

Sichere Routingverfahren für VANETs

Sergey Kovalev

4. Januar 2010

Es gibt eine Vielzahl von Routingverfahren für Ad-Hoc Netze. Diese werden direkt oder leicht modifiziert auch in VANETs eingesetzt. In dieser Seminararbeit wird untersucht, welche besonderen Anforderungen für das Routing in VANETs berücksichtigt werden müssen. Ein Augenmerk liegt hierbei auf der Sicherheit und daher werden auch die allgemeinen Angreifermodelle vorgestellt. Aus der Menge der zur Verfügung stehenden Routing-Verfahren werden Repräsentanten einzelner Kategorien der Sicherheitsprotokolle auf deren Funktionsweise und deren Anforderungserfüllung untersucht.

1 Einleitung

Täglich gehen im Straßenverkehr Zeit, Geld und sogar Menschenleben verloren. Seit einiger Zeit gibt es bereits Ansätze diese Situation mit Hilfe von Technik zu verbessern. Einer dieser Ansätze ist die Verkehrstelematik (*Intelligent Transportation System*), welche als Grundlage VANETs benutzt. *VANET* steht für *Vehicular Ad-hoc NETWORK*. Dabei sind die einzelnen Autos die Teilnehmer des Netzwerks. In VANETs findet eine Kommunikation zwischen den einzelnen Teilnehmern statt. Diese Informationen werden ausgewertet und können dazu benutzt werden, z.B. in Gefahrensituationen den Fahrer zu alarmieren. Auch weitere Anwendungen wie die automatische Führung eines Convoys mit Hilfe von VANETs sind denkbar, setzen allerdings eine ausgereifte und sichere Kommunikationsstruktur voraus. Denn falsche Auswertung, Verfälschung oder Löschen der Informationen beispielsweise im Falle eines Angriffs kann zu fatalen Folgen führen. Daher gilt es ein VANET möglichst sicher zu machen. Da ein VANET aber kein gewöhnliches Ad-hoc Netz ist und einige Besonderheiten mit sich bringt, müssen diese bei der Wahl der Sicherheitsmechanismen berücksichtigt werden. Die Menge der Sicherheitsprotokolle muss hinsichtlich dieser Kriterien untersucht und ausgewertet werden, womit sich auch diese Seminararbeit beschäftigt.

2 Grundlagen

2.1 Besondere Charakteristiken von VANETs

Als eine Modifikation von Ad-Hoc Netzen haben VANETs auch modifizierte Anforderungen, die an das Netz gestellt werden. Die allgemeinen Anforderungen, wie z.B. selbst organisierende Struktur, bleiben auch in VANETs weiterhin bestehen und werden daher nicht explizit betrachtet. Folgende Liste enthält die besonderen Charakteristiken von VANETs[1], aus welchen sich auch deren Anforderungen ableiten lassen.

Hohe Anzahl von Knoten. VANETs können als Basis für Verkehrstelematik (*Intelligent Transportation System*) betrachtet werden und es kann davon ausgegangen werden, dass in der nahen Zukunft immer mehr Fahrzeuge mit den Kommunikationssystemen ausgestattet werden. Außerdem ist es auch nicht auszuschließen, dass diese Kommunikation zwischen Fahrzeugen irgendwann als Standard angenommen wird. Das bedeutet, dass ein VANET aus sehr vielen Teilnehmern bestehen wird. Somit ist das eine der wichtigsten Anforderungen: Ein VANET muss in der Lage sein mit einer sehr hohen Anzahl von Knoten umgehen zu können.

Hohe Mobilität und häufige Topologieänderungen. Einzelne Teilnehmer werden sehr wahrscheinlich oft ihren Standort wechseln oder sich mit hohen Geschwindigkeiten bewegen, z.B. auf einer Autobahn. Das Netz ist sehr dynamisch und die Topologieinformationen müssen dauerhaft up-to-date gehalten werden. Außerdem muss der Datenaustausch schnell stattfinden: Wenn sich beispielsweise zwei Fahrzeuge aufeinander zubewegen, müssen Pakete in der Lage sein in Sekunden versendet und ausgewertet zu werden.

Anwendungsanforderungen an die Zuverlässigkeit. Die auf VANETs basierenden Anwendungen sind stark von den gelieferten Informationen abhängig. Während es bei den Anwendungen, wie z.B. Visualisierung von allgemeinen Verkehrssituationen auf Autobahnen, der Verlust eines Pakets sogar ignoriert werden kann, gibt es auch weitere Applikationen, bei welchen Entscheidungen für das Fahrverhalten getroffen werden müssen. Verfälschung oder Verlust der auszuwertenden Informationen könnte über Leben und Tod eines Teilnehmers entscheiden und ist somit unbedingt zu vermeiden. Auch Verzögerung von Sicherheitsinformationen kann wegen hoher Latenzzeiten diese Informationen unnützlich machen. Daher sollte die Latenz auch möglichst immer niedrig gehalten werden.

Sicherheitsrelevante Verkehrsinformationen Wenn es sich bei gesendeten Daten um Sicherheitsinformationen zu der Verkehrssituation handelt, sind diese Informationen im Interesse aller Teilnehmer. Sie sind nicht als vertraulich zu behandeln und müssen dem globalen Zugriff zur Verfügung stehen.

Datenschutz. Informationen, die hingegen persönliche Daten enthalten, sind streng vertraulich und dürfen nicht an die Umwelt gelangen. Hierbei handelt es sich um die Fahrzeugidentifikation, dessen Geschwindigkeit, Position, Routenziel und eventuell

Fahrer- und Ladungsinformationen. Wie auch in allen anderen Branchen stehen diese Punkte unter Datenschutz und müssen vor unautorisierten Zugriffen geschützt werden. Die einzelnen Teilnehmer eines VANETs müssen weiterhin Anonym für die Außenwelt erscheinen.

Eigene drahtlose Technologie. Derzeit gibt es einen extra für die VANETs modifizierten Standard IEEE 802.11p [2]. Dieser befindet sich zwar noch in der Entwicklungsphase, kann aber dennoch schon als Kommunikationsmittel benutzt werden. Dieser Standard kennzeichnet sich durch Unterstützung von Fahrgeschwindigkeiten bis zu 200 km/h und einen Entfernungsbereich von 1 km.

Geringere Strom- und Recheneinschränkungen. Im Gegensatz zu den klassischen Ad-Hoc-Netzen, z.B. Sensornetzen, verfügen die Teilnehmer eines VANETs über weit größere Energiereserven. Neben einem starken Akku, wird in jedem Fahrzeug auch beim laufenden Motor Strom erzeugt, sodass jeder Teilnehmer während seiner Aktivität dauerhaft ans Netz angeschlossen werden kann. Informationsoverhead, welcher von verschiedenen Sicherheitsprotokollen erstellt wird, kann ruhig beibehalten werden, da die Teilnehmer nun über genügend Energie verfügen. Dass der Energieverbrauch zum Versenden eines Bits einem Vielfachen von Rechenoperationen entspricht, ist in einem VANET keine Einschränkung mehr. Auch Rechenleistung und Speicherkapazität sind kein Problem mehr, sodass auch auf komplexere kryptographische Verfahren zugegriffen werden kann.

Zeitsynchronisation. Unter der Annahme, dass alle zukünftigen Fahrzeuge mit Navigationssystemen, also GPS-Modulen, ausgestattet werden, fällt die Notwendigkeit der Zeitsynchronisation weg, da über GPS die genaue Zeit abgefragt werden kann.

Zugang zu der Infrastruktur. Viele Sicherheitsmechanismen benötigen einen zentralen Server, von welchem z.B. die Vergabe von einzelnen kryptographischen Schlüsseln erfolgt. Es können eventuell die sog. RSUs (Road-Side-Unit) oder öffentliche Hotspots vorhanden sein, die einen globalen oder einen Internetzugang bieten können. Diese Einrichtungen sind aber nicht überall vorhanden. Es ist also davon auszugehen, dass ein Fahrzeug sich nicht immer im von RSU oder Hotspot abgedeckten Bereich befindet. Diese Tatsache muss bei Implementierungen von kryptographischen Verfahren immer berücksichtigt werden.

Zentrale Registrierung und periodische Inspektionen. Ein Fahrzeug besitzt seine eigene lebenslange Identifikation, welche von einem zentralen Server überprüft werden muss, um jeglichen Missbrauch zu verhindern. Folglich soll immer für eine Möglichkeit gesorgt werden die Zentrale zu kontaktieren. Ein weiterer Unterschied zu einem klassischen Ad-Hoc-Teilnehmer ist es, dass Fahrzeuge regelmäßig zu Inspektionen gebracht werden müssen, um die Zulassung im Straßenverkehr beizubehalten. Diese Inspektionen sollten dann auch dazu genutzt werden die Einhaltung der Sicherheitsstandards und die Integrität der Kommunikation zu überprüfen.

2.2 Angreifermodelle

Bevor die einzelnen Sicherheitsprotokolle auf deren Einsetzbarkeit überprüft werden können, muss zuerst das Vorgehen eines Angreifers untersucht werden. Die Angreifermodelle[3] aus dem Ad-Hoc-Bereich können für diese Zwecke direkt übernommen werden, da in VANETs die Kommunikation über einen offenen Funkkanal geschieht. Ein Angreifer kann also mit Hilfe einer Antenne Kommunikationspakete abfangen oder erzeugen. Des Weiteren wird nach Dolev-Yao[3] angenommen, dass der Angreifer sogar ein echter Teilnehmer des Netzes ist. Folgende Angriffe sollen also von den Sicherheitsprotokollen entdeckt und verhindert werden können:

Verfälschung von Routinginformationen. Bei diesem Angriff handelt es sich darum, Pakete mit Routinginformationen zu manipulieren. Es können z.B. nicht nur veraltete oder duplizierte Pakete versendet werden, sondern auch neue vom Angreifer erstellte Pakete mit falschen Routinginformationen. Folgende Fälle können beispielsweise bei diesem Angriff auftreten:

- Entstehung von Schleifen innerhalb des Netzes.
- Vorhandene Routen werden verkürzt oder verlängert.
- Das Netz wird in Unternetze aufgeteilt.
- Latenzzeiten werden erhöht.

Selective Forwarding. Die von Teilnehmern versendeten Pakete werden nur wahlweise weitergeleitet. Eine Variation dieses Angriffs ist es, die empfangenen Pakete zu verwerfen. Da dieses aber leicht von den Nachbarn entdeckt und als Angriff identifiziert werden kann, werden die Pakete nur teilweise verworfen, was dennoch zu Einschränkungen im Datenverkehr eines VANETs führen kann.

Sinkhole-Angriffe. Ziel dieses Angriffs ist den gesamte Datenverkehr in der Nachbarschaft über den Angreiferknoten zu leiten. Dabei wird der Angreiferknoten z.B. durch bessere Latenzzeiten oder stärkeres Funksignal für die Nachbarschaft zum Versenden von Informationen attraktiv gemacht. Wenn dieser Fall eintritt, ist der Angreifer in der Lage nicht nur auf alle Informationen zuzugreifen, sondern auch weitere Angriffsarten wie z.B. selektives Weiterleiten oder einen Wormhole-Angriff auszuüben.

Sybil-Angriffe. Bei einem Sybil-Angriff täuscht der Angreiferknoten mehrere Identitäten vor und kann z.B. die Effektivität von Fehlertoleranzmethoden einschränken. Außerdem können auch die GPS-Positionen eines Knoten vorgetäuscht werden, sodass sich ein Knoten an mehreren Orten gleichzeitig zu befinden scheint.

Wormhole-Angriffe. Wormhole ist eine Unterklasse eines Sinkhole-Angriffs. Zwei Angreiferknoten spannen einen sog. Tunnel auf, über welchen die gesamten Daten aus einem Teilnetz in ein anderes gesendet werden. Kombiniert mit einem Sybilangriff,

kann der aufgebaute Tunnel mehrere Teilnehmer vortäuschen und ist somit nur schwer zu entdecken.

Flooding-Angriffe. Beim Flooding-Angriff geht es darum ein Netz mit Informationen zu fluten. Als konkretes Beispiel wird ein Hello-Flood-Angriff betrachtet. Als Hello-Paket wird ein Paket bezeichnet, mit welchem sich ein Knoten im Netz bei den Nachbarn registriert. Ist die Senderantenne stark genug, kann der Angreifer auch sehr weit entfernt liegende Knoten erreichen und denen eine direkte Nachbarschaft vortäuschen. Die an den Angreifer adressierten Pakete werden folglich ins Leere geschickt. Wenn der Knoten dieses merkt, könnte er versuchen die Informationen über seinen Nachbarn zu versenden, welcher ohne weitere Updates von Routinginformationen ebenfalls ins Leere sendet.

Spoofing von Acknowledgements. Durch versenden von Ack's kann die Existenz nicht vorhandener Knoten oder einer guten Leitung vorgetäuscht werden. Auf diese Art kann der Angreifer z.B. die Knoten davon abhalten über seinen Weg die Informationen zu versenden oder im Gegenteil den meisten Datenverkehr auf sich zu ziehen.

2.3 Routingprotokolle

Im Kapitel 3 werden sichere Routingprotokolle vorgestellt und ausgewertet. Dabei handelt es sich um Erweiterungen zu den weit verbreiteten Protokollen AODV und DSR. Dieses Kapitel dient dazu deren Funktionsweise zu beschreiben und somit eine Grundlage für die Sicherheitsoptimierungen zu verschaffen.

2.3.1 AODV

AODV[4] ist ein extra für die Ad-Hoc-Netze entwickeltes Protokoll und steht für *Ad-hoc On-demand Distance Vector*. Dieses Protokoll ist auf den Einsatz in einer mobilen Umgebung abgestimmt und ist daher für die VANETs besonders geeignet. Des Weiteren gehört AODV zu den reaktiven (*on-demand*) Routingprotokollen und berechnet nur dann den Weg zum Ziel, wenn einer der Teilnehmer tatsächlich ein Paket versenden möchte. Die Funktionsweise des Protokolls lässt sich in zwei mögliche Vorgänge einteilen: *Route Discovery* und *Route Maintenance*.

Ermittlung eines Weges. *Route Discovery.* Die Topologie eines Ad-Hoc-Netzwerkes kann zu jedem Zeitpunkt als ein Graph beschrieben werden. Eine *Route Discovery* wird immer dann vorgenommen, wenn ein Knoten den Weg zu dem nächsten Nachbarn oder zum letztendlichen Ziel nicht kennt. Um die Darstellung der Funktionsweise zu vereinfachen wird die Annahme getroffen, dass alle Verbindungen zwischen zwei Knoten bidirektional sind. Eine Verbindung zwischen zwei Knoten ist vorhanden, wenn diese sich gegenseitig in ihren Sendereichweiten befinden. Dabei werden also die Fälle vernachlässigt, bei welchen Knoten A an einen anderen Knoten B Informationen zusenden kann, aber aufgrund verschiedener Störungen B nicht an A zurücksenden kann. Zur Beschreibung des Algorithmus wird die Abbildung 1 betrachtet, bei welcher Knoten A an den

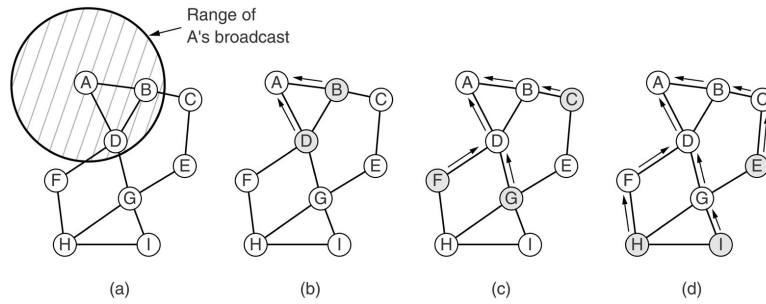


Abbildung 1: Route Discovery[4]

Knoten *I* ein Paket schicken möchte. Knoten *A* schlägt in seiner eigenen Routing-Tabelle nach und findet keinen Eintrag für *I*. Um *I* zu finden wird von *A* ein Route-Request-Paket (RREQ) erstellt und an die Nachbarn geschickt. Dieses Paket beinhaltet die in der Abbildung dargestellten Felder.

Source address	Request ID	Destination address	Source sequence #	Dest. sequence #	Hop count
----------------	------------	---------------------	-------------------	------------------	-----------

Abbildung 2: Aufbau eines Route-Request-Pakets[4]

Source und *Destination Adress* beinhalten in der Regel IDs von Quell- und Zielknoten. *Request ID* (*Anforderungskennung*) ist ein lokaler Zähler, der bei jedem weiteren RREQ-Broadcast um 1 erhöht wird. RREQ-Pakete werden durch die Felder *Source Adress* und *Request ID* eindeutig bestimmt und werden dazu benutzt Duplikate zu identifizieren und diese zu verwerfen. Die *Source Sequence Number* ist der Folgezähler von *A* und die *Destination Sequence Number* ist der aktuelle Wert der Folgennummer von *I*, die *A* erhalten hat. Der letztere wird dazu benutzt, einen neueren Weg von dem gespeichertem alten zu unterscheiden. Ist die *Destination Sequence Number* größer, so ist der Weg neu und wird an die Quelle zurückgesendet. *Hop-Count* (*Teilstreckenzähler*) zeigt die Anzahl der bisher gemachten Hops (Sprünge) und zeigt somit die Länge des Weges an.

Wenn das RREQ-Paket bei den Knoten *B* und *D* ankommt, schlagen die beiden Knoten die Einträge in deren Routingtabellen nach, um *I* aufzufinden. Da es auch hier für *I* keinen Eintrag gibt, wird erneut rundgesendet. Dabei wird der Rückweg zu *A* eingetragen, welcher in der Abbildung 1 durch Rückpfeile dargestellt ist. Die Duplikate (in diesem Fall die Pakete von *D* zu *B* und umgekehrt) werden wie beschrieben beseitigt. Weitere Rundsendungen werden von Knoten solange vorgenommen, bis der Knoten *I* erreicht wird. Wenn das Paket von *G* zu *I* ankommt, stellt *I* fest, dass es das Ziel der RREQ ist und sendet wie in der Abbildung 3 einen *Route-Reply*-Paket zurück (RREP).

Die Werte *Source* und *Destination Adress* werden dem RREQ-Paket entnommen. Die *Destination Sequence Number* wird aus dem Knoten eigenen Zähler übernommen wäh-

Source address	Destination address	Destination sequence #	Hop count	Lifetime
----------------	---------------------	------------------------	-----------	----------

Abbildung 3: Aufbau eines Route-Reply-Pakets[4]

rend der *Hop-Count* wieder auf 0 gesetzt wird. Die Lebensspanne (Lifetime) gibt an wie lange der Weg gültig ist. Das auf diese Art zusammengesetzte RREP-Paket wird an den Knoten gesendet, von welchem das RREQ-Paket kam (hier G) und gelangt über den Rückweg wieder zu A.

Bei jedem Knoten, welcher sich auf dem Rückweg befindet, wird das RREP-Paket untersucht und falls der Weg zu *I* bisher unbekannt, neuer oder kürzer ist, in der eigenen Routingtabelle gespeichert. Auf diese Weise lernen die auf dem Rückweg liegenden Knoten als Nebeneffekt den Weg zu *I* mit.

Dest.	Next hop	Distance	Active neighbors	Other fields
A	A	1	F, G	
B	B	1	F, G	
C	B	2	F	
E	G	2		
F	F	1	A, B	
G	G	1	A, B	
H	F	2	A, B	
I	G	2	A, B	

Abbildung 4: Routingtabelle des Knoten D nach einer Topologieänderung[4]

Verwaltung von Wegen. *Route Maintenance*. Sind die Wege einmal in der Routingtabelle eines Knoten gespeichert, so muss in periodischen Abständen deren Existenz überprüft werden. Dafür werden Hello-Pakete an die Nachbarn verschickt, auf welche geantwortet werden muss. Wenn keine Antwort ankommt, weiß der Sender, dass es keine Verbindung mehr zu diesem Knoten gibt. Dieses kann z.B. durch eine Topologieänderung oder eine Abschaltung des Knoten eintreten. Des Weiteren werden auch die Nachbarn des Senders darüber informiert, dass ein inaktiver Knoten vorliegt. Angenommen ist der Knoten G (Abb. 2d) nicht mehr erreichbar und wird als solcher vom Knoten D identifiziert. Knoten D informiert die Knoten A und B darüber, dass weder der Knoten G noch der Knoten I über den derzeit bekannten Weg erreichbar ist. Auf diese Art werden die nicht mehr gültigen Einträge aus den Routingtabellen gelöscht.

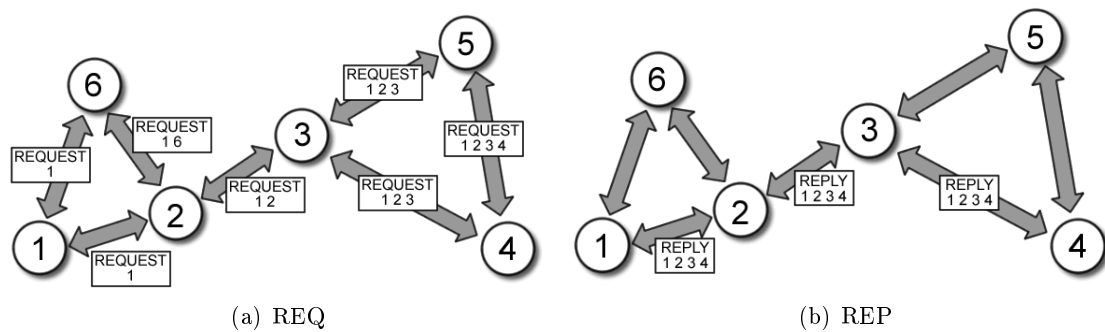


Abbildung 5: Route Request und Reply in DSR [5]

2.3.2 DSR

DSR steht für *Dynamic Source Routing* und ist ebenfalls ein reaktives Protokoll, welches aus den *Route Discovery* und *Route Maintenance* Vorgängen besteht. Der wesentliche Unterschied zu AODV liegt darin, dass DSR zu der Kategorie *Source Routing* gehört. Das bedeutet, dass der Paketheader des Absenders (*Source*) die komplette Route beinhaltet und nicht nur auf den nächsten Hop verweist. Ansonsten ist das Protokoll ähnlich zu AODV.

Wenn Knoten 1 (Abb. 5) beispielsweise eine Nachricht an den Knoten 4 versenden will, ihm aber die Route noch nicht bekannt ist, wird zunächst ein RREQ-Paket erstellt und broadcastet. Wenn dieses Paket bei einem weiteren Knoten ankommt, in diesem Fall bei dem Knoten 2, schaut der Knoten ob er den Weg nicht bereits kennt. Da dieses nicht der Fall ist, trägt sich der Knoten in die Route ein und das RREQ-Paket wird weiterversendet. Etwas später erhält der Knoten 2 noch ein RREQ-Paket vom Knoten 6, welches aber als Duplikat eingestuft und aus diesem Grunde verworfen wird.

Wenn das RREQ-Paket bei dem Knoten 4 ankommt, tritt der Fall ein, in welchem der Knoten die Route zum Ziel kennt, hier - der Knoten selbst. Ein RREP-Paket wird erstellt und per Unicast an den Knoten 1 (Source) geschickt. Wenn das RREQ-Paket von dem Knoten 1 empfangen wird, wird die Nachricht an den Knoten 4 über die im Header gespeicherte Route direkt zum Ziel weitergesendet.

Ähnlich wie in AODV werden auch die Routen gewartet (*Route Maintenance*). Tritt ein Fehler bei der Übertragung von Nachrichten ein, oder fällt einer auf dem Weg liegender Knoten aus, muss die Route gewartet werden. Mittels Error-Pakete werden weitere Knoten darüber informiert. Sind andere Routen im Cache der Knoten vorhanden, werden diese als Alternative ausgewählt. Ansonsten erfolgt das Versenden eines erneuten RREQ-Pakets.

3 Sichere Routingverfahren und deren Einsetzbarkeit in VANETs

Die in der Hauptquelle [1] vorgestellten sicheren Routingverfahren lassen sich nach dem verwendeten Sicherheitsmechanismus grob in fünf verschiedene Kategorien einordnen. In diesem Kapitel wird jeweils ein Repräsentant für jede Kategorie auf seine Funktionsweise untersucht und anschließend hinsichtlich der in dem Kapitel 2.1 beschriebenen Anforderungen analysiert.

Protokoll	Sicherheitsmechanismus
ARAN	Asymmetrische Kryptographie
SAODV	Asymmetrische / Symmetrische Kryptographie
ARIADNE	Symmetrische Kryptographie
CONFIDANT	Reputation-System
DCMD	Plausibilität / Reputation-System

3.1 ARAN

ARAN[6] ist eine Abkürzung für *Authenticated Routing for Ad hoc Networks* und erweitert das AODV-Protokoll um die Authentifizierung der Teilnehmer, während der Phasen von Route-Discovery und Route-Maintenance.

Zu seiner Funktionsfähigkeit braucht ARAN einen vertraulichen Certificate Server, dessen öffentlicher Schlüssel allen berechtigten Teilnehmern bekannt ist. Um einem Netzwerk beizutreten, bekommt jeder Knoten von dem Server ein Zertifikat, welches aus der IP-Adresse des Knotens, seinem öffentlichem Schlüssel, einem Zeitstempel der Zertifikatstellung und der Ablaufzeit des Zertifikats besteht. Dieses Zertifikat wird nun dafür benutzt, die an einem Datenaustausch teilnehmenden Knoten zu authentifizieren.

In der Route-Discovery Phase wird von dem Absender ein *Route Discovery Packet (RDP)* erstellt, welcher aus seinem Zertifikat, eine Nonce, einem mit dem eigenen Schlüssel signierten Zeitstempel und dem Zielknoten zusammengesetzt wird. Jeder weitere Empfänger überprüft die Signatur des Absenders und notiert sich seine IP-Adresse. Danach werden sowohl das Zertifikat als auch die Signatur des Absenders entfernt, wonach die Nachricht weiterversendet werden kann. Der jetzige Absender fügt nun sein Zertifikat der Nachricht hinzu und signiert deren Inhalt.

Wenn der Zielknoten nun das RDP empfängt wird ein Reply-Paket (REP) erstellt. Dieser beinhaltet sein Zertifikat, einen Identifier, die IP-Adresse des Reques-Initiators, sowie seine Nonce und Zeitstempel. Das so zusammengesetzte REP wird per Unicast zurückgesendet. Auf dem Weg werden die Pakete wie auch in der letzten Phase gehandhabt. Nach dem Überprüfen der Signatur des letzten Absenders, werden sein Zertifikat und seine Signatur gelöscht. Der Nachrichteninhalt wird erneut vom jetzigen Knoten signiert und mit seinem Zertifikat per Unicast an den nächsten Knoten der Route versendet. Wenn das REP bei dem Initiator ankommt, werden die Signatur des Zielknotens und die Nonce überprüft.

Zusätzlich hat jede Route im ARAN eine bestimmte Lebenszeit. Wenn in dieser Zeit keine Informationen darüber versendet werden, wird die Route aus den Routing-Tabellen gelöscht. Wenn eine Nachricht dennoch über die Route versendet wird, generieren die Knoten eine signierte Fehlnachricht *error message (ERR)* und senden diese an den Initiator weiter. Die Fehlnachrichten werden aber auch dazu benutzt die nicht mehr vorhandenen Verbindungen zwischen zwei Knoten zu versenden, die beispielweise durch eine Mauer unterbrochen sind.

Auswertung: Dank symmetrischer Kryptographie besitzt ARAN einen hohen Sicherheitsstandard. Im Vergleich zu AODV, verhindert ARAN einige Angriffe wie z.B. Veränderung von Routing-Nachrichten verhindern. Auch Replay-Angriffen lassen sich durch Nonce und Zeitstempeln vermeiden. Dennoch hat ARAN Probleme mit dem Aspekt der Skalierbarkeit. Das Protokoll hat ein großes Overhead und auch die Latenzzeiten werden beeinträchtigt, da jedes Paket unterschrieben werden muss. Als weiterer Nachteil ist die Notwendigkeit eines zentralen Servers zu erwähnen, der in einem VANET nicht immer erreichbar ist.

3.2 ARIADNE

ARIADNE[7] ist eine Sicherheitserweiterung von DSR. Für die Authentifizierung kann dabei eine der drei folgenden Methoden benutzt werden:

- Digitale Signaturen: ein Public-Key im Netzwerk pro Knoten bekannt.
- Paarweise bekannte Schlüssel für jeden Knoten im Netzwerk
- Paarweise bekannte Schlüssel zwischen kommunizierenden Knoten und eine zusätzliche Broadcast-Authentifizierung.

Im ersten Fall handelt es sich nicht um ein asymmetrisches Kryptographieverfahren und ist somit eine sichere Alternative, hat aber einen deutlich höheren Aufwand. Im zweiten Fall geht es um ein einfaches symmetrisches Verfahren, daher wird die dritte Methode bevorzugt: ARIADNE mit der TESLA-Authentifizierungsmethode.

TESLA[8] steht für *Timed Efficient Stream Loss-tolerant Authentication* und ist ein Authentifizierungsschema für Broadcastprotokolle. Neben den symmetrischen kryptographischen Funktionen wird TESLA auch um einen Mechanismus zur Verteilung der Schlüssel erweitert. Der Sender fügt in den *Header* seiner Nachricht einen *Message Authentication Code (MAC)* hinzu, welcher mit dem eigenen Schlüssel verschlüsselt wird. Die Nachrichten werden beim Empfänger solange im Puffer gehalten, bis der Schlüssel von dem Sender veröffentlicht wird. Die Veröffentlichung geschieht nach einem festen Zeitplan und ermöglicht den Teilnehmern die empfangenen Nachrichten zu authentifizieren. Dabei wird überprüft, ob die Nachricht vor der Veröffentlichung des Schlüssels versendet wurde. Wenn das nicht der Fall ist, wird die Nachricht verworfen. Die Zeit benötigt keine globale Synchronisation und eine einfache Zeitsynchronisation reicht, um die lokale Zeit des Senders abzuschätzen.

Bei einem RREQ wird nun vom Absender ein MAC an die Nachricht angehängt. Der Zielknoten kann nun nach der Veröffentlichung des Schlüssels den Sender authentifizieren und die Aktualität der Nachricht überprüfen. Auf diese Art sorgt Ariadne für die Authentizität und Integrität der RREQ-Pakete. Bei der *Route Discovery* müssen die Knoten die Pakete solange im Puffer behalten, bis der entsprechende TESLA-Schlüssel veröffentlicht wurde. Danach werden die Pakete weitergesendet.

Bei der Routenverwaltung existiert ein Limit an fehlgeschlagenen Übertragungen, nach dessen Überschreitung ein RERR-Paket an den Route-Initiator gesendet wird. Da in Ariadne auch die RERR-Pakete signiert werden müssen, bekommt der Angreifer keine Möglichkeit falsche Pakete zu versenden.

Auswertung: Ariadne erweitert das DSR-Protokoll um die Sicherheitsmechanismen, die z.B. einen Schutz gegen Routing-Schleifen und Replay-Angriffe bietet. Es kommt aber zu Problemen bei den Latenzzeiten bei den einzelnen Übertragungen, die dadurch zustande kommen, dass auf die Veröffentlichung des TESLA-Schlüssels gewartet werden muss.

3.3 CONFIDANT

CONFIDANT[5] ist eine Abkürzung für *Cooperation Of Nodes: Fairness In Dynamic Ad hoc Networks* und ist ursprünglich als Erweiterung für DSR entworfen worden. Das CONFIDANT-Konzept kann jedoch auch als Erweiterung für andere Protokolle benutzt werden. Das Ziel dieses Konzeptes ist es, die böartigen Knoten in einem Netz zu identifizieren, indem deren Aktivitäten überwacht und entsprechend in einem Reputation-System ausgewertet werden. Somit können die nicht korrekt handelnden Knoten aus Sendeaktivitäten ausgeschlossen werden.

CONFIDANT bringt vier neue Elemente mit sich: einen Überwachungsmonitor (*monitor*), einen Vertrauensmanager (*Trust Manager*), ein Reputation-System und einen Pfad-Manager (*path manager*). Der Monitor dient dazu, das Sendeverhalten von Knoten in seiner Reichweite zu überwachen. Wird ein Paket gesendet oder weitergeleitet, überprüft der Monitor ob diese Aktion auch korrekt durchgeführt wurde. Durch passive Acknowledgements oder eine allgemeine Überwachung von Protokollaktivitäten wird überprüft, ob die Route eingehalten wird, also ob der nächste Knoten tatsächlich die Nachricht empfangen und weitergeleitet hat. Dabei können bei einem Knoten folgende Anomalien [5] festgestellt werden:

- Daten- oder Kontrollnachrichten werden nicht weitergeleitet oder auf andere Arten manipuliert.
- Es fließen ungewöhnlich viele Daten über einen Knoten. Der Knoten behauptet eine sehr gute Route zu besitzen oder antwortet unerlaubt schnell auf RREQs.
- Eine Route wird verändert obwohl keine Fehler gemeldet worden sind.
- Keine Fehlermeldungen vom Knoten, obwohl dieser von Nachbar als fehlerhaft eingestuft wird.

- Eine Route wird ungewöhnlich häufig aktualisiert.

Wird vom Monitor eine Anomalie festgestellt, so wird das Reputation-System darüber informiert und macht einen entsprechenden Eintrag. Dabei wird die Reputation nur bei hinreichender Anzahl von Anomalien (Beweisen) geändert, welche verschiedene Gewichtung haben können. Sendet ein Knoten Informationen über eigene Anomalien, so sind diese schwer gewichtet. Rückmeldungen über Fehlverhalten eines Knotens von deren Nachbarn werden mit einem kleineren Gewicht versehen. Die Summe der Gewichte der Fehlermeldungen entscheidet nun über die Reputationsveränderung.

Wenn unglaubwürdige Knoten festgestellt werden, kommt der *trust manager* ins Spiel. Dieser ist für das Aussenden und Empfangen der sog. ALARM-Pakete zuständig. Ein Knoten kann ebenfalls ein ALARM-Paket absenden, falls er seine Aktivitäten für ungewöhnlich hält. Das Reputation-System ist dann für die Auswertung dieses ALARMS zuständig. Wird ein festgesetztes Limit an ALARMS für einen Knoten überschritten, so wird der Knoten als nicht vertrauenswürdig eingestuft.

Anhand der durch diese Schritte gewonnen Informationen, führt der *Path Manager* eine Auswertung der Routen durch:

- Den Routen werden anhand von Sicherheitskriterien verschiedene Ratings zugewiesen.
- Die Routen, die einen oder mehrere verdächtige Knoten beinhalten, werden gelöscht.
- RREQ-Pakete von verdächtigen Knoten werden ignoriert.

Auswertung: CONFIDANT eignet sich besonders gut, wenn es sich um ein eher kleineres Netzwerk handelt, welches geringe Mobilität aufweist. Durch Überwachung einzelner Knoten und deren Reputation-Tabellen können die meisten Angriffe entdeckt werden. Steigt jedoch die Anzahl der Teilnehmer in einem Netz, so können die vom Reputation-System auszuwertenden Tabellen für die einzelnen Knoten sehr groß werden. Bei sehr hoher Mobilität kann es zu sehr stark steigendem Overhead innerhalb des Netzes kommen.

3.4 SAODV

Wie der Name schon vermuten lässt, steht *SAODV* für *Sicheres AODV (Secure Ad-hoc On-demand Distance Vector)*[9]. Das Hauptziel dieses Protokolls ist das Absichern von Route-Discovery und Route-Maintenance Prozessen. Hier wird davon ausgegangen, dass der Angreifer bereits im Netz aktiv ist, die Pakete manipuliert und seine Identitäten vortäuscht. Die Integrität und Authentizität stehen bei SAODV weit im Vordergrund. Um diese zu gewährleisten werden zwei Verfahren benutzt: es werden Hash-Chains generiert um die variablen Informationen (Hop-Count) zu sichern, während die unveränderten Daten eines Pakets digital signiert werden.

Bei Erstellung von RREQ- und RREP-Nachrichten wird eine Zufallszahl generiert und eine Max-Hop-Zahl(n) gesetzt. Danach wird *Top-Hash* ausgerechnet, indem die Zufallszahl n -Mal gehasht wird. Zusätzlich wird auch der Identifier der Hashfunktion mit-versendet. Damit kann jeder Empfänger die Hop-Count-Zahl(m) verifizieren, indem die Hashfunktion Maximum-Hop-Count m -Mal auf den Hashwert angewandt wird. Ist der ausgerechnete Wert mit dem Feld Top-Hash identisch, ist der Hop-Count verifiziert. Um jetzt noch zusätzlich die Integrität der Daten gewährleisten zu können, werden alle Felder, außer Hop-Count und Hash-Feld digital signiert.

Bei dem Umgang mit den RREP-Nachrichten gibt es bei SAODV zwei Mechanismen. Einer davon gestattet es nicht den Zwischenknoten ein RREP-Paket zu generieren. In diesem Fall werden die RREQ-Pakete weitergeleitet, sofern die Signatur verifiziert wurde. Kommt dieses Paket bei dem eigentlichen Empfänger an, erstellt dieser das RREP-Paket, welches wie oben beschrieben gehandhabt wird. Im zweiten Fall wird es den Zwischenknoten erlaubt eigene RREP-Pakete zu generieren. Dabei wird nach der Prüfung der eigentlichen Signatur eine weitere hinzugefügt, welche z.B. die IP-Adresse des Ziels und die Gültigkeitszeit des Pakets beinhaltet. Die HELLO-, sowie RERR-Pakete werden auch signiert, da die Identität des Absenders hier ein wichtiger Punkt ist.

Auswertung: SAODV führt neue Sicherheitsmechanismen ein, bekämpft aber nicht das Problem des steigenden Paket-Overheads in Szenarien mit hoher Mobilität. Ganz im Gegenteil, der Overhead wird größer durch Versand von Signaturen, während die zur Auswertung eines Pakets benötigte Zeit aufgrund der Verwendung von kryptographischen Verfahren ebenfalls steigt. Diese Tatsache kann sogar dazu führen, dass es bei Knoten mit geringeren Rechenleistungen zu einem *Denial of Service (DoS)* kommt, ohne dass sich ein Angreifer im Netz befindet.

3.5 DCMD

DCMD[10] steht als Abkürzung für *Detecting and Correcting Malicious Data in VANETs*. Wie der Name schon verrät, wurde dieses Protokoll extra für die VANETs geschrieben. DCMD kann entweder eigenständig als Sicherheitsprotokoll implementiert oder als Erweiterung eines bestehenden Sicherheitsprotokolls benutzt werden.

DCMD basiert darauf, dass jeder Knoten ein Modell des gesamten VANETs besitzt. Darin ist das gesamte Wissen über ein VANET gespeichert. Derartiges Wissen beinhaltet Regeln, die z.B. aus physikalischen Eigenschaften eines VANETs abgeleitet werden (zwei Autos können im Normalbetrieb nicht die gleichen Koordinaten haben) oder aus deren Statistiken (Autos fahren in einer Stadt selten über 70 km/h). Dieses Modell wird zur Laufzeit mit direkt beobachteten Informationen vervollständigt unter der Annahme, dass die selbst gesammelten Informationen glaubwürdig sind. Pakete und Daten die dem VANET-Modell des Knoten entsprechen werden akzeptiert und weiterverarbeitet.

Die bezüglich des VANET-Modells inkonsistenten Daten werden anhand der Heuristik „*Adversarial Parsimony*“ bewertet. Kurz gefasst, besagt diese Heuristik, dass ein Angriff von einer kleineren Zahl bössartiger Knoten wahrscheinlicher als ein Angriff ist, bei welchem es zu einer Absprache zwischen einer Großzahl von Knoten kommt. Anhand dieses

Modells versucht ein Knoten für die Dateninkonsistenz eine möglichst einfache Erklärung für die Unstimmigkeiten zu finden und diese wieder zu korrigieren.

Auswertung: DCMD ist ein Beispiel für ein Protokoll, welches die Sicherheit des Netzes anhand der Plausibilität der gesendeten Informationen beurteilt. Das Protokoll kann sowohl eigenständig als auch als Erweiterung eines anderen Protokolls umgesetzt werden. Die hauptsächliche Schwierigkeit besteht darin, ein geeignetes Modell des VANETs zu erstellen, bzw. es an die Gegebenheiten des Einsatzbereiches anzupassen.

4 Auswertung

Zusammenfassend lässt sich sagen, dass die Anforderungen an die VANETs ähnlich wie in MANETs sind. Die neuen Kernanforderungen sind Anonymität, Vertraulichkeit, Skalierbarkeit und Umgang mit hoher Mobilität und häufigen Topologieänderungen.

Wenn der Aspekt der Skalierbarkeit betrachtet wird, sind symmetrische Verfahren eher ungeeignet. Bei einer Teilnehmerzahl n beträgt die Komplexität eines symmetrischen Protokolls $O(n^2)$, während ein asymmetrisches Verfahren weiterhin Komplexität $O(n)$ hat. Dieses Problem kann beispielsweise durch Einführung von TESLA gelöst werden, stößt dabei aber auf ein weiteres Problem, dass die Nachrichten eine höhere Übertragungsverzögerung haben und somit der Anforderung an die Echtzeit der Datenzustellung nicht entspricht.

Asymmetrische Verfahren scheinen die meisten Anforderungen zu erfüllen, haben dennoch große Schwierigkeiten im Umgang mit Vertraulichkeit und Anonymität. Es ist und bleibt weiterhin ein offener Aspekt dieser Seminararbeit und soll noch zusätzlich untersucht werden. Die nächsten Schwierigkeiten treten bei dem Processing-Aufwand auf. Obwohl die Netzwerkteilnehmer gute Rechenleistungen haben, sind dennoch DoS-Angriffe möglich, da jeder Teilnehmer eine große Menge von Informationen verarbeiten muss.

Diese Verfahren reichen aber dennoch, um ein Netz vor Angriffen seitens fremder Knoten zu schützen. Angriffe die aber von echten Teilnehmern des Netzes erfolgen, können durch Reputation-Systeme ausfindig gemacht werden. Als Problem stellt sich aber wieder die Skalierbarkeit heraus, da bei einer hohen Teilnehmerzahl die Tabellen zur Protokollierung und die Auswertung der Reputation sehr groß werden können. Als eine weitere Lösung bietet sich hier das Plausibilitätsmodell an, um welches ein VANET ohne großen Performanzverlust erweitert werden kann und somit eigentlich immer in einem VANET eingesetzt werden könnte.

5 Zusammenfassung

In dieser Seminararbeit wurden zunächst die Angreifermodelle und die Anforderungen an VANETs beschrieben. Die in der Hauptquelle[1] vorgestellten Verfahren (Abb.6) wurden nach dem verwendeten Sicherheitsmechanismus in Kategorien unterteilt und hinsichtlich der Anforderungen an die VANETs untersucht. Die Ergebnisse einzelner Protokolle

wurden anschließend in Form der Auswertung einzelner Sicherheitsmechanismen zusammengefasst.

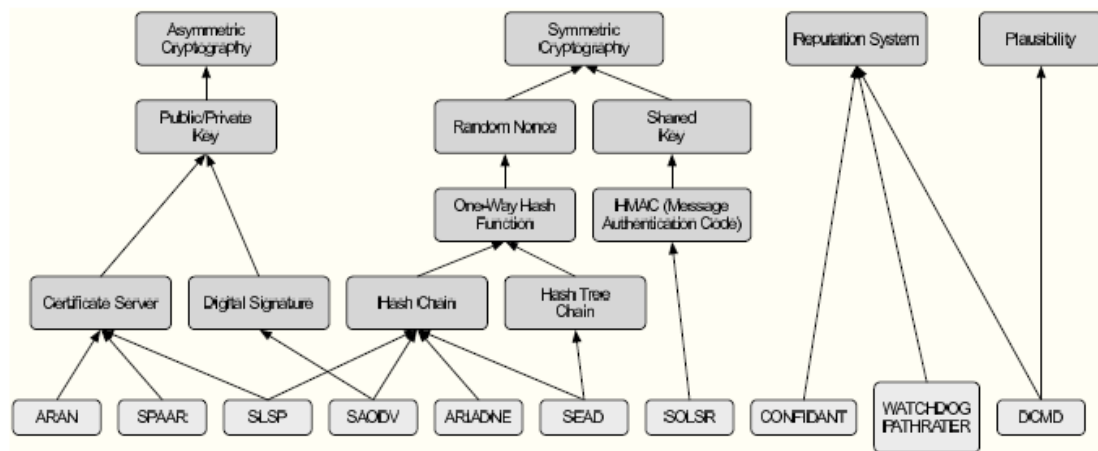


Abbildung 6: Sicherheitsprotokolle in VANETs[1]

Insgesamt lässt sich feststellen, dass die asymmetrischen Kryptographieverfahren den besten Schutz gegen externe Angriffe bieten. In Kombination mit der Auswertung der Plausibilität einzelner Teilnehmerhandlungen kann auch weiterer Schutz gegen interne Angriffe geboten werden. Dennoch gibt es in VANETs noch offene Probleme, wie z.B. die Gewährleistung der Anonymität, an welchen weiter geforscht werden muss, um irgendwann vielleicht alle Anforderungen eines VANETs erfüllen zu können.

Literatur

- [1] Emanuel Fonseca and Andreas Festag. A Survey of Existing Approaches for Secure Ad Hoc Routing and Their Applicability to VANETS. 2006.
- [2] Ieee 802.11p. <http://www.itwissen.info/definition/lexikon/802-11p-IEEE-802-11p.html> Letzter Aufruf: 10.11.2009.
- [3] Sergey Kovalev. Angreifermodelle für Sensornetze. 2009.
- [4] Andrew S. Tanenbaum. *Computernetzwerke*. Pearson-Studium, 2009.
- [5] Jean-Yves Le Boudec Sonja Buchegger. Performance Analysis of the CONFIDANT Protocol(Cooperation of Nodes: Fairness in Dynamic Ad-hoc NeTworks).
- [6] Kimaya Sanzgiri, Daniel LaFlamme, Bridget Dahill, Brian Neil Levine, Clay Shields and Elizabeth M. Belding-Royer. Authenticated routing for ad hoc networks. *IEEE Journal on selected areas in communications*, 23, 2005.

- [7] Frank-Peter Zeh. Seminar: Sicheres distance-vector-routing, 2004/05.
- [8] Andreas Schwarzkopf. Seminar: Sicherheit in car2car-kommunikation, autorisierung, 2008.
- [9] David Reinsch. Secure routing.
- [10] Jessica Staddon Philippe Golle, Dan Greene. Detecting and Correcting Malicious Data in VANETs. 2004.
- [11] Ling Yan and Simon Ortgiese. Aodv - Routing in ad hoc Netzwerken.
- [12] Adrian Perrig Yih-Chun Hu, David B. Johnson. Sead: secure efficient distance vector routing for mobile wireless ad hoc networks. *Ad Hoc Networks 1*, pages 175–192, 2003.
- [13] Zygmunt J. Haas Panagiotis Papadimitratos. Secure link state routing for mobile ad hoc networks. In *IEEE Workshop on Security and Assurance in Ad hoc Networks*, 2003.
- [14] Alec Yasinsac Stephen Carter. Secure position aided ad hoc routing.
- [15] Cai Fu Fan Hong, Liang Hong. Secure olsr. In *19th International Conference on Advanced Information Networking and Applications*, 2005.
- [16] Tobias Sprodowski. Mitigating routing misbehavior in mobile ad hoc networks.
- [17] Jonathan Sevy. Ad hoc routing: The aodv and dsr protocols. gicl.cs.drexel.edu/people/sevy/network/adhoc.ppt Letzter Aufruf 11.11.2009.