll ist beweisbar sicher, d.h. solange die eingesetzte Hash- und MAC Funktion sicher ist, kann auch das Protokoll nicht gebrochen werden. Das Protokoll benötigt insgesamt den Austausch von 3 Nachrichten, und verursacht einen Datenoverhead von 30 Bytes. Zusätzlich muss der öffentliche Schlüssel gespeichert werden, der jeweils 10 Bytes groß ist. Die benötigte Rechenzeit ist vernachlässigbar gering gegenüber einer digitalen Signatur, die jedoch nur eine Nachricht von 40 Bytes benötigt. Für die Länge der Hashketten kann ein kleiner Wert n gewählt werden. Ist die Hashkette

fast aufgebraucht, kann einfach als Nachricht eine neue Hashkette übertragen werden, also $m := a'_n$ bzw. $m := b'_n$.

Wir erwarten, dass unser Protokoll Anwendung finden kann in Peer-to-Peer Netzen, in Routingprotokollen, und in Sensornetzwerken. Eine Erweiterung des Protokolls, die darüber hinaus eine Identifikation ermöglicht, kann in [WW03a] nachgelesen werden. Diese Version benötigt allerdings weitergehende Ressourcen wie eine schwache Zeitsynchronisierung und etwas leistungsstärkere Prozessoren.

4 Zusammenfassung und Ausblick

In dieser Arbeit haben wir das Problem der Sicherheit in Ad-hoc und Sensornetzwerken genauer beleuchtet. Wir haben die Problematik beschrieben, und verfügbare Lösungen beschrieben. Schließlich haben wir ein effizientes Protokoll zur Authentikation in Sensornetzen präsentiert.

Es ist zu erwarten, dass Ad-hoc und Sensornetzwerke in einigen Jahren überall zu finden sind. Sie werden möglicherweise Teil unseres Alltags sein. Dies ist insbesondere bei der RFID Technologie zu erwarten, die als Ersatz für den Barcode vorgesehen ist. Daher sind umfassende Sicherheitsprotokolle zu entwickeln, die auf diesen Kleinstrechnern funktionieren, und die Benutzer vor möglichen Gefahren durch Datenklau und Spionage schützen.

Literatur

[And01] R. Anderson. Protecting embedded systems — the next ten years. In Ç. K. Koç, D. Naccache, and C. Paar, editors, *Workshop on Cryptographic Hardware and Embedded Systems — CHES 2001*, volume LNCS 2162, pages 1–2. Springer-Verlag, 2001. Invited Talk.

[BH01] L. Buttyán and J.-P. Hubaux. Nuglets: a Virtual Currency to Stimulate Cooperation in Self-Organized Mobile Ad Hoc Networks. In *Technical Report DSC/2001/001,Swiss Federal Institute of Technology – Lausanne, Department of Communication Systems*, 2001.

[Cro04] Crossbow. Webpage, 2004. http://www.xbow.com.

[FL01] Scott R. Fluhrer and Stefan Lucks. Analysis of the e0 encryption system. In *Proceedings of the 2001 ACM Symposium on Applied Computing (SAC)*, 2001.

[KKP02] B. S. Kaliski, Jr., Ç. K. Koç, and C. Paar, editors. *Workshop on Cryptographic Hardware and Embedded Systems — CHES*

2002, volume LNCS 2523, Berlin, Germany, August 13-15, 2002. Springer-Verlag.

[KNP01] Ç. K. Koç, D. Naccache, and C. Paar, editors. *Workshop on Cryptographic Hardware and Embedded Systems — CHES 2001*, volume LNCS 2162, Berlin, Germany, May 13-16, 2001. Springer-Verlag.

[KP99] Ç. K. Koç and C. Paar, editors. *Workshop on Cryptographic Hardware and Embedded Systems — CHES'99*, volume LNCS 1717, Berlin, Germany, August 12-13, 1999. Springer-Verlag.

[KP00] Ç. K. Koç and C. Paar, editors. *Workshop on Cryptographic Hardware and Embedded Systems — CHES 2000*, volume LNCS 1965, Berlin, Germany, August 17-18, 2000. Springer-Verlag.

[LPW03] B. Lamparter, K. Paul, and D. Westhoff. Charging support for ad hoc stub networks. 2003. Special Issue on Internet Pricing and Charging: Algorithms, Technology and Applications.

[LWWZ04] S. Lucks, A. Weimerskirch, D. Westhoff, and E. Zenner. Entity Recognition and Message Authentication. In *to appear*, 2004.

[SA99] F. Stajano and R. Anderson. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In *The 7th International Workshop on Security Protocols*. Springer-Verlag, 1999. LNCS 1796.

[Sta02] Frank Stajano. *Security for Ubiquitous Computing*. John Wiley and Sons, February 2002.

[WW03a] A. Weimerskirch and D. Westhoff. Identity Certified Authentication for Ad-hoc Networks. In *ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '03)*, 2003.

[WW03b] A. Weimerskirch and D. Westhoff. Zero Common-Knowledge Authentication for Pervasive Networks. In *Selected Areas in Cryptography - SAC, 2003*, 2003.

[ZH99] L. Zhou and Z.J. Haas. Securing Ad Hoc Networks. In *IEEE Network Magazine*, volume 13, 1999.