UNIVERSITÄT KOBLENZ · LANDAU

Fraunhofer SIT

Analyse und Evaluierung der Angriffserkennung in Car-to-Car Netzwerken

Diplomarbeit
zur Erlangung des Grades eines **Diplom Informatikers**
im Studiengang Informatik

an der Universität Koblenz-Landau

Betreuer:	Prof.Dr. Rüdiger Grimm Universität Koblenz-Landau
Zweiter:	M.Sc. Norbert Bißmeyer, Fraunhofer Institut für Sichere Informations-Technologie
Eingereicht von:	Mohammed Douiri
Matrikelnummer:	203210599

Darmstadt, im April 2010

Eidesstattliche Erklärung

Ich erkläre hiermit, dass ich die vorliegende Diplomarbeit selbständig und ohne fremde Hilfe verfasst, andere als die angegeben Quellen und Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

Darmstadt, im April 2010 Mohammed Douiri

Danksagung

Diese Arbeit bietet mir nicht nur die Möglichkeit, meine interessante interdisziplinäre Arbeit der letzten Monate zu dokumentieren, sondern eröffnet mir vielmehr die Gelegenheit, den Menschen zu danken, die zum Erfolg dieser Arbeit beigetragen haben.

Ich möchte diese Diplomarbeit meinen Eltern widmen, da sie mich in jeder Phase meines Studiums unterstützt haben. Sie haben nicht nur mein Studium zum größten Teil finanziert, sondern auch ständig ein sehr großes Interesse an meiner Arbeit gezeigt und mich so gut es ging unterstützt.

Schließlich gilt mein Dank noch all denen, die mich bei der Anfertigung meiner Diplomarbeit so kräftig unterstützt haben.

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

Kapitel 1

Einleitung

Trotz Abnahme der Verkehrsunfälle in den letzten Jahren haben laut dem Statischen Bundesamt Deutschland [6] im Jahr 2008 täglich durchschnittlich 12 Personen ihr Leben im Straßenverkehr verloren. Als Hauptunfallursache bei Unfällen mit Personenschaden wurde im selben Jahr Fehler beim Abbiegen, Wenden, Rückwärtsfahren sowie Ein- und Anfahren festgestellt. Zweithäufigste Unfallursache war die Missachtung der Vorfahrt bzw. des Vorrangs oder eine nicht angepasste Geschwindigkeit. Dies hat dazu geführt dass unter anderem die Verbesserung der reaktiven Sicherheit immer mehr verlangt wird. Bekannte Systeme in diesem Bereich sind ABS (Antiblockiersystem) und ESP (Elektronisches Stabilitätsprogramm). Diese Systeme können aber nur auf Sensordaten des eigenen Fahrzeugs zugreifen, und nicht frühzeitig vor einer Gefahr warnen. Deshalb wächst die Notwendigkeit nach einen gemeinsamen offenen Industriestandard, der es ermöglicht, eine Verbindung zwischen Fahrzeugen aufzubauen, um die proaktive Verkehrssicherheit zu verbessern.

Diese Verbindung muss bestimmte Eigenschaften erfüllen. Sie sollte drahtlos, Latenzfrei und kostengünstig sein. Somit könnte dem Fahrzeugführer ein wesentlich größeres Wahrnehmungsfeld gegenüber den bisherigen Systemen zur Verfügung gestellt werden. Diese Eigenschaften werden von den proaktiven Sicherheitssystemen (Ad-hoc Netze) erfüllt. Sie sind den bereits eingesetzten reaktiven Sicherheitssystemen weit überlegen. Obwohl die reaktiven Maßnahmen schon zu einem Rückgang der Todesopfer führen, bewirken sie dennoch offensichtlich keinen Rückgang der Unfälle, was aber die proaktiven Verfahren schon schaffen könnten .

Um einen Vergleich zwischen den proaktiven und reaktiven Sicherheitssysteme herzustellen, kann man folgendes Beispiel betrachten.
Unmittelbar nach einer unübersichtlichen Kurve auf der Autobahn hat sich ein Stau gebildet. Die bisherigen Sicherheitssysteme können diesen nicht früh genug erfassen, um einen Unfall zu vermeiden. Dafür müssen sie sich darauf beschränken, die Unfallfolgen zu mildern.

Verständigen sich die Fahrzeuge allerdings in einem automobilen Ad-hoc Netz (Vehicular Ad-hoc Network oder kurz: VANET), erhalten die Fahrer frühzeitig eine Warnung vor dem Stau und können sich diesem langsam nähern oder ihn im Idealfall sogar durch Verlassen der Autobahn frühzeitig umfahren. Abbildung 1.1 stellt eine solche Situation dar.

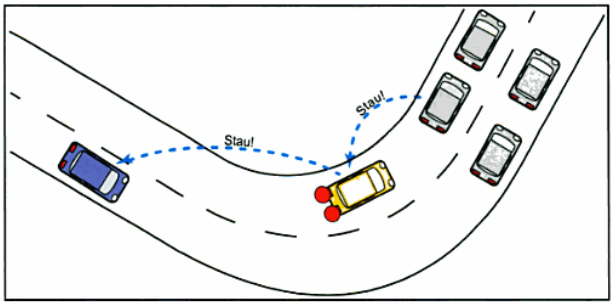

Abbildung 1.1: Warnung vor einem Stau durch ein VANET [18]

Wenn die Nachrichten in einem Ad-hoc Netz ungesichert übertragen werden und dabei die IT-Sicherheit vernachlässigt wird, könnten beliebige Manipulationen durch Angreifer auftreten. Im vorherigen Beispiel könnte eine falsche Staumeldung von einem Angreifer dazu führen, dass die folgenden Fahrzeuge langsamer fahren und dabei echte Staus bilden. Weiterhin könnte es passieren, dass sie einen unnötigen Umweg nehmen müssen und dabei Zeit verlieren und zusätzlichen Kraftstoff verbrauchen.
Daraus folgt, dass eine Absicherung des VANETs gegen Angriffe sehr wichtig ist, damit es auch richtig funktionieren kann. Zudem gehört die Privatsphäre des Netzteilnehmers zur Kategorie der Eigenschaften, die geschützt werden sollen. Dieser Anspruch ergibt sich aus dem Recht der informationellen Selbstbestimmung[1]. Dieser Schutzbedarf ist sehr wichtig. Denn wenn er vernachlässigt wird, könnten Angreifer Informationen unbemerkt sammeln, um beispielsweise die Position einer bestimmten Person zu verfolgen. Durch den Schutz der Privatsphäre sollen solche Angriffe verhindert werden.

Diese Arbeit befasst sich aus dem Grund mit der Erkennung, der Kategorisierung und der Analyse der verschiedenen möglichen Angriffe auf VANETs, wie im folgenden Kapitel beschrieben wird.

[1]Dieses Recht ist nach Ansicht des Bundesverfassungsgerichts aus Artikel 2 Absatz 1 in Verbindung mit Artikel 1 des Grundgesetzes für die Bundesrepublik Deutschland Bestandteil unserer verfassungsmäßigen Ordnung

1.1 Ziel der Arbeit

Ziel dieser Diplomarbeit ist es, eine signaturbasierte Methode zu entwickeln, die einen bestimmten Angriff auf VANETs erkennt. Dabei sollen erstmal die wichtigsten Begriffe definiert werden, die zum VANET Gebiet gehören. Dazu gehören auch die verschiedenen Routingsprotokolle, die in VANETs eingesetzt werden können. Diese sollen diskutiert und verglichen werden, um ein geeignetes für VANETs auszusuchen und anzuwenden. Die verschiedenen möglichen Angriffe auf VANETs sollen dann gesammelt, analysiert und nach Gefährlichkeit, Eintrittswahrscheinlichkeit und Risiko kategorisiert werden. Die vorhandenen Methoden für Angriffserkennung sollen zunächst analysiert und anschließend bewertet werden. Anschließend soll ein Intrusion Detection System entwickelt und implementiert werden, das basierend auf Angriffserkennungsalgorithmen automatisch Angreifer detektieren kann. Zum Schluss wird die Funktionalität des entwickelten IDS Anhand von Tests geprüft, um zu untersuchen, ob es die Anforderungen erfüllt.

1.2 Aufbau der Arbeit

Nach einer umfassenden Einleitung im ersten Teil, werden im zweiten Teil der Arbeit die Begriffe Ad-hoc Netze, MANETs und VANETs erläutert. Dabei wird jedes Netz einzeln behandelt und erläutert, wie es sich auf das andere aufbaut. Danach werden die verschiedenen Routingsprotokolle aus dem Bereich der MANETs vorgestellt und analysiert, um darunter die geeigneten für VANETs rauszufiltern. Danach werden erstmal die verschiedenen Angriffsszenarien auf VANETs grob definiert und in passive und aktive Angriffe kategorisiert. Ein modifiziertes OSI-Schichtenmodel wird dann vorgestellt und die möglichen verschiedenen Angriffe auf die zugehörenden Schichten erläutert. Die Analyse der Angriffe wird dazu dienen, eine zusammenfassende Tabelle zu erstellen. Dabei wird zwischen internen und externen Angriffen verglichen sowie die Angriffe nach Intensität, Attraktivität, Eintrittswahrscheinlichkeit und Risiko analysiert und gruppiert. Anschließend werden konkrete Angriffsszenarien auf VANETs dargestellt und erläutert. Hierzu wird ein Szenario ausgesucht, welches eine hohe Eintrittswahrscheinlichkeit besitzt. Dieses Szenario wird dann detailliert analysiert und als Grundlage dienen, um ein Intrusion Detection System zu entwickeln. Dabei werden Randbedingungen gesetzt, damit dieses IDS funktionieren kann. Im dritten Kapitel wird das Konzept erläutert, wobei die erwähnten Randbedingungen beachtet werden. Zum Schluss kommt eine Zusammenfassung und ein Ausblick.

Kapitel 2

Vehicular Ad-hoc Networks

In den Mobilen Ad-hoc Networks werden viele verschiedene Routingprotokolle eingesetzt. Da die große Anzahl dieser Routingprotokolle den Rahmen dieser Arbeit sprengen würde, werden in diesem Kapitel, nach einer Definition von den Begriffen Ad-hoc Network, Mobile Ad-hoc Network und Vehicular Ad-hoc Netwok, paar Routingprotokolle in MANETs vorgestellt und definiert. Unter diesen Routingprotokollen werden diejenigen ausgefiltert, die für VANETs geeignet sind. Anschließend werden die verschiedenen möglichen Angriffe auf VANETs gruppiert und analysiert.

2.1 Ad-hoc Network

Im Jahr 1997 wurde der 802.11 Standard eingeführt [3]. Dieser Standard hat die Funktion, eine drahtlose Kommunikation zwischen Computern zu ermöglichen. Heutzutage wird er unter dem Begriff WLAN[1] verwendet und kann neben dem häufig benutzten Infrastruktur-basierten Modus auch im Ad-hoc Modus verwendet werden [3]. In diesem Fall spricht man von einem Ad-hoc Netzwerk.
Ad-hoc Netze sind kleine lokale Netzwerke, vor allem mit drahtlosen oder nicht dauerhaft verdrahteten Verbindungen, die sich je nach Bedarf bilden und hierfür nicht notwendigerweise eine existierende Infrastruktur, wie Access Points, benötigen [24].

Die wörtliche Übersetzung des lateinischen Ausdrucks „ad hoc“ lautet im übertragenen Sinne „nur für diesen Zweck“ und „vorübergehend“. Dieser Begriff wird auch bei Büro- oder Heimnetzwerken angewandt, die neue Geräte schnell aufnehmen können und dabei beispielsweise Technologien wie Bluetooth oder WLAN nutzen. Verschiedene Computer und andere Geräte können sich über drahtlose Schnittstellen miteinander verbinden.

Bei Ad-hoc Netzen unterscheidet man zwei Arten von Netzwerktopologien.

[1] Wireless Local Area Network

- Single-Hop: Bei Single-Hop Verbindungen stehen die Knoten nahe aneinander und können direkt miteinander kommunizieren, solange sie in der Reichweite von einander sind. Die Grafik in Abbildung 2.1 zeigt zwei Knoten die miteinander über Single Hop kommunizieren.

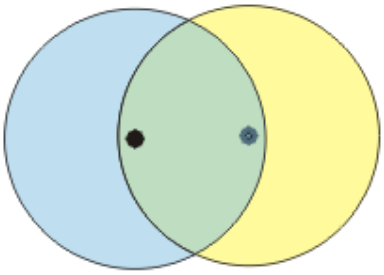

Abbildung 2.1: Beispiel einer direkten Verbindung in einem Ad-hoc Netz

- Multi-Hop: Bei Multi-Hop Verbindungen kann ein Knoten einen weit entfernten Knoten nur über Zwischenknoten erreichen. Diese müssen Routingfunktionalitäten besitzen. Die Grafik in Abbildung 2.2 zeigt Knoten die entweder direkt miteinander oder über Zwischenknoten kommunizieren.

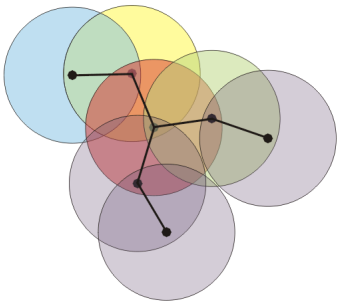

Abbildung 2.2: Beispiel einer indirekten Verbindung zwischen Knoten eines Ad-hoc Netzwerkes

Diese Art von vermaschten Netzwerktopologien bieten viele Vorteile:

- Keine zentrale Verwaltung
- Niedrige Netzwerkkosten
- Gute Lastverteilung

  - Bei Ausfall eines Endgerätes ist evtl. durch Umleitung die Datenkommunikation weiterhin möglich, falls alternative Routen vorhanden sind

Nachteile vermaschter Netzwerktopologien sind allerdings:

- Komplexes Routing nötig
- Jedes Endgerät arbeitet als Router und ist demnach oft aktiv
- Die Endgeräte sollten möglichst eingeschaltet bleiben
- Höherer Stromverbrauch im Endgerät durch Routing
- Keine Kommunikation wenn keine Route vorhanden ist
- Hohes Sicherheitsrisiko durch drahtlose Schnittstellen.

2.1.1 Mobile Ad-hoc Network

Ein mobiles Ad-hoc Netzwerk (MANET) besteht aus mobilen Endgeräten, die mit drahtlosen Kommunikationsgeräten ausgestattet sind. Die Nachrichtenübertragung durch einen Knoten wird von allen Knoten innerhalb seiner Reichweite empfangen. Das geschieht mit Hilfe der Broadcast Eigenschaft in der kabellosen Kommunikation omni-direktionaler Antennen. Wenn zwei Knoten sich nicht in der Reichweite von einander befinden, müssen andere mobile Rechner zwischen ihnen ihre Nachrichten weiterleiten und so eine Netzwerkbrücke zwischen zwei weit entfernten Knoten aufbauen. Durch die Mobilität der drahtlosen Knoten muss jeder mit der Fähigkeit ausgestattet sein autonom zu handeln, oder eine Routingfunktion ohne statisch etablierte Infrastruktur oder zentrale Administration durchzuführen. Dabei kann sich jeder Knoten beliebig bewegen und sich ein- oder ausschalten ohne andere Knoten zu benachrichtigen. Die Mobilität und Autonomie der Netzwerkteilnehmer führen zu einer dynamischen Topologie des Netzes. Nicht nur weil End-Knoten transient sind, sondern auch, weil Zwischenknoten transient sind [1]. Abbildung 2.3 zeigt ein Beispiel für mobile Geräte die miteinander drahtlos kommunizieren.

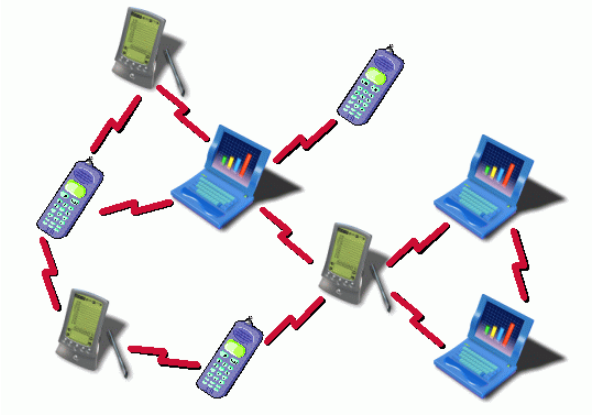

Abbildung 2.3: Beispiel eines MANETs

2.1.2 Vehicular Ad-hoc Network

Nach Plößl [18], ein VANET ist ein Ad-hoc Netz aus verschiedenen Fahrzeugen, die selbständig Funkverbindungen untereinander etablieren. Diese Fahrzeuge bewegen sich in der Regel im Rahmen ihrer Rahmenbedingungen (Straßennetz, Verkehr und Verkehrsregeln). Die starke Eigenschaft dieses Netz liegt darin, dass die Fahrzeuge in der Lage sind sich an die aus den Rahmenbedingungen resultierende Veränderungen der Netztopologie, die häufig und abrupt auftreten können, eigenständig anzupassen.

VANETs kann man in die Kategorie der MANETs unterteilen. Die Knoten können dabei eine sehr große Anzahl haben und bewegen sich evtl. mit sehr hoher Geschwindigkeit. VANETs bilden einen Grundstein für Intelligent Transportation Systems (ITS). Wie die Abbildung 2.4 zeigt können Fahrzeuge untereinander über Inter-Vehicle Communication (IVC) sowie mit Basisstationen am Straßenrand via Roadside-to-Vehicle Communication (RVC) kommunizieren. Das hat dazu geführt, dass die Fahrzeugnetzwerke ein sichereres und effizienteres Fahren durch rechtzeitige Übermittlung von Informationen an die Fahrer und die betroffenen Behörden ermöglichen können.

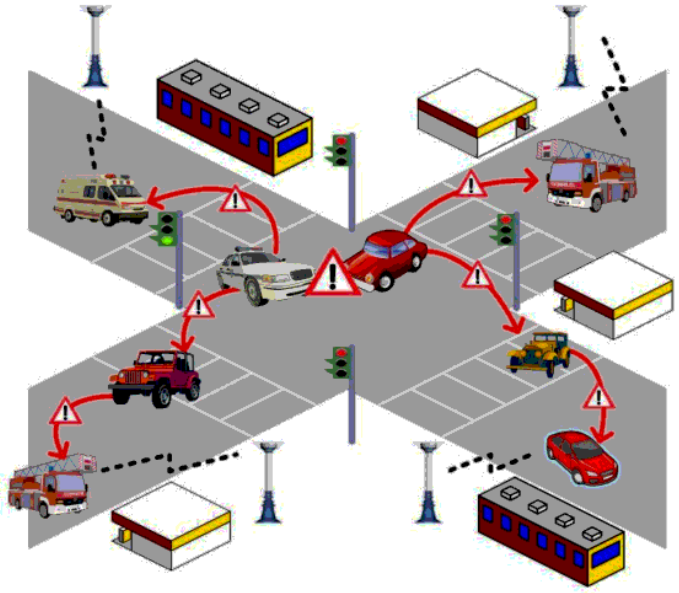

Abbildung 2.4: Kommunikationsbeispiel in einem VANET

Nach Gerla [12] gibt es einige zu beachtende Eigenschaften für VANETs, aus denen sich hohe Anforderungen ergeben:

- `Mobilität:` In früheren Phasen der Computerentwicklung war der Bedarf nach Infrastrukturfreien Netzwerken sehr gering, da die Komponenten sehr groß waren und somit eine feste Position hatten. Mit der Entwicklung leichter Geräte wie z.B. Laptops, PDAs und Smartphones, sowie deren Mobilität, hat sich dieser Zustand geändert.
  Bei Ad-hoc Netzen und besonders bei VANETs soll die Mobilität an erster Stelle gewährleistet werden. Dabei spielt die absolute Position und die relative Position der einzelnen Teilnehmer eine sehr große und wichtige Rolle. Die absolute Position der einzelnen Knoten spielt dabei nur in Bezug auf das umgebende Gelände und die Anbindung der Knoten an existierende andere Netzwerke eine große Rolle.

- `Robustheit:` Wegen technischer Defekte oder Überschreiten der Sendereichweite können manche Knoten unangekündigt ausfallen. Deshalb müssen sich die verwendeten Netzwerkprotokolle darauf einstellen und der Ausfall einzelner Knoten darf keine wesentliche Beeinträchtigung für das Gesamtnetzwerk bedeuten, auch wenn diese Knoten nicht mehr zum Weiterleiten von Nachrichten zur Verfügung stehen.

- `Selbstorganisation`: Das Netzwerk muss ein hohes Maß an Selbstorganisation besitzen. Es muss in der Lage sein, autonom für eine Situation die bestmögliche Konfiguration zu wählen, bei der alle Knoten erreichbar sind. Dies sollte so weit wie möglich unter geringem Ressourcenverbrauch stattfinden.

- `Energieverbrauch`: Energie sparen spielt eine sehr große Rolle bei MANETs, besonders beim Defekt eines Knotens. So wird zum Beispiel das Weiterleiten im Falle eines geringen Batteriestandes unterlassen, um die eigene Erreichbarkeit länger zu erhalten [24]. Bei VANETs werden die Batterien ständig geladen, somit spielt die Energieversorgung keine große Rolle. Das Ausfallen eines Knotens hat keine große Wirkung, wenn die Anzahl der Netzwerkteilnehmer hoch ist. Die Erreichbarkeit anderer Knoten wird in diesem Fall durch Umwege gewährleistet.

- `Skalierbarkeit`: In VANETs kann eine große Anzahl an Geräten verwendet werden. Hierfür sollten dynamische Hierarchien aufgestellt werden können, so dass die Systemkomplexität für einen einzelnen Knoten möglichst gering gehalten werden kann. Wenn zum Beispiel mehrere Knoten gleichzeitig senden, kann es zu Störungen der Signale kommen. In diesem Fall sollten die Protokolle so gewählt werden, dass es trotz Szenarien mit hoher Knotendichte zu möglichst wenigen Kollisionen kommt.

- `Identifizierbarkeit`: Alle Netzteilnehmer müssen ihre eigene Identität besitzen, dies wird mit Adressen gewärleistet. Als Beispiel können die Knoten das Internet Protokoll (IP) benutzen und somit globale Adressen besitzen. Die Nutzung von MAC Adressen ist auch möglich, da diese auch eine eindeutige Identifikation darstellen. Dabei muss aber die Privatsphäre der Knoten gewährleistet werden.

- `Sicherheit`: VANETs basieren auf kabellosen Verbindungen. Diese sind immer größeren Angriffsmöglichkeiten ausgesetzt, als kabelgebundene Verbindungen in Netzwerken. Zum Beispiel besteht immer die Gefahr des Mithörens durch Dritte, da die Anbindung an ein drahtloses Netzwerk wegen dem Broadcast der Daten weniger leicht kontrolliert werden kann. Eckert spricht in [10] je nach verwendeter Sendetechnik und -stärke, und in Abhängigkeit des Geländes und des Empfangsequipments, von möglichen Abhörreichweiten bis zu 1,5 Kilometer.

- `Unsichtbarkeit`: Das System sollte für den Fahrer unsichtbar bleiben. Das heißt es muss in der Lage sein, selbst Gefahrensituationen zu erfassen und

dementsprechend Kommunikation zu anderen Fahrzeuge aufzubauen um die Nachricht zu verteilen, sodass der Fahrer nichts machen muss und sich nur auf das Fahren konzentrieren kann.

2.2 Routing in VANETs

Eine wichtige Funktion ist die Erfassung und Verbreitung von Informationen, wie zum Beispiel die Position und Bewegung benachbarter, bzw. des eigenen Fahrzeugs. Die Position der einzelnen Knoten kann durch ein Positionierungssystem wie GPS (Global Positioning System), GALILEO, oder aus anderen On-Board-Geräten bereitgestellt werden.
Eine grundlegende Eigenschaft der Fahrzeuge ist ihre teilweise festgelegte Mobilität, da sie auf Straßen fahren. Diese Straßen können einfach zugeordnet und digital dargestellt werden. Darüber hinaus können Verkehrsregeln auch elektronisch dargestellt werden.

Das wichtigste Problem, welches mit Fahrzeugnetzwerken verbunden ist, sind die potenziell hohen Geschwindigkeiten mit denen sich die verschiedenen Knoten bewegen. Die hohe Mobilität führt zu einer häufigen und schnellen Änderung der Netzwerktopologie und zu einer weiteren Destabilisierung des Funkkanals.
Daher ist die größte Herausforderung für die Forscher, die im Bereich der Routingprotokolle für VANETs forschen, sich mit Schätzungen über folgende Bedingungen zu befassen:

- Die verfügbare Bandbreite
- Medium Access Control
- Das Problem der verborgenen und ausgesetzten Knoten
- Die hohe Mobilität der einzelnen Knoten
- Unterstützung von heterogenen Fahrzeugen
- Bewegung der Knoten
- Hindernisse usw.

2.2.1 Routingprotokolle

Routing über Ad-hoc-Netzwerke kann allgemein in topologie- und positionsbasierte Ansätze aufgeteilt werden. Topologiebasierte Routingprotokolle hängen von Informationen über bestehenden Verbindungen im Netzwerk ab und nutzen sie für die Weiterleitung der Datenpakete. Sie können weiter unterteilt werden, als proaktive und

reaktive Routingprotokolle [8].

2.2.1.1 Proaktive Verfahren:

Die proaktiven Verfahren sind tabellengesteuert. Dabei besitzt jeder Knoten eine sogenannte Routingtabelle, in der er Informationen über alle Knoten des Netzwerks, wie zum Beispiel Entfernung in Hops, Nachbarknoten, speichert. Die Knoten tauschen dabei regelmäßig in bestimmten Perioden Informationen über den Zustand des Netzwerks mit ihren Nachbarn aus. Auf dieser Weise hat jeder Knoten die Topologieinformationen über das gesamte Netzwerk, unabhängig davon, ob sie für ihn von Bedeutung sind oder nicht. Es ist ein sehr großer Vorteil wenn die Route vor dem Senden nicht zuerst berechnet oder gesucht werden muss. Dennoch hat dieses Verfahren erhebliche Nachteile. Der hohe Ressourcenbedarf und die hohe Auslastung des Netzwerks sind eine große Grundlast. Zwei oft eingesetzte proaktive Routingprotokolle sind das Optimized Link State Routing (OLSR) und das Destination Sequenced Distance Vector Protocol (DSDV).

2.2.1.2 Reaktive Verfahren:

Bei den reaktiven Verfahren wird im Gegensatz zu den proaktiven Verfahren der Weg im Netzwerk erst dann berechnet, wenn ein Quellknoten auch wirklich Pakete versenden möchte. Dabei werden die Knoten, die nicht auf dem Weg zu dem Zielknoten liegen, auch nicht mit Routinginformationen bezüglich fremder Wege belastet. Sie nehmen daher auch nicht an dem Austausch von Routinginformationen teil. Daneben müssen die Knoten jedoch auch ihre Nachbarn kennen, um beispielsweise die Änderung der direkten Netzwerktopologie festzustellen. Die reaktiven Routingprotokolle nutzen im Vergleich zu den proaktiven Protokollen das Netzwerk und die Ressourcen weitaus effizienter aus. Ein wesentlicher Nachteil ist allerdings die Verzögerung, die zwischen Kommunikationsbedarf und Beginn des Datentransfers auftritt, da diese Zeit für das Suchen der Route benötigt wird. Zwei oft eingesetzte typische Vertreter sind das Ad-hoc On-Demand Distance Vector Routing (AODV) und das Dynamic Source Routing (DSR) Protokoll.

2.2.1.3 Hybride Verfahren:

Hybride Verfahren kombinieren lokales proaktives und globales reaktives Routing, um ein höheres Maß an Effizienz und Skalierbarkeit zu erreichen. Dies bedeutet, dass bei Knoten innerhalb der Reichweite das proaktive Verfahren, und bei Knoten außerhalb der Reichweite das reaktive Verfahren angewandt wird.

Positionsbasierte Routingalgorithmen haben einige Einschränkungen der topologiebasierte Ansätze überwunden, indem sie sich zusätzlichen Informationen zu Nutze machen. Positionsbasierte Protokolle verlangen, dass die physikalische Position der

teilnehmenden Knoten jeder Zeit bekannt ist. Jeder der Knoten bestimmt seine eigene Position durch den Einsatz des Global Positioning System (GPS) oder einer anderen Art der Positionierungsbestimmung [13]. Der Absender verwendet einen Lokalisierungsservice, um die Position der Zielknoten zu bestimmen und um diese im Zieladressebereich des Pakets zu integrieren. Hier basiert der Routingprozess an jedem Knoten auf der Position der Empfängerknoten, die im Paket gespeichert sind, und die Position der Nachbarknoten, die die Aufgabe haben, das Packet weiterzuleiten. Positionsbasiertes Routing erfordert keine Errichtung oder Aufrechterhaltung von Routen, welches aber in der Regel mit Hilfe einer zusätzlichen Hardware gewährleistet wird. Als eine weitere Verbesserung unterstützt das positionsbasierte Routing das Versenden von Paketen an alle Knoten in einem bestimmten geographischen Gebiet auf natürliche Weise. Dieses Verfahren heißt Geocasting.

2.2.2 Vergleich der Routingprotokolle

In dem vorherigen Abschnitt wurden ein paar Routingprotokolle näher erläutert. In diesem Abschnitt wird nach [24] zuerst eine Definition für die verschiedenen Routingsprotokolle gegeben und anschließend eine Bewertung und einen Vergleich zwischen den verschiedenen Routingsprotokollen dargestellt. Die Bewertung und der Vergleich dienen dazu, um zu prüfen, ob diese Routingprotokolle für VANETs geeignet sind.

2.2.2.1 Destination-Sequenced Distance-Vector Protocol:

Das Destination-Sequenced Distance-Vector Protocol (DSDV) [11] ist ein proaktives Hop-by-Hop-Distanz-Vektor-Routing Protokoll, in dem jeder Knoten Routingupdates in regelmäßigen Abständen Broadcastet. Jeder Knoten besitzt im Netzwerk eine Routingtabelle für alle möglichen Ziele innerhalb des Netzes und die Anzahl der Zwischenknoten bis zu jedem Ziel. Dabei wird jeder Eintrag mit einer von dem Zielknoten zugeordnete Sequenznummer markiert. Die Sequenznummern werden dazu verwendet, um veraltete Routen von neuen zu unterscheiden, was zur Vermeidung von Routingschleifen führt. Routingtabellenupdates werden regelmäßig im gesamten Netzwerk übertragen, um die Konsistenz in den Tabellen zu erhalten.

Um den potenziell großen Netzwerk Updateverkehr zu verringern, können zwei mögliche Arten von Paketen eingesetzt werden. Full dumps bzw. kleine inkrementelle Pakete.
Full dump Pakete führen alle verfügbaren Routinginformationen und können mehrere Netzwerkprotokoll Dateneinheiten (NPDUs) erfordern. Diese Pakete werden während der Bewegungen der Knoten nicht mehr so oft übertragen. Kleinere inkrementelle Pakete werden verwendet, um nur die Informationen, die sich seit dem letzten full dump geändert haben, zu übertragen. Jeder dieser Broadcasts soll in einer Standard

großen NPDU passen. Dabei wird der generierte Verkehr verringert.

Eine neu gebroadcastete Route enthält die Adresse des Zielknotens, die Anzahl der Zwischenknoten um den Zielknoten zu erreichen, die Sequenznummer von den empfangenen Informationen, sowie eine neue Sequenznummer, die für den Broadcast reserviert ist. Die Route, die mit der neuesten Sequenznummer gekennzeichnet ist, wird immer verwendet. Für den Fall, dass zwei Updates die gleiche Sequenznummer haben, wird die Route mit der kleineren Metrik zur Optimierung (Verkürzung) des Pfads verwendet. Knoten beobachten die Regelungszeit der Routen oder die gewichtete durchschnittliche Zeit, in der Routen zu einem Zielknoten schwanken können, bevor die Route mit der besten Metrik erhalten wird. Durch die Verzögerung des Broadcasts von Routingupdates, durch die Länge der Regelungszeit, können Knoten den Datenverkehr reduzieren.

2.2.2.2 Optimized Link State Routing Protocol (OLSR):

Das Optimized Link State Routing (OLSR) Protocol [11] ist ein proaktives Protokoll für mobile Ad-hoc Netze, das auf dem Link-State-Algorithmus basiert. In einem reinen Link-State-Protocol, werden die ganzen Verbindungen zu den benachbarten Knoten deklariert und im gesamten Netz geflutet. OLSR reduziert erstens die Größe der Kontrollpakete. Er deklariert nur eine Teilmenge der Verbindungen zwischen seinen Nachbarn, die als seine Multipoint-Relay-Selektoren dienen, anstatt die ganzen Verbindungen zu deklarieren. Zweitens minimiert es die Flutung dieser Verkehrskontrolle, indem es nur die ausgewählten Knoten, die so genannten Multipoint-Relais, zur Verbreitung ihrer Nachrichten im gesamten Netzwerk benutzen. Als Reaktion auf das Ausfallen oder Hinzufügen von einem Link, generiert das Protokoll keine extra Kontrollnachrichten, außer den normalen periodischen Kontrollnachrichten. Das Protokoll hält die Routen für alle Ziele im Netzwerk. Daher ist es für die Verkehrsmuster mit einer großen Teilmenge von Knoten, die miteinander kommunizieren, von Vorteil, und so ändern sich die Paare <source, destination> ständig mit der Zeit.

Das Protokoll eignet sich besonders für große und dichte Netze, da die Optimierung über die Multipoint-Relays gut in diesem Kontext funktioniert. OLSR ist entworfen worden, um völlig dezentral zu arbeiten (ohne zentrale Einheit). Es bedarf keine sichere Übertragung seiner Kontrollnachrichten. Jeder Knoten sendet seine Kontrollnachrichten periodisch, und kann daher den Verlust von einigen Paketen wegen Kollision oder Übertragungsprobleme ertragen. Darüber hinaus braucht OLSR keine Übertragung seiner Pakete in der richtigen Reihenfolge. Jede Kontrollnachricht enthält eine Sequenznummer, so kann der Empfänger die Pakete in der richtigen Reihenfolge bringen. Das OLSR-Protocol führt Hop-by-Hop-Routing. Das bedeutet dass jeder Knoten seine neuesten Informationen benutzt, um Pakete zu verschicken. Wenn ein Knoten sich bewegt, können seine Pakete erfolgreich an ihn geschickt werden. Das geschieht wenn seine Geschwindigkeit so angepasst ist dass seine Bewegung

von seinen Nachbarknoten verfolgt werden kann.

2.2.2.3 Ad-hoc On-demand Distance Vector:

AODV [25] basiert auf den Distance Vector Algorithm von Bellman-Ford und wurde so konzipiert dass es in mobilen Umgebungen funktioniert. Dabei wird die begrenzte Bandbreite und Batterielebenszeit betrachtet. Außerdem gehört es zu den On-demand Verfahren, d.h. es erstellt die Route zu einem Knoten nur wenn jemand ein Paket zu diesem Knoten schicken möchte.
Die Route wird auf folgende Art und Weise berechnet. Man beachte das Ad-hoc Netz in der Abbildung 2.5, dabei möchte ein Prozess im Knoten A ein Paket zum Knoten I schicken. Der AODV Algorithmus führt eine Routingtabelle in jedem Knoten. Diese enthält Informationen über das Ziel, und welche Nachbarn in Frage kommen, wenn man eine Nachricht zu diesem Ziel schicken möchte. Bei diesem Beispiel, wenn Knoten A in seiner Routingtabelle sucht und keinen Eintrag für Knoten I findet, muss er die Route zu Knoten I erkunden, und das macht dieser Algorithmus auf Anfrage (engl. „On-demand“).

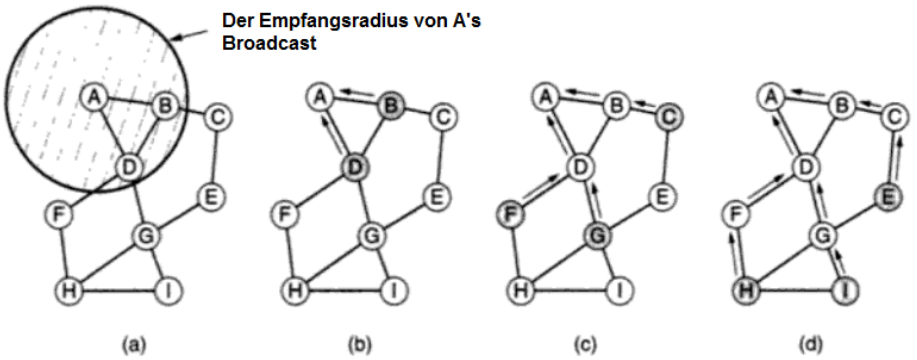

Abbildung 2.5: Verschiedenen Etapen zur Erkundung der Route zwischen den Knoten A und I [3]

Um Knoten I zu lokalisieren, erzeugt Knoten A ein ROUTE REQUEST Paket und broadcastet es. Knoten B und Knoten D bekommen das Paket, da sie die direkten Nachbarn von Knoten A sind. Anschließend sucht der Empfängerknoten nach dem Ziel in seiner Routingtabelle. Wenn die Route zum Ziel bekannt ist, wird ein ROUTE REPLY Paket erzeugt und an Knoten A zurückgesandt, um ihm zu zeigen, wie er das Ziel erreichen kann. Wie dieses Paket aussieht, wird in Grafik 2.6 dargestellt.

Sender Adresse	Anfragen ID	Empfänger Adresse	Sender Sequence#	Empfänger Sequence#	Knoten-anzahl

Abbildung 2.6: Aufbau eines ROUTE REQUEST Paket in AODV

Wenn der Empfängerknoten keinen Eintrag vom Ziel in seiner Routingtabelle hat, dann broadcastet er das ROUTE REQUEST Paket weiter und trägt die Daten vom Paket in seiner reverse Routingtabelle als neuen Eintrag ein. In diesem Fall wissen Knoten B und Knoten D nicht wo Knoten I ist, und so erzeugen sie einen reverse Routeneintrag der auf Knoten A zeigt. Knoten B broadcastet das ROUTE REQUEST Paket weiter zu Knoten C und Knoten D, und Knoten D broadcastet es weiter zu Knoten F, zu Knoten G und Knoten B. Das Paket wird von Knoten B und Knoten D gelöscht weil es als Duplikat behandelt wird. Knoten G bekommt das Paket und weiß wo Knoten I liegt (weil Knoten I sein direkter Nachbar ist). Als Antwort auf den ROUTE REQUEST erzeugt Knoten I ein ROUTE REPLY Paket. Dieses Paket sieht aus wie in Grafik 2.7 dargestellt.

Sender Adresse	Empfänger Adresse	Empfänger Sequence #	Knoten-Anzahl	Lebens-dauer

Abbildung 2.7: Aufbau eines ROUTE REPLAY Paket in AODV

Die Source Adresse, Destination Adresse und die Anzahl der Knoten werden von der empfangenen Anfrage kopiert. Danach wird das Paket per unicast an den Knoten versendet, der das ROUTE REQUEST Paket geschickt hat, und zwar an den Knoten G. Dann wird die reverse Route zum Knoten D und anschließend zum Knoten A gefolgt. Bei jedem Knoten wird die Anzahl der Knoten (hops count) inkrementiert, so kann man wissen wie weit der Knoten A vom Knoten I entfernt ist.

2.2.2.4 Dynamic Source Routing (DSR):

DSR [5] ist ein innovatives Konzept für das Routing in MANETs, in dem Knoten über festgelegte Routen kommunizieren. Diese Routen werden erst dann gesucht wenn der Knoten einen konkreten Verbindungswunsch hat.
Eine bemerkenswerte Optimierung von DSR ist, dass bei den Knoten keine Routingtabellen geführt werden, wenn sie Pakete verschicken oder weiterleiten möchten. Stattdessen wird eine Liste aller Zieladressen in jedem Paket gepackt wie in Abbildung 2.8 dargestellt ist. Das hat den Vorteil, dass die Notwendigkeit der weiterleitenden Knoten ihre Daten zu synchronisieren und immer aktuelle Routingtabellen zu ha-

ben, verringert wird. Das führt dazu dass die Übertragung von Routingdaten wesentlich verringert wird. Somit können die weiterleitenden Knoten einfacher aufgebaut werden (geringere Hardwareanforderung) und müssen auch keinen großen Speicher für die Routingtabellen besitzen.
Bei DSR entdecken die Knoten die Routen in dem sie auch ein Route Request Paket broadcasten. Das Paket enthält die Adresse vom Ziel- und Senderknoten und eine Identifikationsnummer. Jeder Knoten der das Route Request Paket empfängt prüft, ob er die Route zum Ziel kennt. Wenn nicht, dann fügt er seine eigene Adresse zur Menge der Senderknoten und schickt das Paket weiter.
Wenn das Route Request Paket den Zielknoten oder einen Zwischenknoten erreicht, der die Route zum Zielknoten kennt, wird ein Route Replay Paket erzeugt, und an den Absender geschickt. Die Zwischenknoten, die das Paket durchlaufen muss, sind dann im Paket gespeichert, wie auf der Abbildung 2.8 dargestellt ist.

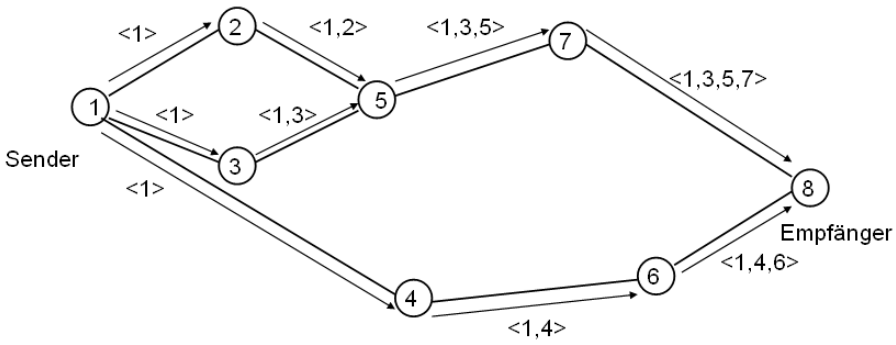

Abbildung 2.8: Routenberechnung in DSR

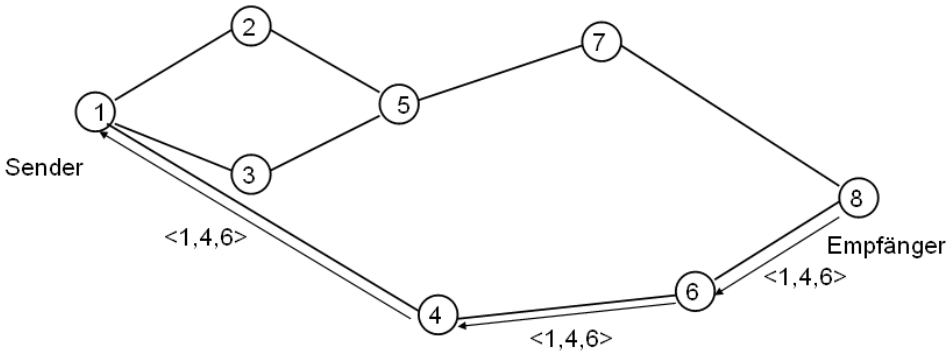

Abbildung 2.9: Propagation von Response Replay Paket in DSR

Teilnehmer belauschen den lokalen Netzwerkverkehr, um weitere Routinginformationen zu bekommen. Dies ist möglich, da in jeder übertragenen Nachricht eine Liste

mit Adressen anderer Knoten steckt. Darüber hinaus erkennen sie Routenanfragen, Routenfehler und Informationen anderer Rechner, welche sie später einmal selbst verwenden können.
DSR arbeitet beim Auffinden von Routen fast gleich wie AODV. In kleinen wenig ausgelasteten Funknetzwerken haben die beiden Protokolle ein ähnliches Performanzverhalten. Ist die Auslastung dagegen höher, verursacht DSR nur etwa ein d rittel des Datenaufkommens. Die DSR-Knoten bekommen sehr viele Informationen durch das Belauschen des Netzwerkverkehrs und müssen so das Netz mit weniger eigenen Routenwünschen belasten. Das ist auf der einen Seite sehr vorteilhaft aber auf der anderen Seite verursacht das Belauschen das Problem, dass viele Informationen gesammelt werden. Dabei müssen ältere Informationen erkannt und aus den Routingtabellen entfernt werden, und das kann durchaus sehr aufwändig sein.

2.2.2.5 Analyse

Broadcasten in Netzwerken ist eine verbreitete Operation zum Lösen vieler Aufgaben. In VANETs wird durch die hohe Mobilität und Geschwindigkeit der einzelnen Knoten die Verwendung von Broadcasten noch stärker ausgeprägt sein, als das in MANETs der Fall ist. Wie bereits in den vorherigen Routingprotokolle erwähnt wurde, z.B zum Pflegen von Routingtabellen oder um einen Überblick über die Nachbarknoten zu behalten.
Bei VANETs entsteht das Problem, da sich Funksignale überlappen und überlagern können, wird einfaches Broadcasting erstens eine teuere Operation u.a. mit vielen redundanten Datensendungen, Paketkollisionen und hoher Netzlast. Zweitens kann einfaches Broadcasting, wie in dem nächsten Abschnitt erklärt wird, eine gute Basis für Angriffe auf Routing darstellen.

Außerdem wurden die vorherigen Routingprotokolle für VANETs für ungeeignet eingestuft, denn das Hauptproblem bei diesen Protokolle ist die Instabilität, und die traditionelle knotenzentrierte Sicht der Routen. Das heißt, eine etablierte Route (eine feste Abfolge von Knoten, die sich zwischen Quelle und Ziel befinden) kann nicht mehr gewährleistet werden und führt zu häufigen Defekten der Routen wegen der hohen Mobilität in VANETs. Folglich werden viele Pakete verworfen und die Gesamtkosten aufgrund von Reparaturen der Routen ist wesentlich höher, was zu einer niedrigen Senderate und hoher Übertragungsverzögerungen führen kann.

Als Lösung für das Problem wird das Geocasting oder geographische Routing vorgeschlagen, da es zum einen nahezu jedes Protokoll in irgendeiner Weise Broadcasting in Form von Flooding einsetzt, und dabei unnötiges Broadcasting vermieden wird. Außerdem hat das Geocasting den Vorteil, dass jeder Knoten, der schneller zum Ziel führen könnte, für das Forwarding benutzt werden kann [22].

2.2.2.6 Geocasting:

In einem Geocast sendet ein Knoten eine Nachricht an eine Empfängergruppe, die durch ihre geografische Position bestimmt ist (Zielgruppe). Die Idee des Geocastings wird zuerst von Jiang und Camp [15] erwähnt und dort als Erweiterung des Internets vorgeschlagen, und nicht als Anwendung für mobile Ad-hoc Netzwerke. Es wird dort beschrieben, wie man Nachrichten an Knoten innerhalb eines Polygons oder Kreises senden kann. Dieses Polygon wird durch Breiten und Längenangaben definiert.
Grundsätzlich hat Geocasting eine große Ähnlichkeit zu einem Multicast, nur dass anstatt einer Multicastadresse eine geografische Region adressiert wird. Man tritt einer Geocast Gruppe bei, indem man ein bestimmtes Gebiet betritt, und verlässt die Gruppe, indem man das Gebiet wieder verlässt. Im Gegensatz zu einem Multicast, in dem man einer Multicast-Gruppe durch abbonieren einer bestimmten Multicast-Adresse betritt.
Geocasting setzt voraus, dass die Knoten des Netzwerkes ihre eigene Position kennen.
Die in den letzten Jahren vorgestellten Geocasting Protokolle lassen sich, wie die Abbildung 2.10 zeigt, grob in zwei Kategorien einteilen: floodingbasierte und routenerzeugende Protokolle [21].

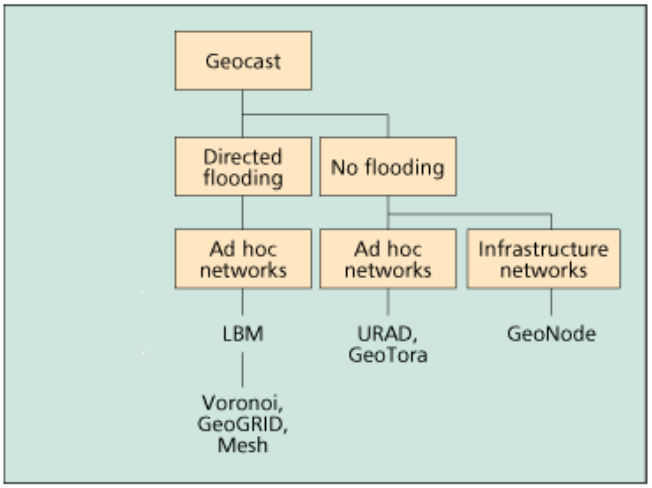

Abbildung 2.10: Geocast Taxonomy

1. **Floodingbasierte Protokolle:**

   Floodingbasierte Protokolle nutzen Flooding, oder eine Variante davon, um die Nachrichten ins Zielgebiet zu übermitteln. Das Ziel ist dabei, eine möglichst hohe Zuverlässigkeit der Datenübertragung zu erreichen. Zuverlässigkeit bedeutet, möglichst viele der potenziellen Zielknoten sollen erreicht werden. Ins-

besondere wird keine Route zum Zielgebiet durch Routingprotokolle erstellt. Vorteile des floodingbasierten Ansatzes sind u.a dass kein Overhead für das Erstellen und Verwalten von Routen anfällt und gleichzeitig eine hohe Zuverlässigkeit erreicht wird. Der größte Nachteil liegt sicherlich in der meist erheblich höheren Netzlast, die diese Protokolle erzeugen.
Von den floodingbasierten Protokollen werden Location Based Multicast (LBM) und GeoGRID vorgestellt [21].

- **Location Based Multicast**
  LBM basiert grundsätzlich auf Flooding, vermeidet aber Flooding im gesamten Netz durch Definition einer Flooding Zone. Außerhalb dieser Flooding Zone wird das Paket einfach verworfen.
  Zwei Schemen zur Verbesserung des Multicast Flooding mit Positionsinformation werden vorgestellt.
  Beim einfachen Flooding, wie oben beschrieben, wird eine so genannte Forwarding Zone definiert. In der Regel ist es das kleinste Rechteck welches sowohl das Zielgebiet, als auch den Sendeknoten umfasst. Ein Zwischenknoten kann das Paket nur weiterleiten wenn er zur Forwarding Zone gehört. Durch das Vergrößern der Forwarding Zone wird auch die Wahrscheinlichkeit des Paketempfangs durch die Zielknoten größer, wobei aber auch der Aufwand größer wird. Die Forwarding Zone ist in jedem Geocast Paket enthalten, womit festgestellt wird, ob der Knoten zu dieser Zone gehört oder nicht.

  Beim zweiten Schema, wird die Entscheidung, ob ein Knoten ein Geocasting-Paket weiterleitet (floodet) nicht anhand einer flooding Zone gefällt, sondern anhand eines Distanzvektors zum rechnerischen Mittelpunkt.
  Dabei werden Pakete nicht nur von Knoten weitergeleitet, die in der Nähe der Zielregion sind, sondern wird auch von bestimmten Knoten weitergeleitet. Diese Knoten sind nicht mehr als eine bestimmte Entfernung weiter vom Ziel entfernt, als der Knoten, von welchem das Datenpaket empfangen wurde.

  Abbildung 2.11 zeigt den Unterschied zwischen den zwei Methoden.

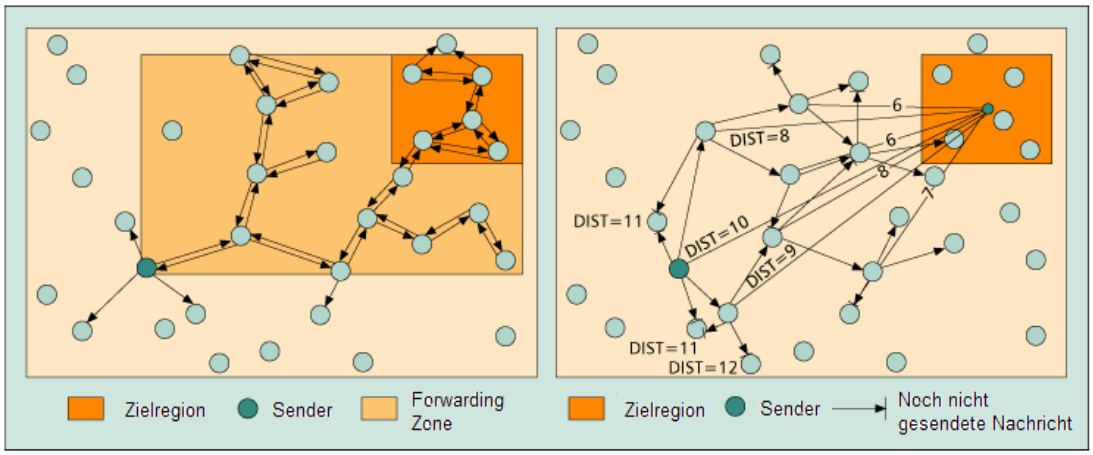

Abbildung 2.11: Beispiel eines Geocasts mit LBM: a) Quadratische Forwarding Zone, b) Distance-basierte Forwarding Zone

Diese beiden Methoden reduzieren wesentlich die Netzlast, dafür sind sie in ihrer Zuverlässigkeit nur geringfügig besser als pures Flooding. Deshalb wird GeoGRID als Optimierung für flooding basierte Protokolle vorgeschlagen.

- **GeoGRID:** GeoGRID unterteilt das Netzwerk in logischen Netzen (Kästchen) mit einem einzigen gewählten Gateway in jeder Partition. Nur Gateways, die anderen Knoten von ineffizienten flooding entlasten, leiten Pakete weiter.
  Bei GeoGRID ist es sehr wichtig, eine vernünftige Wahl des Gateways zu treffen. Optimal wäre dabei ein Gateway der möglichst nahe am Mittelpunkt des Kästchens ist, damit die Nachbarkästchen besser abgedeckt werden.
  Vor dem Senden eines Geocast Pakets werden keine Geocast spezifischen Routen festgelegt. Außerhalb der Flooding Region wird ein empfangenes Paket einfach verworfen. Wenn ein Gateway innerhalb der forwarding Region ein Paket enthält, broadcastet er das Paket an seine Nachbargatways weiter sofern das Paket nicht schonmal broadcastet wurde.

  Neben dieser flooding-basierten GeoGRID haben die Autoren einen Ticket-basierten GeoGRID entworfen. Dabei verschickt ein Gateway ein Paket innerhalb der forwarding Region. Um die Anzahl der Gatways zu reduzieren, schickt ein Gateway das Paket nur an maximal 3 benachbarte Gateways weiter, um das überall-Flooding zu begrenzen. Durch die Vergabe einer begrenzten Anzahl von Tickets wird das Flooding begrenzt.
  Wenn ein Gateway nicht innerhalb der Zielregion liegt, selektiert er drei benachbarte Gateways, dessen Grids sich in der Nähe der Zielregion und

in der Forwarding Zone befinden. Das Geocast Paket wird dann an die ausgewählten Gateways übermittelt und die Tickets werden gleichmäßig unter ihnen aufgeteilt. Wenn nur ein Ticket übrigbleibt, wird ein Paket nur an genau einen Nachbar geschickt.
GeoGRID ist somit eine Weiterentwicklung der LBM-Idee und diesem Protokoll gegenüber vorzuziehen. Dabei empfiehlt es sich vor allem die Verwendung von Ticket basiertem GeoGRID, da dieses gemessen an dem Trade-off zwischen Zuverlässigkeit und Netzlast die bessere Wahl darstellt. Abbildung 2.12 zeigt ein Beispiel eines Geocasts mit GeoGRID mit beiden Ansätze.

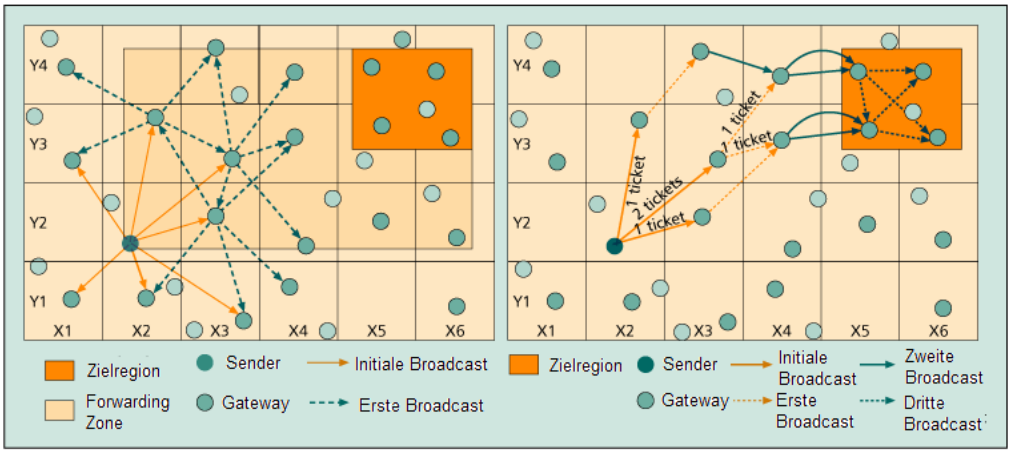

Abbildung 2.12: Beispiel eines Geocasts mit GeoGRID a) flooding basiert, b) Ticket basiert

2. **Routenerzeugende Protokolle:**
   Routenerzeugende Protokolle erstellen zuerst eine Route zum Zielgebiet und nutzen diese dann zum Übermitteln der Daten in die Zielregion. Der größte Vorteil dieses Ansatzes liegt darin, dass wesentlich weniger Netzlast anfällt. Dafür muss aber der erhöhte Aufwand für das Erstellen und Pflegen der Routen in Kauf genommen werden.

   Von den routenerzeugenden Protokollen wird GeoTORA vorgestellt.
   GeoTORA hält für jede Geocast Gruppe einen gerichteten azyklischen Graphen mit allen Netzwerkknoten, die in die Routing Richtung zur Zielregion zeigen.
   GeoTORA ist ein weiteres Geocast Protokoll für Ad-hoc Netze. Es basiert auf TORA (Temporally Ordered Routing Algorithm), was ein unicast Routing Algorithmus für Ad-hoc Netze darstellt. In TORA wird ein Directed Acyclic Graph (DAG) für jedes Ziel erzeugt. Der DAG zeigt auf jeden Knoten in Richtung der Zielregion, und das kann für das Verschicken eines Pakets genutzt

werden.

Bevor man den GeoTORA Algorithmus definiert, sollte man zunächst den Begriff Anycast definieren. Anycast ist ein Verfahren zur Aktualisierung der Routing-Tabellen in IPv6. Anycast sendet eine Nachricht an den nächsten Knoten innerhalb einer Gruppe. Der Knoten wiederum schickt sie an seinen nächsten Knoten weiter.

Der GeoTORA Algorithmus basiert auf einer Anycast Änderung von TORA. Zuerst wird ein DAG für jede Anycastgruppe gehalten. Zwischen den Mitgliedern der Anycastgruppe gibt es keine Richtung in dem DAG. Die Richtungen innerhalb der DAG sind durch die Zuweisung einer Höhe zu jedem Knoten definiert. Ein Paket wird immer einem Nachbar mit einer geringeren Höhe weitergeleitet. Grundsätzlich ist die Höhe die Entfernung zu der Zielregion. Zwei Mitgliedern der Geocast Gruppe wird die Höhe 0 zugeordnet.
Das initiale DAG wird wie folgt erstellt. Wenn ein Knoten eine Route zu einer Geocast Gruppe erfordert, sendet er eine Anfrage an alle Nachbarn. Die Anfrage wird immer wieder verschickt, bis ein Mitglied der DAG gefunden wird. Dabei sind die Nachbarknoten der Zielregion bereits Mitglieder. Nach Eingang einer Abfrage reagiert ein Mitglied der DAG mit dem Broadcasten seiner Höhe an seine Nachbarn. Ein Knoten, der auf eine Verbindung mit dem DAG wartet, setzt seine Höhe auf die minimale Höhe von allen seinen Nachbarn, dann erhöht er sie um eins und broadcastet sie.

Da Knoten sich ständig bewegen, ist der DAG nicht stabil. Jedoch wird das Beibehalten der DAG ohne Flooding gewährleistet. Damit gehört GeoTORA zu den Routingprotokollen ohne Flooding. Er reagiert auf Veränderungen in der DAG, wenn ein Knoten keine ausgehende Links mehr hat. Dann wird die Richtung einer oder mehrerer Verknüpfungen geändert, die sogenannte Link Umkehrung. Nachbarknoten sind nur von dieser Maßnahme betroffen, wenn sich ihr letzter ausgehender Link auf eine eingehende Verbindung verändert hat. Was bedeutet, dass sie das Linkumkehrungsprozess wiederholen müssen. Wenn die gerichteten Links vom DAG ein Anycast Paket übermitteln, wird es schließlich zu einem zufälligen Knoten der Anycast Gruppe geliefert. Geocasting mit diesem Algorithmus arbeitet wie in Abbildung 2.13 gezeigt. Er beginnt mit einem Anycast zu einem beliebigen Mitglied der Geocast Gruppe mit dem oben beschriebenen Ansatz. Nach Erhalt des ersten Geocast Pakets flooden die Mitglieder der Geocastgruppe das Paket innerhalb der Geocastregion.

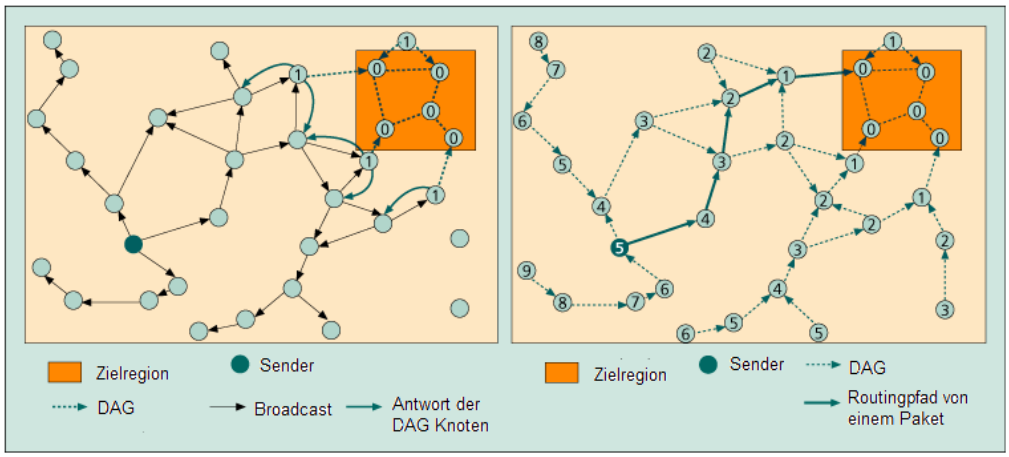

Abbildung 2.13: Beispiel eines Geocasts mit TORA a) Erzeugung des DAGs, b) Forwarding von einem Geocast Paket

2.2.3 Routingprotokolle in VANETs

Wegen den unterschiedlichen Anforderungen der Anwendungsszenarien werden bei VANETs, wie in Abbildung 2.14 gezeigt, mehrere Routingverfahren eingesetzt [23]. Diese Verfahren basieren auf allen schon vorgestellten Geocast Ansätze.

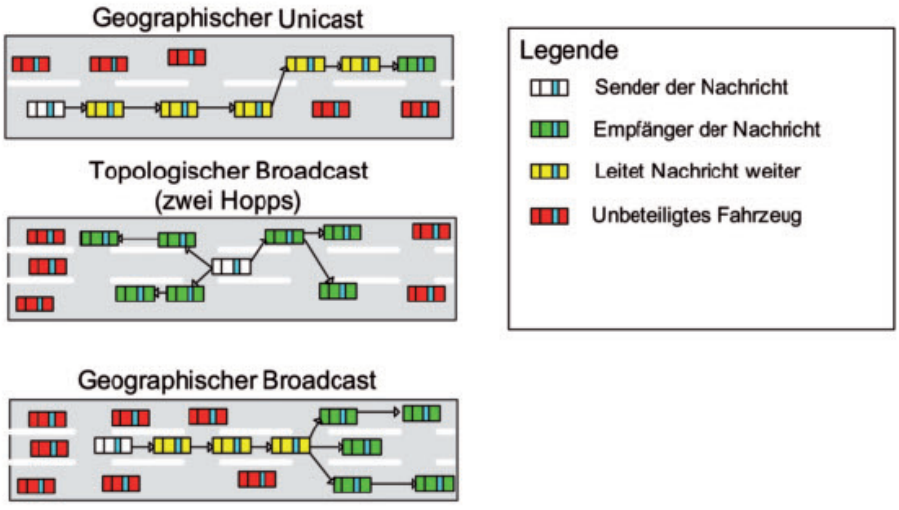

Abbildung 2.14: Weiterleitung einer Nachricht im VANET

- **Geographischer Unicast:** Dieses Verfahren wird benutzt, wenn ein Fahrzeug mit einem einzelnen weit entfernten Fahrzeug kommunizieren möchte. Das Weiterleiten der Pakete übernehmen dann Zwischenfahrzeuge über den so genannten Multihop Verfahren. Jeder Zwischenknoten, der die Nachricht bekommen hat, leitet sie zu dem Nachbarknoten, welcher der geographischen Position des Zielknotens am nächsten liegt.
  Bei diesem Verfahren sind die lokalen Maxima ein potentielles Problem. Das entsteht, wenn das Paket an einen Knoten weitergeleitet wird, welcher doch keine Möglichkeit hat dieses Paket in die Richtung des Zielknotens zu weiterzusenden, da in dieser Richtung kein weiterer Nachbarknoten in Empfangsreichweite existieren.
  Das Problem wird gelöst, indem das Paket an den nächsten Nachbarknoten in einem bestimmten Winkelbereich weitergeleitet wird.

- **Topologischer Broadcast:** Dieses Verfahren wird eingesetzt, wenn Fahrzeuge in der Nachbarschaft bestimmte Informationen verbreiten wollen. Die Nachricht wird mittels topologischen Broadcast versendet, d.h. die Nachricht wird zu allen Nachbarknoten innerhalb einer bestimmten Anzahl von Hops weitergeleitet.

- **Geographischer Broadcast:** Dieses Verfahren wird eingesetzt, wenn Fahrzeuge in einer bestimmten Zone über ein Ereignis informiert werden sollen. Zum Beispiel könnten das Fahrzeuge in derselben oder in der Gegenfahrrichtung sein. Dabei kann der geographische Broadcast in zwei verschiedenen Fällen angewendet werden. Der erste Fall ist die Benachrichtigung aller Fahrzeuge die sich direkt in der Senderichtung befinden. In diesem Fall wird unmittelbar ein Broadcast in Senderichtung durchgeführt. Der zweite Fall ist die Benachrichtigung von Fahrzeugen, die sich in Senderichtung innerhalb von einem Zielgebiet befinden. Wenn dieses Gebiet nicht unmittelbar an den Sender grenzt, wird die Nachricht zunächst mittels Unicast in das Zielgebiet gebracht und erst dort per Broadcast an alle Fahrzeuge im Zielgebiet weitergeleitet.

2.3 Sicherheitsanforderungen

Bevor über die Sicherheit diskutiert werden kann, muss man zwischen 2 Kategorien der Sicherheit unterscheiden: Funktionssicherheit (engl. *safety*) und Informationssicherheit (engl. *security*).
Nach Eckert [10] versteht man unter der Funktionssicherheit eines Systems die Eigenschaft, dass die realisierte Ist-Funktionalität der Komponenten mit der spezifizierten Soll-Funktionalität übereinstimmt. Ein funktionssicheres System nimmt keine funktional unzulässigen Zustände an. Anders formuliert versteht man unter der Funktionssicherheit eines Systems, dass es unter allen normalen Betriebsbedingungen funktioniert.

Die Informationssicherheit ist die Eigenschaft eines funktionssicheren Systems nur solche Systemzustände anzunehmen, die zu keiner unautorisierten Informationsveränderung oder -gewinnung führen. Man möchte im IT-Bereich Gefährdungen für die Daten des Systems durch beabsichtige Angriffe abwehren.
Da sich diese Arbeit mit der Kommunikationssicherheit im VANET beschäftigt, liegt der Schwerpunkt auf der Informationssicherheit. Die Funktionssicherheit wird in diesem Abschnitt nur eingeschränkt betrachtet.
Diese Arbeit folgt dem in der Literatur gängigen Verständnis, nachdem alle Schutzziele zum Schutz gegen beabsichtigte Angriffe vollständig durch die folgenden grundlegenden Schutzziele dargestellt werden können

2.3.1 Authentizität

Unter der Authentizität eines Objekts bzw. Subjekts versteht man die Echtheit und Glaubwürdigkeit des Objekts bzw. Subjekts, die anhand einer eindeutigen Identität und charakteristischen Eigenschaften überprüfbar ist. Das heißt ein Absender einer Nachricht soll zweifelsfrei zu erkennen sein.

2.3.2 Datenintegrität

Ein System kann die Datenintegrität gewährleisten wenn es Subjekten nicht möglich ist, die zu schützenden Daten unautorisiert und unbemerkt zu manipulieren. In VANETs können absichtlich gefälschte oder inkorrekte Daten weitreichende Folgen nach sich ziehen. Das Spektrum reicht von Verwirrung und Irrführung bis hin zu lebensgefährlichen Verkehrssituationen. Im folgenden sind einige Beispiel dafür aufgeführt.

- Ein Anwohner einer vielbefahrener Strasse möchte das Verkehrsaufkommen vor seinem Haus reduzieren. Er könnte falsche Staumeldungen erzeugen und so einen Großteil der Verkehrsteilnehmer dazu bringen eine längere Alternativroute zu wählen.

- Ein Besitzer einer Autobahnraststätte könnte ebenfalls falsche Staumeldungen verschicken und darauf hoffen, dass viele Fahrer das Ende des vermeintlichen Staus auf seiner Raststätte abwarten und ihm zusätzlichen Umsatz bescheren.

- Ein Angreifer gibt sich als Polizist aus und verschickt Anhalteanweisungen an seine Opfer.

- Ein Angreifer provoziert Unfälle, in dem er zwei Fahrzeugen an einer Kreuzung jeweils freie Fahrt signalisiert, obwohl sich die Fahrspuren kreuzen.

2.3.3 Informationsvertraulichkeit

Ein System kann die Informationsvertraulichkeit gewährleisten, wenn es keine unautorisierte Informationsgewinnung ermöglicht. Dabei werden folgende Anforderungen gewährleistet:

- Schutz der Kommunikationsinhalte: Außenstehende, die nicht an einer Kommunikation beteiligt sind, dürfen den Inhalt der ausgetauschten Nachrichten nicht erfahren. Um das zu gewährleisten werden in der Praxis meist symmetrische oder asymmetrische kryptographische Verschlüsselungssysteme verwendet.

- Sicherheit vor unbefugtem Gerätezugriff: Im VANET ist auch die Sicherheit vor unbefungtem Gerätezugriff wichtig. Dieser Schutz ist, verglichen mit der Situation in anderen Netzen, relativ schwer zu erreichen, da Fahrzeuge und stationäre Knoten oft in periodischen oder für Angreifer vorhersehbaren Zeitabständen den Zugriffsbereich beziehungsweise dessen Funkreichweite verlassen.

- Schutz der Privatsphäre: Der Schutz der Privatsphäre zielt hauptsächlich auf die Geheimhaltung der Kommunikationsumstände bzw. -Teilnehmer ab.

2.3.4 Verfügbarkeit

Verfügbarkeit bedeutet, dass Daten und Informationen dort und dann zugänglich sind, wo und wann sie von Berechtigten gebraucht werden. Wenn Dienste und Informationen, auf die sich Fahrer in VANETs gelernt haben zu verlassen, nicht verfügbar sind, dann könnten Gefahrensituationen entstehen. Das kann dazu führen, dass dies nicht nur den Ruf dieses speziellen Dienstes und des gesamten VANETs nachhaltig schädigen, sondern bei verkehrssicherheitskritischen Anwendungen sogar Fahrzeuge und Leben der Verkehrsteilnehmer gefährden.

2.3.5 Verbindlichkeit

Ein System kann die Verbindlichkeit bzw. Zuordenbarkeit einer Menge von Aktionen gewährleisten, wenn es nicht möglich ist, dass ein Subjekt im Nachhinein die Durchführung einer solchen Aktion abstreiten kann. Mit anderen Worten, Verbindlichkeit

bezeichnet die Möglichkeit, einer IT-Transaktion während und nach der Durchführung zweifelsfrei und gegebenenfalls gerichtsverwertbar den Durchführenden zuordnen zu können. Dies kann z.B. durch die Nutzung von qualifizierten und sicheren Aufbewahrung von Logdateien erreicht werden. Die Dauer der Zuordenbarkeit hängt von der Aufbewahrung der Logdateien ab und wird durch das Datenschutzrecht reglementiert.

2.3.6 Anonymisierung und Pseudomisierung

Anonymisierung und Pseudomisierung sind sehr wichtige Maßnahmen der IT-Sicherheit.

- Unter der Anonymisierung versteht man das Verändern personenbezogener Daten in der Art, dass Einzelangaben über persönliche oder sachliche Verhältnisse nicht mehr oder nur mit einem unverhältnismäßig großen Aufwand an Zeit, Kosten und Arbeitskraft einer bestimmten oder bestimmbaren natürlichen Person zugeordnet werden können.

- Die Pseudomisierung ist eine schwache Form der Anonymisierung. Es handelt sich dabei um das Verändern personenbezogener Daten durch eine Zuordnungsvorschrift. Die Einzelangaben über persönliche oder sachliche Verhältnisse ohne Kenntnis oder Nutzung der Zuordnungsvorschrift können nach Veränderungen nicht mehr einer natürlichen Person zugeordnet werden.

2.4 Angriffsszenarien

In dem folgenden Abschnitt werden die zwei Begriffe Bedrohung und Angriff definiert und als Einleitung für die Angriffszenarien diskutiert.
Nach Eckert [10] zielt eine Bedrohung (engl. threat) des Systems darauf ab, eine oder mehrere Schwachstellen oder Verwundbarkeiten auszunutzen, um einen Verlust der Datenintegrität, der Informationsvertraulichkeit oder der Verfügbarkeit zu erreichen, oder um die Authentizität von Subjekten zu gefährden.

Unter einem Angriff versteht man einen nicht autorisierten Zugriff bzw. einen nicht autorisierten Zugriffsversuch auf das System. Man unterscheidet dabei zwischen passiven und aktiven Angriffen. Passive Angriffe betreffen die unautorisierte Informationsgewinnung und zielen auf den Verlust der Vertraulichkeit ab. Aktive Angriffe betreffen die unautorisierte Modifikation von Datenobjekten und richten sich somit gegen die Datenintegrität oder Verfügbarkeit eines Systems.

Bei VANETs können Angreifer verschiedene Absichten haben, wenn sie einen Angriff ausüben möchten. Sie möchten entweder individuelle Vorteile gewinnen, den

Angriff kommerziell nutzen oder einfach aus Spaß das System stören.
Wenn ein Angreifer einen Angriff auf ein VANET ausübt, versucht er in der Regel eine gewisse Sicherheitsanfälligkeit auszunutzen, oder einfach nur das schwächste Glied der Sicherheitsmechanismen zu attackieren. Ähnlich wie die Angriffe auf normale IT-Systeme kann ein Angriff auf VANETs bei der Entwicklung, der Produktion oder während des eigentlichen Einsatzes durchgeführt werden. Im Rahmen dieser Arbeit werden nur Angriffe betrachtet die innerhalb des Einsatzes von VANETs ausgeübt werden können.

Die meisten Angriffe auf VANETs werden aus dem MANET Bereich abgeleitet, da VANETs als eine Teilmenge von MANETs angesehen werden können. Die Angriffe werden in zwei Kategorien unterteilt [14]. Dies wird in den folgenden Abschnitten beschrieben.

2.4.1 Passive Angriffe

Passive Angriffe sind äußerst schwer zu identifizieren, da die Kommunikation und der Nachrichteninhalt dabei nicht beeinflusst werden. Sie bedrohen in erster Linie die Vertraulichkeit der Daten. Bei dieser Art von Angriffe wird hauptsächlich auf die Informationsbeschaffung abgezielt. Der Angreifer sendet selbst keinerlei Daten. Er verhält sich sehr passiv, indem er lediglich den Datenverkehr anderer Teilnehmer belauscht, ohne diesen aktiv zu verändern. Damit erhält er wichtige Vermittlungs- und Benutzerinformationen. Das Ganze hilft ihm dabei zum Beispiel Verkehsflussanalyse des Netzes durchzuführen und somit einen Einblick über die Struktur des Netzes zu bekommen. Sämtliche abgefangenen Informationen können ihm als Ausgangsbasis für einen aktiven Angriff dienen.

2.4.2 Aktive Angriffe

Bei den aktiven Angriffen werden die Angreifer in Gegensatz zu den passiven Angriffen sehr aktiv, indem sie in die Kommunikation eingreifen um Daten, IT-Systeme oder Benutzer zu manipulieren. Diese Art von Angriffe beinhalten die nicht autorisierte Modifikation von Daten und richten sich somit in erster Linie gegen die Datenintegrität und die Verfügbarkeit. Wenn der Angreifer seinen Angriff erfolgreich durchgeführt hat, dann hat er direkten Zugang zu fremden Betriebsmitteln und kann diese aktiv missbrauchen. So kann er durch Wiederholung, Verzögerung, Einfügung, Modifikation und Löschung bestimmter Daten eine falsche Identität vortäuschen und eventuell Rechte und Attribute modifizieren.

2.4.3 Angriffszuordnung zum modifizierten OSI-Schichtenmodel

Das OSI-7-Schichtenmodell ist ein Referenzmodell für Kommunikationssysteme. OSI bedeutet Open System Interconnection (Offenes System für Kommunikationsverbin-

dungen), das von der ISO (International Organization for Standardization) als Grundlage für die Bildung von Kommunikationsstandards entworfen und standardisiert wurde. Das OSI Schichtenmodell oder OSI Referenzmodell basiert auf dem DoD Schichtenmodell (Department of Defense), auf dem das Internet basiert. Im Vergleich zum DoD Schichtenmodell ist das OSI-Schichtenmodell feiner aufgegliedert.
Das OSI-Schichtenmodell besteht aus 7 Schichten. Jede Schicht ist in Arbeitseinheiten eingeteilt. Die Arbeitseinheiten haben ihre Funktion innerhalb der Schicht oder kooperieren mit den Arbeitseinheiten der benachbarten Schichten. Damit diese Arbeitseinheiten miteinander arbeiten können müssen sie sich an ein Regelwerk halten. Dieses Regelwerk nennt man Protokoll.
Da das OSI-Referenzmodell [7] reger Kritik unterliegt, werden in dieser Arbeit die unterschiedlichen Angriffsmöglichkeiten nach einem modifizierten Hybridmodell klassifiziert. Die Schichten dieses Models werden in der folgenden Grafik 2.15 dargestellt.

Abbildung 2.15: Modifiziertes Hybridmodell eines OSI-Schichtenreferenz-Modells

2.4.3.1 Bitübertragungsschicht

Die Bitübertragungsschicht besteht bei VANETs meistens aus Funkverbindungen zwischen Knoten. Da diese die Grundlage darstellen, werden im folgenden die wichtigsten Angriffsmöglichkeiten auf Bitübertragungsschicht in Bezug auf VANETs behandelt.

- Eavesdropping [16] gehört zu der Kategorie der passiven Angriffe. Der Angreifer macht es sich zu Nutzen, dass sich bei Funkverbindungen das Signal in alle Richtungen gleichmäßig ausbreitet. Somit kann er mit einer Antenne dieses Funksignal empfangen und die Daten mitschneiden. Er muss sich hierfür nur in Empfangsreichweite des sendenden Knotens befinden und muss dabei keinerlei Daten senden.

- Jamming [16] gehört zu der Kategorie der aktiven Angriffe. Dabei greift der Angreifer aktiv an indem er die Funkverbindung zwischen bestimmten Netzwerkteilnehmer des VANETs stört. Das geschieht mit Hilfe von Störsender. Diese überdecken die Frequenzbänder mit starkem Rauschen, so dass die Endgeräte keine gültigen Signale mehr von anderen Endgeräten empfangen können.

2.4.3.2 Sicherungsschicht

Die Sicherungsschicht ist verantwortlich für die zuverlässige und fehlerfreie Kommunikation zwischen direkten Nachbarn in VANETs. Angriffe auf die Sicherungsschicht beziehen sich daher meist auf die direkte Nachbarschaft eines Angreifers und nicht auf die übrigen Knoten des VANETs. Allerdings ist zu berücksichtigen, dass ein erfolgreicher Angriff auf die Sicherungsschicht als Vorstufe zu weiteren Angriffen dienen kann. Die wichtigsten Angriffe auf dieser Schicht werden im Folgenden diskutiert.

- Bei der Datenverkehrsanalyse muss der Angreifer Teil des Netzes sein. Auch wenn der Datenaustausch eventuell verschlüsselt ist, profitiert er von der Eigenschaft, dass Kontrollnachrichten des MAC Protokolls jeden teilnehmenden Knoten über eine MAC Adresse identifizieren. Diese kann er dann sammeln und mit der Menge und der Frequenz der gesendeten Daten in Verbindung bringen um detaillierte Profile darüber zu erstellen, wer mit wem im Netzwerk kommuniziert.

- Beim Manipulieren des Network Allocation Vectors handelt es sich um einen internen aktiven Angriff. Am Anfang einer Übertragung reserviert der Sender den Funkkanal mit einer Request-To-Send (RTS) Nachricht und beginnt erst dann zu senden. Diese RTS Nachricht enthält einige Informationen für alle mithörenden Teilnehmer des Netzes zwecks Kollisionsvermeidung. Der Sender teilt seinen Nachbarn mit, wie lange er das Medium für das Senden seines Frames benötigen wird. Diese Informationen bezeichnet man als den Network Allocation Vector (NAV). Ein Angreifer kann den NAV einfach modifizieren, um für jedes von ihm gesendeten Frame immer die maximale mögliche Länge anzugeben. Das führt dazu dass alle seine Nachbarn lange Zeit passiv bleiben und ihr Datendurchsatz somit reduziert wird.

- Der Angreifer kann eine kleine Back-Off Zeit wählen, um immer sehr kleine Werte für seine Back-Off Zeit zu bekommen, und nicht zufällig wie es normalerweise ist. Das führt zum Folgenden: Wenn der Angreifer über einen längeren Zeitraum Daten mit einem Nachbarn austauscht, wählen die anderen Teilnehmer in seinem Umkreis immer höhere Backoff Zeiten. Somit sind sie gezwungen länger warten zu müssen. Ein solches Verhalten könnte zum Denial Of Service führen [29].

2.4.3.3 Vermittlungsschicht

Die dritte Ebene des Hybrid Modells befasst sich mit der Problematik, wie sich die Pakete von der Quelle zum Ziel übertragen lassen. Es geht dabei primär um die Übertragung und die Wegfindung von einem Endpunkt zum Anderen. Die Vermittlungsschicht muss über die Topologie des Netzwerkes kennen, um geeignete Pfade wählen zu können. Dieses Routing bildet die Grundlage für den Datenaustausch zwischen den einzelnen Knoten. Die klassischen Netze benutzen dafür spezielle Geräte (Router), die diese Aufgabe der Ende-zu-Ende-Kommunikation wahrnehmen. Routern setzen Routingprotokolle ein, wodurch unterschiedliche Paketgrößen oder heterogene Protokolle miteinander kommunizieren können. Der Endanwender hat im Allgemeinen keinen Zugriff auf diese Geräte und kann daher auch die Wegfindung selbst nicht aktiv beeinflussen.

In VANETs sieht dieses Szenario ganz anders aus. Im Gegensatz zu den klassischen Netzen sind die einzelnen Teilnehmerknoten des Netzes gezwungen selbst das Routing zu übernehmen. Jeder Knoten wird selbst zu einem Router.
Die unterschiedlichen Ansätze im Aufbau der Routingprotokolle stellen zwar eine erste Unterscheidung dar, sind aber für Angriffe unerheblich, da diese meist auf alle Verfahrensarten anwendbar sind. Im Folgenden werden Angriffe auf die Vermittlungsschicht erläutert und diskutiert.

- Das Ziel der häufigsten Flooding Angriffe [17] ist die Beeinflussung von Routinginformationen und der Kommunikation zwischen zwei Knoten. Durch Fälschung, Veränderung oder wieder neu einspielen von Routinginformationen, kann der Angreifer das Routing wesentlich beeinflussen.

  Durch HELLO Pakete wird es für ein Knoten möglich, einem bestehenden Netzwerk beizutreten. Wenn ein Knoten ein HALLO Paket empfängt, stellt er automatisch fest, dass der Senderknoten ein Nachbarknoten ist. So kann ein Angreifer allen Knoten im Netzwerk vortäuschen, dass alle anderen Knoten seine Nachbarn sind. Um den Angriff erfolgreich auszuüben, muss der Angreifer die gefälschten HELLO Pakete im Netz broadcasten, dafür benutzt er seine überlegene Sendeleistung.

  Ein Angreifer kann andere Knoten dazu bringen ihre Nachrichten ins Leere zu senden. Dazu spielt er ihnen eine gute Anbindung an virtuellen Knoten vor, die überhaupt nicht existieren. Der Angreifer kann alte HELLO-Pakete, die er zuvor gesammelt hat, wieder neu einspielen. Er muss gar nicht erst in der Lage sein gültige Pakete zu erzeugen, um diesen Angriff durchzuführen. Im Gegensatz zu seinem Namen, benutzt der HELLO Flooding Angriff einen einzelnen Hop broadcast, um eine Nachricht an eine Vielzahl von Empfängern zu übertragen. Es gibt viele böswillige Routingangriffe, die auf die Routing Discovery

oder Routing Maintenance Phase abzielen. Beispiele solcher Angriffe, die in der Routing Discovery Phase zum Tragen kommen können, sind:

- HELLO flooding
- ACK flooding
- RREQ flooding
- RREP flooding

Dabei laufen die flooding Angriffe alle nach dem gleichen Schema ab. Wie bereits bei den Hello Flooding Angriffe beschrieben. Anstatt HELLO-Pakete zu senden werden beispielsweise ACK- oder RREP-Pakete verwendet.

- Beim Sybil Angriff [9] versucht ein Knoten, im Netzwerk mehrere Identitäten anzunehmen.

  Um diesen Angriff auszuüben kann der Angreifer zwei verschiedener Methoden folgen:

  - Er kann eine neue Identität generieren und diese zusätzlich zu seiner echten vortäuschen.
  - Oder er kann eine existierende echte Identität eines anderen Knotens verwenden.

  Der Angreifer kann den Sybil Angriff als Basis für andere weitere Angriffe z.B. einen Blackhole Angriff verwenden. Das fällt ihm besonders leicht wenn er mehrere Identitäten besitzt. So kann er unter Umstände verstärkt im Netzwerk agieren, ohne dabei entdeckt zu werden. Das führt dazu dass er die weiteren vorgesehenen Angriffe ganz einfach verschleiern kann. Ein Angreifer kann beispielsweise versuchen, für jede existierende Route in einem Netzwerk die Identität eines beteiligten Knotens anzunehmen. Dann könnte er alle Pakete abfangen, verändern oder löschen und ggf. weitere Angriffe starten.

- Ziel des Rushing Angriffs [20] ist einen wesentlich höheren Einfluss auf die Routenbildung nehmen zu können. Das kann ein Angreifer schaffen indem er versucht, Nachrichten möglichst schnell weiterzuleiten, damit sie bei der Routenfindung anderen Nachrichten zuvorkommen. Das funktioniert besonders gut, da viele Routingprotokolle über Sicherheitsmechanismen gegen Duplikate verfügen, so dass nur das jeweils zuerst eintreffende Datenpaket ausgewertet wird und alle weiteren verworfen werden.

  Wenn ein Angreifer es geschafft hat, rechtzeitig ein gefälschtes Paket in das Netzwerk einzuschleusen, bevor die korrekten Pakete anderer Knoten ihr Ziel erreichen, so kann er dafür sorgen, dass nur sein gefälschtes Paket akzeptiert wird und alle folgenden echten Pakete verworfen werden. Damit kann der Angreifer das ausnutzen, um weitere Angriffe auszuüben. Er kann z.B. vorzutäuschen, dass er selbst auf der besten Route zu einem bestimmten Ziel liegt.

Ein Angreifer kann sich auch Möglichkeiten der niedrigeren Netzwerkschichten bedienen, um einen Rushing Angriff erfolgreich durchführen zu können. So kann er beispielsweise bestimmte Regeln ignorieren, die ihn normalerweise dazu zwingen, eine Nachricht nicht sofort zu verschicken, sondern eine gewisse Zeitspanne zu warten.

- Knoten eines Netzwerks, die über eine zusätzliche Direktverbindung verfügen, können sich einfach verbünden, und so einen Wormhole Angriff [28] ausüben, wie in Abbildung 2.16 dargestellt ist. Das führt dazu, dass der Datenverkehr umgeleitet wird. Die beiden Knoten müssen dazu außerhalb der normalen Netzwerkkommunikation einen zusätzlichen Kanal etablieren, der ihnen als Tunnel dient. Diese Abkürzung wird in Anlehnung an ein hypothetisches physikalisches Phänomen, als sog. Wurmloch (engl. wormhole), bezeichnet. Die beiden Knoten X und X' behaupten, dass sie direkte Nachbarn seien und daher über eine gute und schnelle Verbindung zum jeweils anderen Knoten und seinen Nachbarn verfügen. Da tatsächlich keine Pakete verloren gehen, sind Wurmlöcher schwer zu entdecken.

  Ein Wurmloch ist für das Netzwerk zunächst nicht unbedingt nur negativ, denn eine solche Abkürzung kann zu einer Entlastung des Netzwerks führen und zu einer geringeren Paketlaufzeit auf den Routen, die das Wurmloch enthalten. Ein Angreifer erreicht damit, dass seine Routen im Netzwerk attraktiver erscheinen und somit also auch mehr Daten zum Beispiel von A nach B über sie geroutet werden. Wie die schon beschriebenen Angriffe kann auch der Wormhole Angriff als Grundlage für weitere Angriffe genutzt werden.

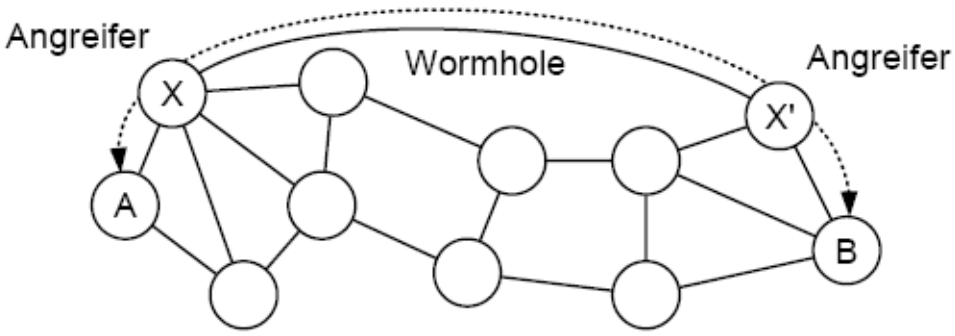

Abbildung 2.16: Datenverkehr bei einem Wormhole Angriff

- Die Idee die hinter einem Blackhole Angriff [2] steckt ist, gezielt falsche Routen zu erzeugen, so dass Pakete nicht mehr ihren eigentlichen Zielknoten D zugestellt werden, sondern stattdessen verloren gehen oder bei einem Angreifer landen. Es wird also so etwas wie ein schwarzes Loch (engl. black hole) gebildet, welches Pakete verschluckt. Abbildung 2.17 zeigt im linken Teil einen

exemplarisch normalen Datenverkehr, der über benachbarte Knoten zu Knoten D weitergeleitet wird. Im rechten Teil der Abbildung sind die Auswirkungen eines erfolgreichen Angriffs zu sehen. Bestimmte Nachrichten erreichen nicht ihr eigentliches Ziel, sondern werden vom Angreifer abgefangen.

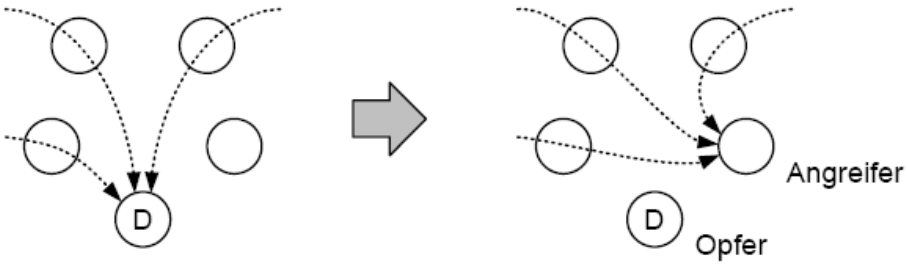

Abbildung 2.17: Datenverkehr vor und während eines Blackhole-Angriffs

In der Phase der Routenfindung bzw. bei Updates der Routinginformationen, verbreitet der Angreifer gefälschte Routinginformationen so dass er selbst Teil möglichst vieler gültiger Routen des Netzwerks wird. Wie bereits bei den vorher beschriebenen Angriffen kann ein Blackhole Angriff für einen Angreifer auch als Grundlage für die Durchführung weiterer Angriffe dienen.

- Bei den Selective-Forwarding Angriffen [17] kann ein Angreifer entscheiden, ob er empfangene Datenpakete weiterleitet oder nicht. Er kann sogar alle Pakete unterdrücken oder verwerfen. Bei diesem Angriff ist die Chance einen bösartigen Knoten zu entdecken sehr gering, besonders wenn er sich bereits auf einen Datenpfad befindet und die wichtigen Pakete selektiert, die er unterdrücken oder manipulieren möchte.

- Bei einem Spoofing Angriff übernimmt ein Angreifer die Identität eines Opfers A, und so kann er einen Man-in-the-Middle Angriff ausüben. Er gibt sich als Knoten A aus, und kann so aktive oder passive Angriffe ausüben.
  Um diesen Angriff erfolgreich auszuüben befindet sich der Angreiferknoten X zwischen Opferknoten A und B, wie in der Grafik 2.18 dargestellt ist. Er gibt sich für Knoten A gegenüber B, und für Knoten B gegenüber A aus. Somit kann er mühelos Pakete von beiden Opfern bekommen.

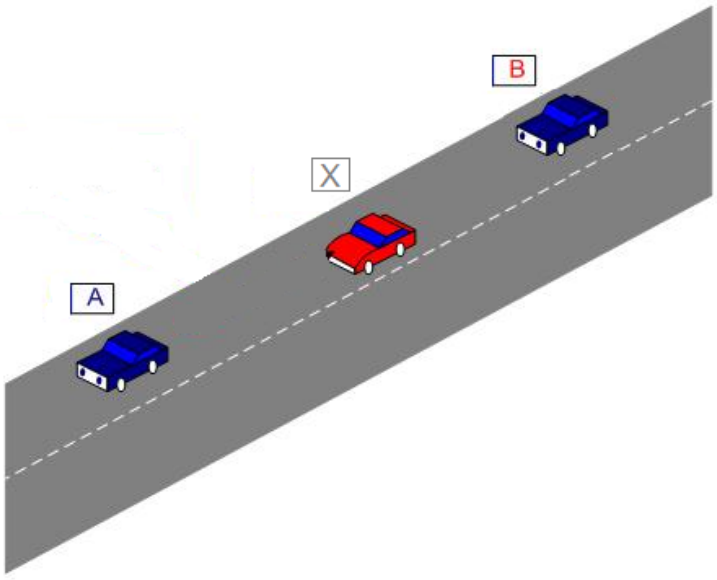

Abbildung 2.18: Ein passiver Angreifer beim Ausüben eines Spoofing Angriffs

- Beim Angriff auf Geocasting kann ein Angreifer einer von den vorher definierten Angriffe als Basis verwenden, um das Geocasting anzugreifen. So kann ein Angreifer zum Beispiel den Selective Forwarding Angriff ausüben und somit nur bestimmte Pakete weiterleiten, und wichtige Informationen unterdrücken oder verändern, besonders das manche Pakete nur per Einzelhop weitergeleitet werden. Anderfalls kann er auch den Spoofing Angriff ausüben und sich in einer bestimmten Gruppe von Fahrzeugen einschleusen (z.B. Firmenfahrzeuge). So kann er mühelos wichtige Informationen abhören, manipulieren oder unterdrücken.

2.4.3.4 Transportschicht

Die Transportschicht dient dazu, zwischen zwei Knoten an beliebigen Stellen im VANET eine verlässliche Verbindung aufzubauen um Daten auszutauschen. Hierbei wird von der Vemittlungsschicht abstrahiert.
In VANETs werden auf der Transportschicht häufig bekannte Protokolle wie das Transmission Control Protocol (TCP) oder das User Datagram Protocol (UDP) eingesetzt. TCP wird vor allem bei Komfort Diensten angewendet, zum Beispiel bei den positionsbasierten Diensten (engl. Location Based Services, LBS) wie Hotel oder Restaurantfinder, bei mobilen Internetverbindung oder bei mobilen Zahlverfahren. UDP kann für kritische Situationen angewendet werden, um zum Beispiel Stau- oder Unfallmeldung abzusetzen.

Da die große Anzahl von Angriffen auf TCP und UDP den Rahmen dieser Arbeit

sprengen würde, werden im Folgenden zwei wichtige Beispiele für Angriffe auf TCP ausgewählt, da diese Angriffe in VANETs realistisch umgesetzt werden könnten.
Der erste dieser TCP-Angriffe ist der sogenannte SYN Flooding Angriff. Hierfür ist es notwendig, den Three-Way-Handshake des Transmission Control Protocols zu erläutern. Der Three-Way Handschake wird durchgeführt, um eine Verbindung zwischen zwei Knoten aufzubauen.

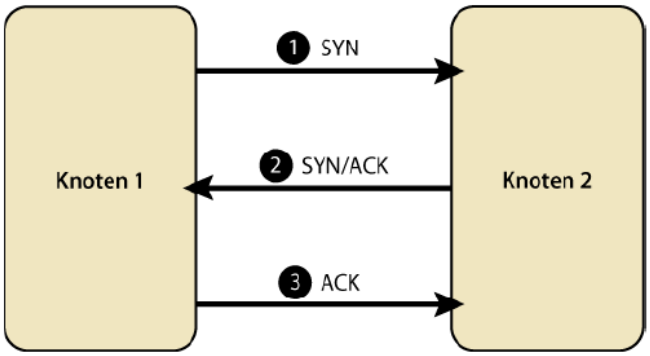

Abbildung 2.19: TCP Tree-Way Handshake

Die Abbildung 2.19 zeigt den schematischen Ablauf des Three-Way Handschakes. Knoten 1 startet den Aufbau einer Verbindung zu Knoten 2, indem er ein SYN (engl. synchronize) Paket an Knoten 2 sendet. Sobald Knoten 2 dieses Paket empfangen hat, legt er in seinen Zustandstabellen einen Eintrag für die jetzt halb geöffnete Verbindung an und sendet seinerseits ein SYN/ACK Paket an Knoten 1. Sobald Knoten 1 nun das 2. Paket erhalten hat, geht er von einer vollständig aufgebauten Verbindung aus und sendet Paket 3 an Knoten 2. Sobald dieser Knoten das Paket erhalten hat gilt auch für ihn die Verbindung als aufgebaut und er verschiebt die Zustandsinformationen für diese Verbindung aus der Tabelle für Halb-offene Verbindungen in die Tabelle für geöffnete Verbindungen.
Folgende zwei Angriffe werden behandelt:

- TCP SYN Flooding: Das Ziel dabei ist den begrenzten Hauptspeicher der mobilen Geräte zum überlauf zu bringen. Dabei sendet ein Angreifer ein SYN Packet zum Aufbau einer neuen Verbindung, wobei die Antwort des Opfers ignoriert wird. Das Opfer hat einen Eintrag für eine halb offene Verbindung in seiner Zustandstabelle aufgenommen, wobei die Verbindung nie komplett aufgebaut wird.

- TCP Session Hijacking: Dabei versucht ein Angreifer die bestehende Session eines anderen Knoten zu übernehmen. Der Angreifer muss nur Kenntnis über

die letzte vom Knoten X (Opfer) verwendete Sequenznummer erlangen und so den anderen Knoten gefälschte Pakete schicken, als ob er Knoten X wäre.

In VANETs wird für verkehrssicherheitsrelevante Nachrichten das UDP Protokoll eingesetzt. Dieses ist zwar verbindungslos und dadurch unzuverlässig aber spart viel Zeit und Aufwand bei kritischen Situationen. Außerdem gibt UDP keine Garantie auf die Zustellung der Pakete oder dass diese in der richtigen Reihenfolge ankommen. Eine Anwendung die UDP nutzt, muss daher gegenüber verlorengegangenen und unsortierten Paketen unempfindlich sein oder selbst entsprechende Korrekturmaßnahmen beinhalten.

Bei UDP werden sogenannte Ports angewendet, damit die versendeten Daten das richtige Programm auf dem Zielrechner erreichen. Außerdem bietet UDP die Möglichkeit einer Integritätsüberprüfung an, indem eine Prüfsumme mitgesendet wird. Dadurch kann eine fehlerhafte Übertragung erkannt werden.
Da die Bestätigung des Empfangs aller gesendeten Nachrichten das Netz unnötig belasten würde, wird in VANETs ein verbindungsloses, UDP ähnliches Protokoll auf der Transportschicht verwendet. Um die Zuverlässigkeit von UDP zu erhöhen werden Teile der Überlastungsüberwachung aus dem Datagram Congestion Control Protocol (DCCP) übernommen. Die eigentliche Kontrolle der Übermittlung soll dabei in die Anwendungsschicht verlagert werden.

2.4.3.5 Anwendungsschicht

Die fünfte Ebene des hybriden Referenzmodells trägt die etwas missverständliche Bezeichnung Anwendungsschicht (engl. Application Layer), da es sich um Protokolle handelt, welche die eigentlichen Anwendungen unterstützen.
Angriffe auf der Anwendungsschicht gehören zu der Kategorie der aktiven Angriffe. Durch die systemübergreifenden Protokolle wie HTTP, SMTP, FTP, usw. sind Angriffe mehr oder weniger die gleichen wie bei klassischen drahtgebundenen Netzen und zielen gegen die Vertraulichkeit und Integrität der Daten ab.
Als spezieller Angriff auf die Anwendungsschicht kommt das Einspielen falscher Informationen an erster Stelle. Im Folgenden werden einige Beispiele aufgeführt:

- Vortäuschen von Gefahren oder Behinderungen z.B. Stau.
- Bewegung eines Fahrzeugs verfolgen um Fahrzeugprofile zu erzeugen.
- Identität eines Einsatzfahrzeuges übernehmen.
- Falsche Straßen- oder Verkehrsinformationen melden.
- Identität einer stationären Einrichtung übernehmen um vertrauliche Daten empfangen zu können.

2.4.4 Angriffe auf VANETs

Tabelle 2.1 gibt eine Übersicht über die verschiedenen Angriffe auf dem schon eingeführten Hybridmodel. Diese Tabelle dient dazu, den verschiedenen Angriffen nach Risiko zu klassifizieren, so kann später ein Angriff ausgesucht werden, der einen hohen Risikowert besitzt, um das dazugehörige Intrusion Detection System zu entwickeln. In dieser Arbeit wird angenommen, dass fast alle Nachrichten verbindungslos verschickt werden. Die Angriffe werden nach den folgenden Klassen klassifiziert.

Schutzbedarf: Aufteilung nach Schutzbedarfsklassen. Für manche Systeme besteht kein besonders hoher Schutzbedarf. Das betrifft die Schutzziele und Akteure.
Schutzklasse 1 (niedrig): Ein Verstoß gegen Gesetze oder die Unversehrtheit von Personen ist nicht betroffen. Kunden könnten verärgert werden und die finanzielle Verluste sind akzeptabel.
Schutzklasse 2 (mittel): Personen könnten zum Teil gesundheitlich betroffen sein oder Dienste werden nicht mehr nutzbar. Finanzieller Schaden und Kunden werden in nicht tolerablerweise beeinträchtigt.
Schutzklasse 3 (hoch): Die Privatsphäre und Gesetze werden ignoriert, Personen werden geschädigt oder der finanzielle Schaden ist nicht tolerierbar.

Betroffene Schutzziele: Diese Spalte beschreibt die schon erwähnten Sicherheitsanforderungen, wie sie in Abschnitt 2.3 diskutiert wurden.

Beteiligte Akteure und Systemkomponenten: Ein Angreifer kann beim ausüben eines Angriffs verschiedene Rollen annehmen. Diese Rollen werden im Folgenden aufgezählt, dabei ist aber zu beachten, dass auch Kombinationen aus verschiedenen Rollen möglich sind.

- Fahrzeugführer *(Besitzer)*: Wenn der Fahrzeugführer gleichzeitig auch Fahrzeughalter ist, werden diese Rollen stellvertretend mit Fahrer bezeichnet. Ein denkbares Ziel wäre eine Dienstnutzung unter falscher Identität (Spoofing), um die Bezahlung des Dienstes umzugehen.

- Netzbetreiber: Sind Betreiber von stationären Netzen, die den Teilnehmern des VANETs Kommunikationsdienste vergleichbar mit zellulären Netzen anbieten. Bei Netzbetreibern besteht vor allem die Gefahr, da diese eine Erstellung von Bewegungs- oder Nutzungsprofilen anfertigen.

- Außenstehende: Darunter sind alle Parteien zu verstehen, die nicht aktiv am VANET teilnehmen, exklusive der bisher genannten. Das können also zum Beispiel Anwohner sein, denen Vor- oder Nachteile durch einen bestimmten Verkehrsverlauf entstünden. Ebenso werden Attentäter in diese Rolle eingeordnet, die zum Beispiel einen Unfall herbeiführen wollen.

Attraktivität für Angreifer: Aufteilung nach Aktivität für Angreifer. Manche Angriffe können bei VANETs völlig unattraktiv sein. Die Attraktivität wird von 1 bis 3

bewertet wobei 1 für wenig attraktiv, 2 für attraktiv und 3 für sehr attraktiv steht.

Aufwand und Schwierigkeit für Angreifer: Der Aufwand wird von 1 bis 3 bewertet. Für manche Angriffe ist der Aufwand in VANETs sehr groß, was mit einer 3 bewertet wird. Manche anderen sind weniger aufwendig und bekommen dafür eine 2. Eine letzte Kategorie die einen sehr kleinen Aufwand haben bekommen eine 1.

Eintrittswahscheinlichkeit: Die Wahrscheilichkeit mit der ein Angriff auftreten kann, resultiert durch das Verhältnis von Attraktivität zum Aufwand wie in der folgenden Formel abgebildet ist:

$E = \frac{Attraktivitaet}{Aufwand}$

Risiko: Die letzte Spalte ist reserviert für das Risiko. Unter einem Risiko [19] wird dabei ein Problem verstanden, welches noch nicht eingetreten ist. Das Ziel des Risikomanagements lässt sich in diesen zwei Punkten darstellen.

- Entweder zu verhindern, dass Risiken zu Problemen werden
- Oder aber den Schaden beim Eintritt der Angriffe oder Störungen zu minimieren.

Das dient somit auch der Schaffung von Transparenz, der Beseitigung von Unsicherheiten und ermöglicht die Nutzung von Chancen. Durch die konsequente Suche und Analyse möglicher Risiken wird zudem auch eine möglicherweise vorhandene Betriebsblindheit beseitigt. Das Risiko wird aus der Eintrittswahrscheinlichkeit und dem Schaden (Schutzbedarf) bewertet. Die allgemeine Formel lautet:

$Risiko = E \times Schutzbedarf$

Schichten des Hybridmodells	Beschreibung	Betroffene Schutzziele	Beteiligte Akteure und Systemkomponente	Attraktivität für Angreifer	Aufwand und Schwierigkeit für Angreifer	Eintrittswahrscheinlichkeit	Schutzbedarf	Risiko
Bitübertragungs- Schicht	Jamming	Verfügbarkeit	Aussenstehende Konkurrenten Netzbetreiber	3	1	3	3	9
	Eavesdropping	Vertraulichkeit Integrität	Netzbetreiber Aussenstehende Fahrzeugführer	3	2	1,5	3	4,5
Sicherungsschicht	Traffic Analysieren	Vertraulichkeit	Fahrzeugführer Netzbetreiber	2	1	2	2	4
	Network Allocation Vector manipulieren	Verfügbarkeit	Fahrzeugführer Aussenstehende	2	2	2	3	6
	Kleine Back-Off Zeiten wählen	Verfügbarkeit	Fahrzeugführer Aussenstehende	2	2	2	3	6
Vermittlungs- Schicht	Flooding Angriffe	Verfügbarkeit Integrität	Fahrzeugführer Aussenstehenden Konkurrenten	1	1	1	3	3
	Sybil Angriffe	Verbindlichkeit Authentizität	Aussenstehende	3	2	1,5	3	4,5
	Rushing Angriffe	Integrität	Fahrzeugführer Aussenstehenden Konkurrenten	1	3	0,33	1	0,33
	Wormhole Angriffe	Integrität Verfügbarkeit	Aussenstehenden Fahrzeugführer	1	3	0,33	1	0,33
	Sinkhole/Blackhole Angriffe	Integrität Verfügbarkeit Vertraulichkeit	Aussenstehenden Fahrzeugführer	3	2	1,5	3	4,5
	Selective Forwarding Angriffe	Integrität Verfügbarkeit Vertraulichkeit	Fahrzeugführer Netzbetreiber Konkurrenten	3	1	3	3	9
	Spoofing	Integrität Verfügbarkeit	Fahrzeugführer Aussenstehende Konkurrenten	3	2	0,66	3	2
	Angriffe auf Geocasting	Integrität Verfügbarkeit Vertraulichkeit	Netzbetreiber Aussenstehende Fahrzeugführer	2	3	1,5	3	4,5
Transport- Schicht	TCP Session Hijacking	Integrität Vertraulichkeit	Fahrzeugführer Netzbetreiber Aussenstehende	1	1	1	1	1
	TCP SYN Flooding	Integrität Vertraulichkeit	Fahrzeugführer Netzbetreiber Aussenstehende	1	1	1	3	3
Anwendungs- Schicht	Einspielen falscher Informationen	Integrität Vertraulichkeit	Fahrzeugführer Netzbetreiber Aussenstehende	3	1	3	3	9

Tabelle 2.1: Auflistung der verschiedenen möglichen Angriffe auf VANETs

2.5 Intrusion Detection

Eine endgültige Sicherheit vor Angriffen auf ein Netzwerk kann durch keine Firewall garantiert werden. Es ist schon üblich, dass Angreifer die Firewall passieren, oder, was für VANETs sogar noch unangenehmer ist, dass ein interner Angreifer bewusst oder unbewusst die Ressourcen unangemessen nutzt. Um solche Angriffe zu entdecken und zu stoppen wird eine Technik verwendet, welche die Arbeit der Firewall ergänzen soll: das Intrusion Detection System (engl. IDS).

Heberlein, et al. haben Intrusion wie folgt definiert:
„Eine Menge von Handlungen, deren Ziel es ist, die Integrität, die Verfügbarkeit oder die Vertraulichkeit eines Betriebsmittels zu kompromittieren“.
Anhand von Methoden versucht die Intrusion Detection, Angriffe auf ein IT-System zu erkennen und ihren Schaden zu minimieren. Somit ist sie, wie eine Art Alarmanlage, die vor Misshandlung des Netzes so früh wie möglich warnt.
Sicherheit in Computer Netzwerken befasst sich mit allen Angriffsformen, die über ein Netzwerk ausgeführt werden können. Es gibt verschiedene Ansatzpunkte in Software und Hardware, um ein Sicherheitssystem zu integrieren, welches vor einem Großteil der Angriffe schützt oder sie zumindest erkennt. Zuerst muss definiert werden, welche Komponenten mögliche Angriffe vermeiden und erkennen können.

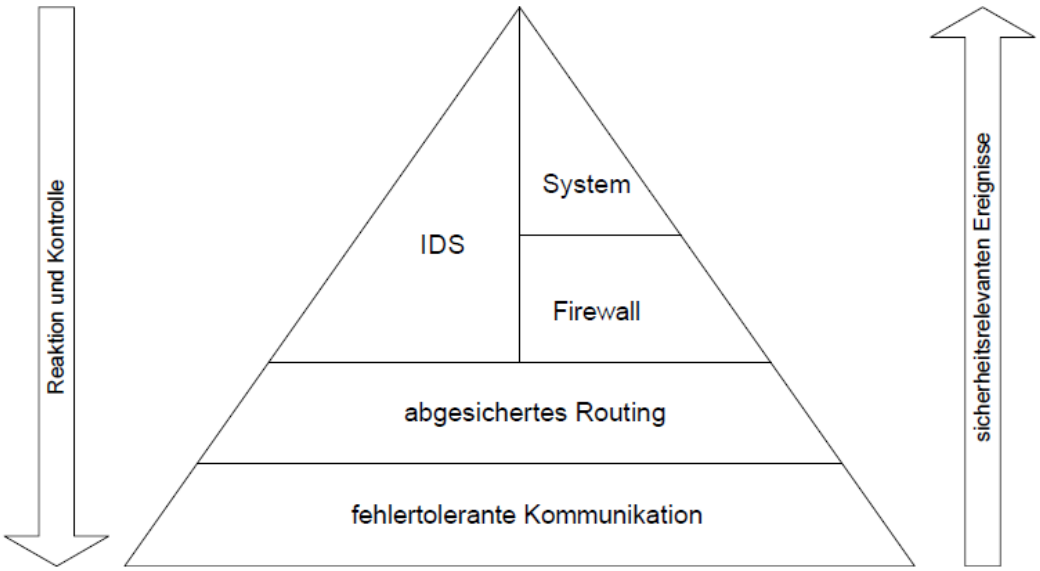

Abbildung 2.20: Schichtenmodell eines Sicherheitssystems für Ad hoc Netzwerke

In Abbildung 2.20 wird schematisch eine Menge von Sicherheitsmechanismen dargestellt, die bei Ad-hoc Netzwerke in Kraft treten. Die Funkschnittstelle ist gegen

physikalische Störungen der Kommunikation geschützt. Das abgesicherte Routing gibt bestimmte Garantien bezüglich Authentizität des Absenders und Korrektheit der Routingsinformationen. Darauf setzt die Firewall auf, die das System von Angreifern abschottet. Hinzu kommen Sicherheitsmechanismen im System selbst.

Die Intrusion Detection operiert auf den obersten Ebenen parallel zu der Firewall und der Systemsicherheit. Sie kann oberhalb des abgesicherten Routings Angreifer passiv und aktiv erkennen und analysiert Aktivitäten an der Firewall und im System. Sie stellt oft die letzte Abwehrfront eines Computer Systems gegen Eindringlinge dar, daher muss sie Angriffe erkennen und Gegenmaßnahmen einleiten können.
In dieser Arbeit wird nur die Erkennung von Angriffen analysiert.
Intrusion Detection kann durch zwei Methoden realisiert werden, die in den folgenden Abschnitten diskutiert werden.

2.5.1 Anomalie Erkennung

Bei der Anomalie Erkennung [4] entfällt eine Vorkonfiguration des Systems, dafür wird eine Anlaufphase benötigt. Das System überwacht in dieser Phase autonom alle Aktionen, die innerhalb des VANETs stattfinden und bewertet diese als legitimes Verhalten. Nach der Anlaufphase besitzt das IDS eine Datenbank, anhand deren es die überwachten Daten auf ihr Gefahrenpotential überprüft. Mit verschiedenen Heuristikverfahren wird die Ähnlichkeit zwischen dem neuen und dem bekannten Verhalten gemessen.
Sobald ein vorher festgelegter Schwellwert überschritten wird klassifiziert das IDS das neue Verhalten als Anomalie, und schlägt Alarm. Auf die Art kann auf vorgegebene Muster für Angriffe verzichtet werden, was es sogar möglich macht, noch unbekannte Angriffstechniken zu durchschauen und Alarm auszulösen.

Da nicht jede Anomalie auch einen Angriff darstellt, ist die Anomalie Erkennung sehr anfällig gegen häufige falsche Alarme. Durch neue Aufgaben für die Benutzer ändert sich auch das Nutzerverhalten mit der Zeit, was auch „conceptual drift“ genannt wird. Das Anomalie-IDS muss auf diese Änderungen reagieren können und seine Definition der Anomalien entsprechend anpassen.
Ein Angreifer kann aber auch diese Eigenschaft ausnutzen, indem er langsam das Verhaltensmuster in der Weise ändert, dass seine Attacke nicht mehr als Anomalie klassifiziert wird. Die Anomalie Erkennung ist völlig unbrauchbar, wenn ein Angreifer schon während der Anlaufphase Zugriff auf das System besitzt und diese Aktionen als Normalzustand bewertet wird.

2.5.2 Signatur Erkennung

Bei der Signaturbasierten Erkennung [4] werden Erfahrungswerte aus bekannten Angriffsmechanismen gesammelt. Diese Menge an Erfahrungswerte wird in einer vorgegebenen Datenbank gespeichert und an die Erkennung verbunden.
Solch eine Datenbank basiert auf der Tatsache, dass aus den Angriffen bestimmte Muster und Regelmäßigkeiten extrahiert werden können. Darunter fallen zum Beispiel:

- falsche Pakete
- verdächtige Ziel- oder Quelladressen
- auffällige Sequenz von Paketen
- ungewöhnlich viele Pakete
- verdächtige Ports
- ungewöhnliche TCP-Flag Kombinationen
- ungewöhnlicher Nachrichteninhalt
- widersprüchlicher Nachrichteninhalt

Durch Abgleich mit diesen Regeln versucht das IDS illegitimes Verhalten zu erkennen. Bekannte Attacken werden sehr zufällig durch die Vorgaben erkannt, aber schon leichte Änderungen an der Angriffsstrategie können zum Passieren der Kontrolle durch die Datenbank ausreichen, wenn keine charakteristischen Merkmale einer Kategorie von Angriffen gefunden werden kann.
Eine kontinuierliche Wartung des IDS ist in diesem Falle sehr wichtig, da neue und schwer erkennbare Angriffe durch neue Angriffstechniken zustande kommen. Daher konnten für diese Angriffe in diesem Zeitpunkt keine Regeln aufgestellt werden.

Eine Idee wäre die Festlegung aller rechtmäßigen Aktionen in einem Netzwerk. Doch solch eine Datenbank würde schnell durch die Anzahl der verschiedenen Nutzer und die Komplexität der verwendeten Software und unterschiedlicher Situationen sehr groß werden.

2.6 Angriffsszenarien auf VANETs

In diesem Abschnitt werden einige konkrete Angriffsszenarien auf VANETs definiert und vorgestellt, die dazu dienen, die Gefahr der Angreifer auch visuell darzustellen und zu analysieren. Auf eins von diesen Angriffszenarien wird im weiteren Verlauf der Arbeit zurückgegriffen.
Zuerst werden diese Szenarien skizziert und beschrieben. Dabei werden die Randbedingungen, Ziele des Angreifers und mögliche Erkennungsmaßnahmen erläutert.

2.6.1 Verhalten des Angreifers

Wie in den Abschnitten 2.4.1 und 2.4.2 beschrieben wurde definiert das Verhalten, in wieweit ein Angreifer sich passiv oder aktiv, bzw. beobachtend oder verändernd bezüglich der ihm möglichen Handlungen verhält.

- Jeder Angreifer kann unerkannt beobachten (passives Verhalten), solange er sich im Empfangsradius aufhält.
- Je höher Kompetenz, Ressourcen und Verbreitung eines Angreifers sind, desto wirkungsvoller können aktive Kontrolle oder Eingriffe werden.

2.6.2 Konkrete Angriffsszenarien

Aufgrund der vielfältigen Verkehrssituationen bieten sich für Angreifer mehrere Möglichkeiten, einen aktiven oder passiven Angriff auszuüben. Besonders bei unübersichtlichen und weiten Strecken, wo der Fahrer sich auf das VANET verlässt. Gerade auf der Autobahn, bei scharfen Kurven, an unübersichtlichen Kreuzungen, wegen dichtem Verkehr oder Bäumen könnte eine solche Situation auftreten. Deshalb werden in diesem Abschnitt zur Veranschaulichung einige Angriffszenarien vorgestellt. Ein Szenario davon wird in Abschnitt 3.1 näher untersucht.

2.6.2.1 Kurve

Scharfe Kurven stellen eine sehr große Herausforderung für manche Fahrer dar, besonders Nachts oder bei schlechtem Wetter. Zudem kann man nie genau wissen, was hinter der Kurve auf einen zukommt und wie die Kurve im Weiteren verläuft. Ein Angreifer könnte zum Beispiel einen virtuellen Stau oder ein Hindernis auf der Fahrbahn simulieren, indem er Warnnachrichten broadcastet. Dies könnte dazu führen, dass Unsicherheit zwischen den Fahrzeugen, die in die Kurve reinfahren, verbreitet wird. In Folge dessen kann sich ein richtiger Stau bilden, da die Fahrzeuge langsamer werden oder sogar Unfälle verursachen wenn ein Fahrer eine starke Bremsung aus Panik tätigen würde. Die Abbildung 2.21 zeigt einen Angreifer, der falsche Staumeldungen broadcastet und damit einen richtigen Stau verursacht.

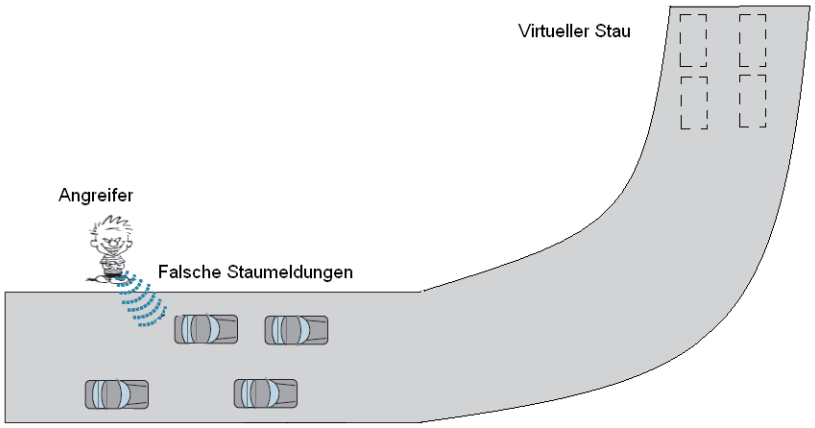

Abbildung 2.21: Angreifer beim Vortäuschen eines Staus vor einer scharfen Kurve

2.6.2.2 Unübersichtliche Nachtfahrt

Die größte Herausforderung die auf die meisten Autofahrer beim Fahren trifft, ist die Fahrt bei schwierigen Konditionen. Zum Beispiel bei einer nebeligen Nacht wo sich die Sicht erschwert. Dies können sich Angreifer zu Nutze machen, in dem sie zum Beispiel in diesem Szenario falsche Informationen in dem VANET verschicken, um die Reaktionen und Entscheidungen der Fahrzeugfahrer zu beeinflussen. In der Abbildung 2.22 wird ein solcher Fall dargestellt. Dabei verschickt der Angreifer zwei entgegengesetzte Nachrichten zu zwei hintereinander fahrenden Fahrzeuge. In der Nachricht, die an das fordere Fahrzeug gerichtet ist, steht eine Warnmeldung vor einer virtuellen Gefahr, die sich unmittelbar direkt vor dem Fahrer befinden soll. Dies führt dazu, dass der Fahrer des Fahrzeugs seine Geschwindigkeit reduziert oder sogar eine Vollbremsung hinlegt. In der Nachricht, die an das hintere Fahrzeug gerichtet ist, steht die Information, dass die Fahrbahn frei ist und dass er beschleunigen kann. Diese falschen Meldungen können bei Unaufmerksamkeit des Fahrers, bei schlechtem Wetter oder unübersichtlicher Straße, zur Gefahr werden.

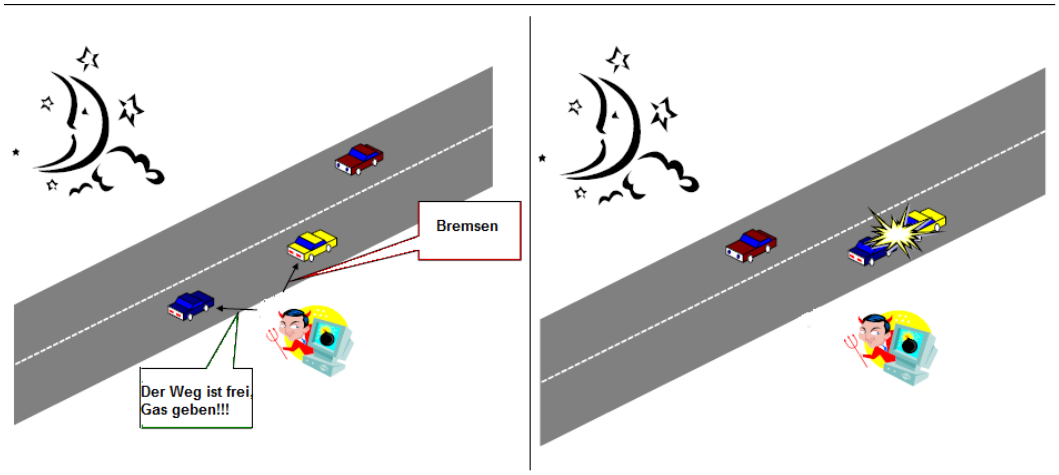

Abbildung 2.22: Angreifer beim Verbreiten von falschen Informationen bei Nacht und Nebel [27]

2.6.2.3 Eavesdropping

In diesem Szenario geht es um die Datensicherheit zwischen zwei oder mehrerer kommunizierender Fahrzeuge. Wie bereits im Abschnitt 2.4.3.1 erwähnt wurde kann ein Angreifer einen passiven Angriff ausüben, in dem er einfach die Identität eines Fahrzeugs übernimmt und somit die Kommunikation zwischen den Fahrzeugen mithört, oder einen aktiven Angriff, indem er die empfangenen Daten manipuliert oder verwirft.

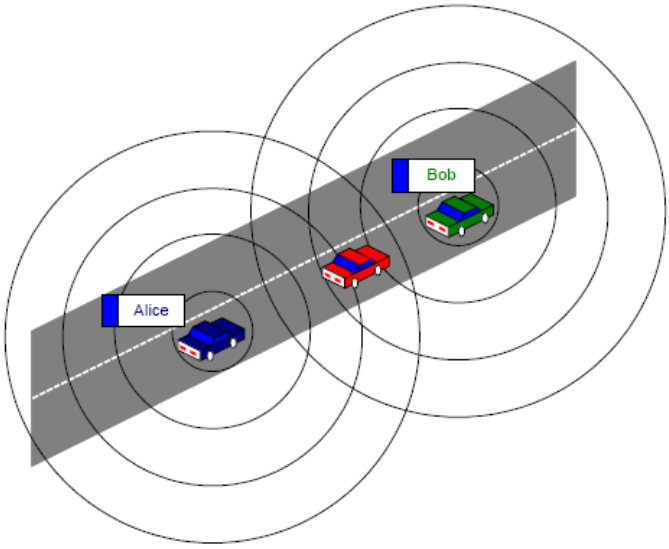

Abbildung 2.23: Angreifer beim Abhören der Kommunikation zwischen zwei Fahrzeugen [27]

Man kann als Gegenmaßnahme gegen Eavesdropping symmetrische und asymmetrische Kryptographie [26] anwenden. Dabei entstehen aber andere Nachteile.
Bei symmetrischer Kryptographie müssen zwei Kommunikationspartner, A und B, denselben Schlüssel verwenden. D.h. beide kennen den Authentifizierungsschlüssel. Dies wird zum Problem, wenn die Authentifizierung nicht gegenüber einer vertrauenswürdigen Stelle (engl. Trusted Third Party, TTP) erfolgt. Dann nämlich kennt der nicht vertrauenswürdige Kommunikationspartner B den Authentifizierungsschlüssel von A und kann sich in Folge als A ausgeben.

Bei der asymmetrischen Kryptographie besitzt jeder Teilnehmer seinen eigenen privaten Schlüssel und den dazugehörigen öffentlichen Schlüssel, der mit Hilfe einer Public Key Infrastruktur (PKI) in einem Zertifikat einer Identität zugeordnet wird. Ein Kommunikationspartner kann mit Hilfe des Zertifikats, und dem darin enthaltenen öffentlichen Schlüssel, die Authentifizierung (Signatur) einer Nachricht prüfen,

er kann aber selbst keine gefälschte Signierung vornehmen.
Der Nachteil asymmetrischer Kryptographie sind im Vergleich zur symmetrischen Kryptographie die sehr langen Zeiten, die zur Signaturgenerierung bzw. -verifizierung benötigt werden. Ein weiterer Nachteil ist, dass Informationen zu Zertifikatsrückrufen verteilt werden müssen.

2.6.2.4 Autobahn

Viele Fahrzeugführer wissen nicht was in den nächsten 100 Kilometern los ist. Das können Angreifer ausnutzen, um falsche Stauinformationen zu verbreiten. Ziel dieses Angriffs ist es, eine freiere Bahn für den Angreifer zu bekommen, nach dem die Opfer einen Umweg genommen haben. Angreifer können solche Angriffe nur aus Spaß ausüben, andere können finanzielle Ziele haben. Zum Beispiel ein Besitzer einer Raststätte, die unmittelbar an einer Autobahnausfahrt liegt, könnte einen Stau vortäuschen. Damit Autofahrer einen Umweg nehmen, der direkt über seine Raststätte läuft. Abbildung 2.24 zeigt ein vereinfachtes Beispiel einen solchen Angriffs.

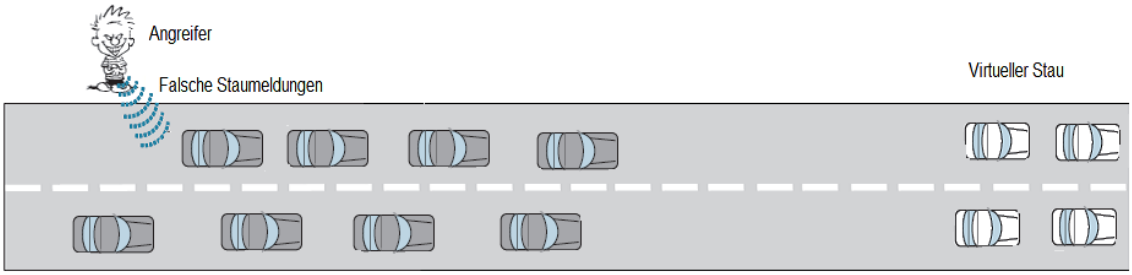

Abbildung 2.24: Angreifer beim Vortäuschen eines Staus auf einer Autobahn

Dieses Beispiel wird als Angriffsbasis für das zu entwickelnde IDS-System genommen. Dabei wird ein Teil einer dreispurigen geraden Autobahn genommen. Eine genauere Beschreiben folgt im nächsten Kapitel.

Kapitel 3

Konzept

Wie schon zu Begin der Arbeit als Ziel gesetzt wurde, soll ein IDS für einen bestimmten Angriff auf ein VANET entwickelt werden. In dem vorherigen Abschnitt wurden einige Angriffsszenarien auf VANETs grob beschrieben. In diesem Kapitel wird ein Szenario aus den beschriebenen Angriffsszenarien ausgesucht. Anhand dieses Szenarios werden die Komponenten des IDS beschrieben und detailliert erklärt, wie sie bei der Erkennung des Angriffs vorgehen.

3.1 Autobahn

Das Szenario der Autobahn wurde für das zu entwickelnde IDS als Basis genommen. In diesem Kapitel werden die Randbedingungen aufgezählt, die bei der Entwicklung des IDS beachtet werden. Anschließend wird detailliert und schrittweise erklärt, wie das entwickelte IDS bei der Angriffserkennung vorgeht.

3.2 Randbedingungen:

Auf dem simulierten Teil der Autobahn fahren verschiedene Fahrzeuge, die mit einem IDS System ausgestattet sind. Unter Fahrzeugen, zählen in dieser Arbeit alle Fahrzeuge, die für die Autobahn zugelassen sind. Diese Fahrzeuge sind mit den folgenden Komponenten ausgestattet:

- Ein Transmitter für den omnidirektionalen Empfang und Versand von Daten im VANET
- Eine Rechen- und Speichereinheit, welche die Verarbeitung, die Speicherung und die Auswertung der Nachrichten des VANETs vornimmt. Dazu gehört auch das Empfangen, Speichern und Versenden von Nachrichten
- Ein System zur Bestimmung der Position

- Die Position von Nachbarfahrzeuge kann per WLAN ermitteln werden

Die Autobahn wird in kleinen geographischen Kästchen unterteilt, in jedem diesen Kästchen passt höchstens ein Fahrzeug hinein.
Um die Fahrzeuge untereinander zu unterscheiden, wird jedem Fahrzeug eine Identität (ID) zugewiesen, jedes Fahrzeug das neu auf dem Autobahnteil erscheint, wird eine neue ID zugewiesen. Diese ID bleibt solange konstant bis das Fahrzeug den Autobahnteil verlassen hat.
Die Identität eines Fahrzeugs stellt ein Wiedererkennungsmerkmal dar, anhand dessen man zum Beispiel korrekt funktionierende Teilnehmer von böswillige oder fehlerhaften unterscheiden kann. Das impliziert natürlich, dass ein Knoten weder anonym auftreten, noch seine Identität beliebig ändern können darf, da sonst sämtliche Maßnahmen der Regulierung oder der Erkennung von böswilligen Fahrzeuge ins leere greifen würden.
Für diese Arbeit ist festzuhalten, dass die Identität eines Knotens ihn eindeutig charakterisiert und von anderen unterscheidbar macht.

Die Fahrzeuge bewegen sich alle mit der gleichen Geschwindigkeit, welche immer konstant bleibt. Zudem haben die Fahrzeuge eine Position als Tupel(x,y), x und y sind Koordinaten im Koordinatensystem. Anhand der Position des Fahrzeugs kann jedes Fahrzeug feststellen wo es sich befindet, wer seine Nachbarn sind und wo ein Ereignis stattfindet. Außerdem kommunizieren die Fahrzeuge miteinander, in dem sie Nachrichten verschicken und empfangen. Der Nachrichtenaustausch, bei der Meldung von Gefahren oder wichtigen Ereignissen ist verbindungslos. Damit wird Zeit gespart, wobei Nachrichten schnell ausgetauscht und verbreitet werden können. Ein Fahrzeug kann entweder mit seinen direkten Nachbarn per Broadcast, oder mit weit entfernten Fahrzeuge per Geocast kommunizieren. Abbildung 3.1 zeigt den Aufbau einer Nachricht.

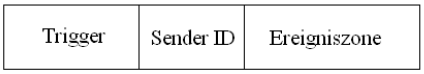

Abbildung 3.1: Aufbau eines VANET Pakets

Die Nachrichten sind wie folgt aufgebaut:

- SenderID ist die ID des Fahrzeugs, welches die Nachricht gesendet hat. Diese ID spielt eine sehr wichtige Rolle. Sie dient dazu, dass das Empfängerfahrzeug nicht die gleiche Nachricht von dem selben Senderfahrzeug empfängt. Das wiederum spielt eine sehr entscheidende Rolle im Konzept. Darauf wird im nächsten Abschnitt ganz detailliert eingegangen.
- Die Ereigniszone ist die Zone, in der das gemeldete Ereignis aufgetreten ist. In dieser Zone sollte das Ereignis stattfinden. Jedes Fahrzeug, dass zum ersten

mal eine Nachricht mit einer gewissen gemeldeten Zone empfängt, generiert für sich selbst eine größere Zone als die in der Nachricht gespeichert ist. So kann er weitere empfangene Pakete, dessen Zone ein bisschen von der ersten gemeldeten Zone abweicht und die vom selben Absender stammen, als Duplikat betrachten und verwerfen.

- Der Trigger zeigt, ob es sich um eine normale Ereignisnachricht oder einen Trigger handelt. Bei einem Trigger wird das Feld mit 1 besetzt, ansonsten mit 0. In den nächsten Abschnitten wird der Unterschied zwischen einer normalen Nachricht und einer Triggernachricht erklärt.

3.3 Szenario

In dieser Diplomarbeit wird ein IDS entwickelt, welches falsche Ereignismeldungen erkennen soll. Das Szenario sieht folgendermaßen aus:
Ein Fahrzeug sendet eine Nachricht an seine Nachbarfahrzeuge per Broadcast. In dieser Nachricht steht, dass in einer bestimmten Ereigniszone ein Unfall passiert ist. Die Fahrzeuge, die an dem Ereignis beteiligt waren, sind beschädigt und können deshalb keine Nachrichten mehr verschicken oder empfangen was zu einem Stau führen könnte. Diese Nachricht kann entweder eine wahre Ereignismeldung, aber auch eine falsche Meldung sein. Deshalb ist es die Aufgabe, des zu entwickelnden IDS, falsche Ereignismeldungen zu entdecken. Das IDS funktioniert aber unterschiedlich, je nach dem ob sich die Fahrzeuge, welche die Ereignisnachricht bekommen haben, in der Nähe der Ereignisregion sind oder nicht. Fahrzeuge, die sich in der Nähe der Ereignisregion befinden, können selber überprüfen. Andere müssen sich auf die erste Kategorie der Fahrzeuge verlassen, in dem sie ihnen eine Triggernachricht schicken, um sie dazu aufzufordern, das Ereignis zu prüfen.

Das folgende Aktivitätsdiagramm, dargestellt in Abbildung 3.2, stellt die vom IDS auszuführenden Aktionen dar, um falsche Ereignismeldungen zu entdecken. Die einzelnen Schritte sind im Diagramm nummeriert und anschließend textuel beschrieben.

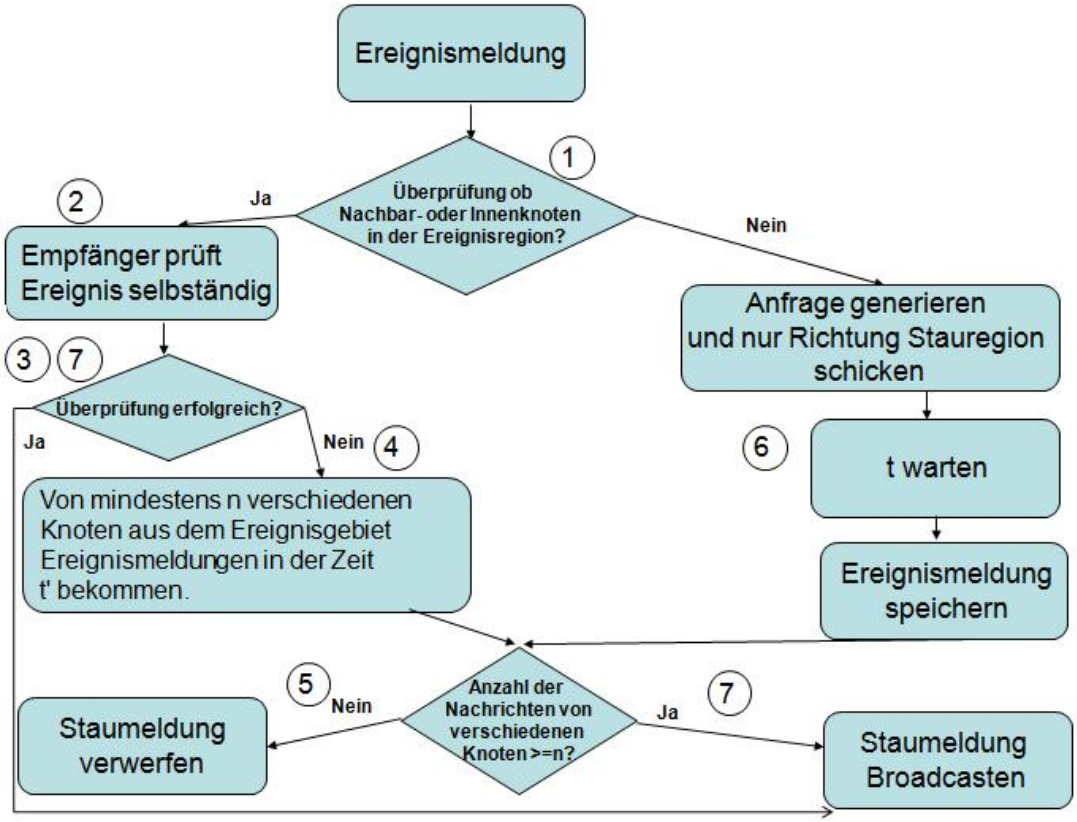

Abbildung 3.2: Ablauf nach dem Senden einer Ereignismeldung

Das IDS läuft in folgenden Schritten.

1. Wenn ein Knoten eine Ereignismeldung bekommt überprüft er, ob er ein Nachbar- oder Innenknoten der Ereignisregion ist.

2. Wenn das der Fall ist, sollte er selbständig das Ereignis prüfen, indem er beobachtet ob die Position eines Nachbarfahrzeugs, das sich in der Ereignisregion befindet, sich innerhalb von einer Periode t ändert oder Konstant bleibt.

3. Wenn die Überprüfung erfolgreich ist wird das Ereignis bestätigt und die Ereignismeldung innerhalb eines bestimmten Zeitintervalls gebroadcastet.

4. Wenn das Fahrzeug es nicht geschafft hat, das Ereignis selbst zu prüfen, dafür mindestens n Ereignisnachrichten von x verschiedenen Fahrzeugen mit verschiedenen IDs bekommen hat, broadcastet es die Ereignismeldung weiter, wie in Kapitel 2.2.3 erläutert wurde.

5. Wenn das Fahrzeug weder das Ereignis festgestellt hat, noch n Ereignismeldungen von x verschiedenen Fahrzeugen innerhalb von einer bestimmten Periode

t' bekommen hat, ignoriert es die Ereignismeldung und broadcastet sie nicht weiter.

6. Der zweite Fall tritt ein wenn das Fahrzeug kein Nachbar- oder Innenknoten in der Ereignisregion ist. Es speichert die Ereignismeldung, und falls es keine Anfrage bekommen hat, generiert es eine Anfrage und schickt sie per Geocast an die Ereignisregion. Das Fahrzeug muss dabei eine gewisse Periode t warten. Wenn es keine weitere Ereignismeldungen bekommen hat, wird die erste Ereignismeldung ignoriert. Die Periode wird in dieser Arbeit in Runden und nicht in Sekunden berechnet. Das heißt, die Periode ist eine gewisse Zahl an Runden, die das Fahrzeug warten muss, bis er eine gewisse Anzahl n von Nachrichten bekommen hat.

7. Das Ereignis wird bestätigt und gebroadcastet, wenn das Fahrzeug n weitere Ereignismeldungen von x verschiedenen Fahrzeugen bekommen hat, oder wenn das Fahrzeug das Ereignis selbst erfolgreich geprüft hat.

Das Verhalten der Knoten nach dem Erhalt einer Triggernachricht (Anfrage), wird in dem folgenden Aktivitätsdiagramm 3.3 dargestellt. Die einzelnen Schritte sind im Diagramm nummeriert und anschließend textuel beschrieben.

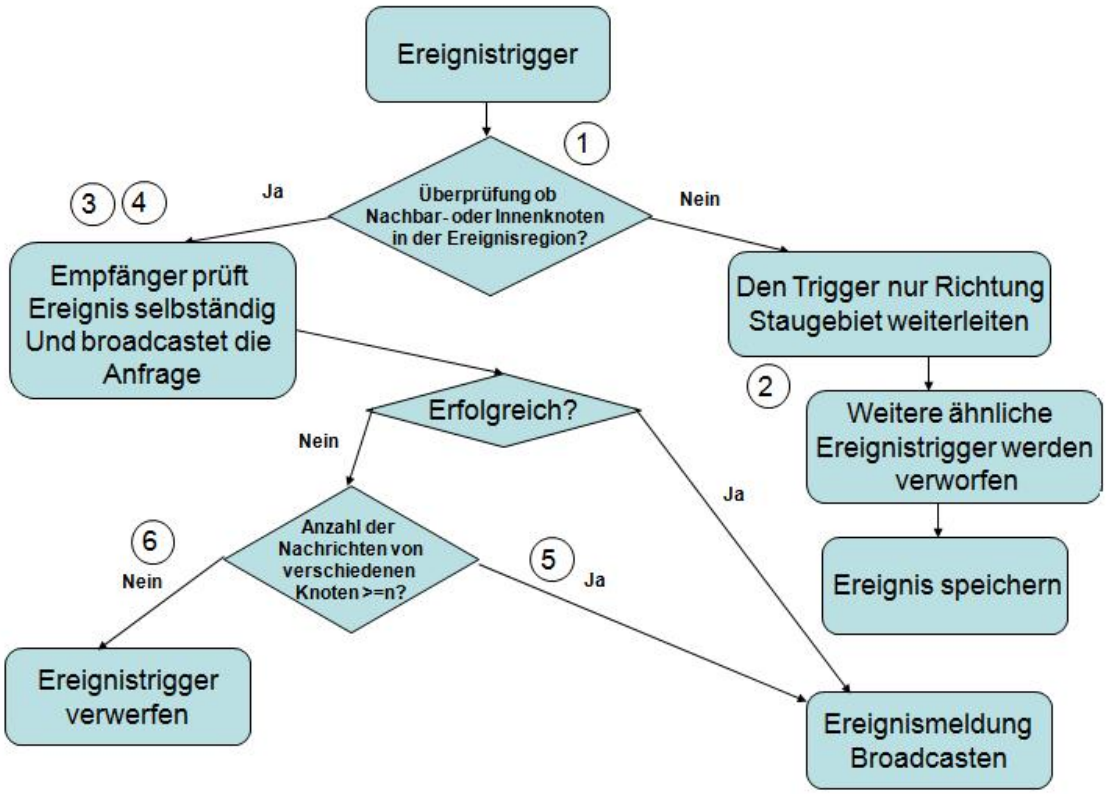

Abbildung 3.3: Ablauf nach dem Senden einer Triggermeldung

1. Wenn ein IDS eine Anfrage bekommt, prüft es, ob es ein Nachbar- oder Innenknoten in der Ereignisregion ist.

2. Wenn das Fahrzeug kein Nachbar- oder Innenknoten der Ereignisregion ist, und eine Anfrage bekommt, speichert es eine Ereignismeldung und schickt die Anfrage weiter Richtung Ereignisregion per Geographisches Broadcast, wie in Kapitel 2.2.3 erläutert wurde. Die Anfrage wird ebenso gespeichert, damit andere ähnliche Anfragen von anderen Fahrzeugen nicht berücksichtigt werden.

3. Wenn das Fahrzeug ein Nachbar- oder Innenknoten in der Ereignisregion ist, überprüft es selbst, ob das Ereignis stimmt, wie bereits im letzten Abschnitt beschrieben wurde. Wenn das Ereignis stimmt, broadcastet das Fahrzeug eine Ereignismeldung.

4. Wenn die Anfrage die Ereignisregion erreicht hat, wird sie von dem Fahrzeug broadcastet, damit andere Fahrzeuge in der Nähe oder innerhalb der Ereignisregion auch das Ereignis prüfen können.

5. Wenn das Fahrzeug es nicht geschafft hat das Ereignis zu prüfen, dafür aber n

Ereignismeldungen von x verschiedenen Fahrzeugen innerhalb von einer Periode bekommen hat, wird das Ereignis bestätigt und die Ereignismeldung weiter gebroadcastet.

6. Wenn das nicht der Fall ist wird die Anfrage ignoriert.

In dieser Diplomarbeit prüfen Fahrzeuge die Position der Nachbarfahrzeuge mit Hilfe des WLANs. Eine weitere Alternative wäre, die Position der Nachbarfahrzeuge mit Hilfe von Beacons zu überprüfen. Als Beacons werden periodisch versendete Nachrichten bezeichnet. Sie enthalten mindestens die aktuelle Position, Fahrrichtung, Geschwindigkeit und Beschleunigung (positiv oder negativ) des Fahrzeugs sowie die Sendezeit. Sie spiegeln also eine Art aktuellen Status des Fahrzeugs wieder. Beacons werden per Broadcast an alle Fahrzeuge innerhalb der Sendereichweite verteilt und nicht weitergeleitet (Single-Hop).

Anhand des oben beschrieben Verfahren soll das IDS in der Lage sein, empfangene Nachrichten auf Richtigkeit prüfen zu können. Um dies zu testen, wird eine Simulation implementiert, die möglichst die Arbeitsweise des IDS darstellt. Im nächsten Kapitel werden dann die Schritte detailliert erklärt, wie diese Simulation funktioniert, um falsche Nachrichten zu erkennen. Damit wird verifiziert, ob das Konzept die geforderte Aufgabe erfüllt oder nicht.

Kapitel 4

Simulation

In dem letzten Kapitel wurde ein Konzept entwickelt. Dieses Konzept dient dazu dass Fahrzeuge empfangene Nachrichten anhand verschiedener Methoden auf Plausibilität prüfen. In diesem Kapitel wird das IDS anhand einer Simulationssoftware getestet. Dabei werden die Komponenten der IDS Simulation vorgestellt und beschrieben.

Abbildung 4.1 zeigt die grobe Softwarearchitektur des entwickelten IDS. Sie besteht aus fünf Klassen, dessen Aufgaben in den folgenden Abschnitten im Detail erläutert werden.

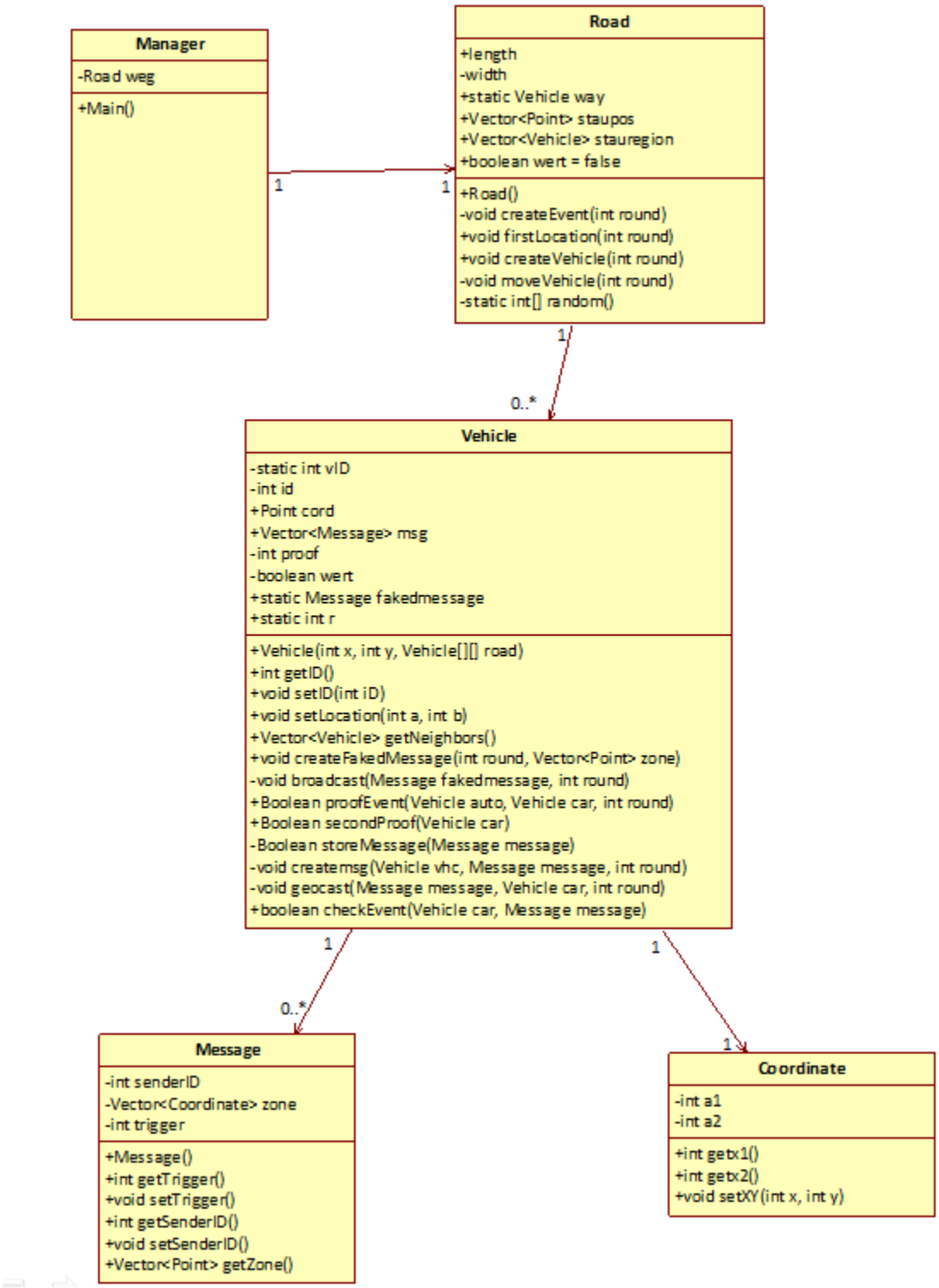

Abbildung 4.1: UML Klassendiagramm der Konzeptsimulation

4.1 Manager

Die Klasse Manager ist der Generator des IDS. Die erste Aufgabe dieser Klasse ist es, den Autobahnteil, beziehungsweise die Simulationsstrecke zu erzeugen. Die zweite Aufgabe ist es dass man in dieser Klasse die Anzahl der Runden bestimmen kann. Eine Runde bedeutet, die Verschiebung aller Fahrzeuge um ein Kästchen. Hier ist die Bewegung der Fahrzeuge nicht zeitbasiert, sondern rundenbasiert.

Wenn der Weg erfolgreich erzeugt wurde, werden die zwei Methoden *firstLocation()* und *createVehicle()* aufgerufen. Die Methode *firstLocation()* belegt den Weg mit Fahrzeugen in einer bestimmten Position, wie der Streckenteil in der Runde 0. Die Methode *createVehicle()* bewegt die vorhandenen Fahrzeuge auf der Simulationsstrecke um ein Kästchen, und das jede Runde. Danach erzeugt sie neue Fahrzeuge und platziert sie jeweils an jedem Ende der Autobahn. Damit wird gewährleistet, dass der Autobahnteil wie eine echte Autobahn aussieht. Die Funktionalität der zwei Methoden wird im Folgenden, unter Road, genau beschrieben.

4.2 Road

Die Klasse *Road* spielt eine entscheidende Rolle in der Simulation. Dabei wird ein Zweidimensionaler Array vom Typ *Vehicle* erzeugt. In Runde 0 ist der Weg so besetzt, wie auf der Abbildung 4.2 dargestellt ist. Dafür ist die Methode *firstLocation()* zuständig.

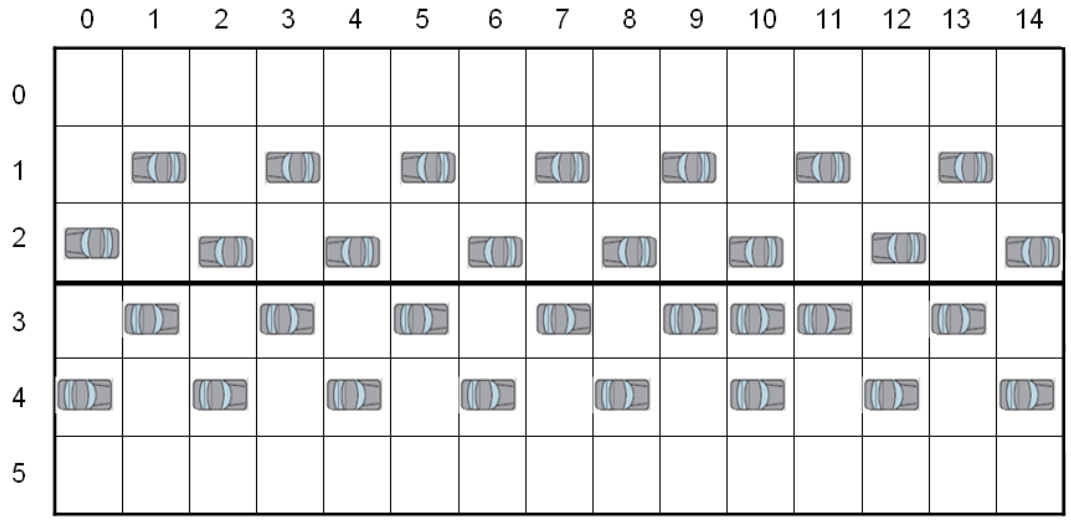

Abbildung 4.2: Ausblick aus dem Autobahnteil bei Runde 0

Nach jeder Runde werden neue Fahrzeuge erzeugt. Für die Erzeugung und die Plat-

zierung der Fahrzeuge auf der Autobahn sind die Methoden *createVehicle()* und *random()* zuständig. Neue Fahrzeuge werden immer in den drei rechts oberen oder drei links unteren Felder erzeugt. Vor der Erzeugung und der Plazierung der Fahrzeuge müssen zuvor die anderen Fahrzeuge auf der Autobahn bewegt werden. Die Fahrzeuge werden dann jede Runde um ein Kästchen bewegt. Deshalb ruft die Methode *createVehicle()* die Methode *moveVehicle()* auf, bevor sie neue Fahrzeuge erzeugt und auf dem Feld plaziert. Fahrzeuge die das Ende der Autobahn erreicht haben, werden gelöscht.

Die Methode *random()* erzeugt Zufallszahlen und speichert sie in einem Array. Diese Zufallszahlen sind in einem Bereich eingeschränkt: $0 \leq Zufallszahl \leq x$. Die Spuren der sechsspurigen Autobahn sind mit 0 bis 5 markiert. Die Methode funktioniert wie folgt. Die Zufallszahlen werden zunächst durchsucht. Wenn die Methode eine Zahl entdeckt, so dass Zahl = Spur ist, wird ein Fahrzeug in dieser Spur erzeugt. Der Bereich dieser Zufallszahlen kann geändert und somit die Anzahl der erzeugten Fahrzeuge gesteuert werden. Umso größer der Bereich ist, desto weniger Fahrzeuge werden erzeugt.

Die Methode *firstLocation()* ruft die Methode *createEvent()* auf. Dabei broadcastet das Fahrzeug, dass sich zum Beispiel in der Position (4,0) in Runde 0 befindet, eine Ereignismeldung. Das Ereignis in diesem Beispiel ist ein Stau in den Positionen (3,10),(4,10) und (5,10), der durch einen Unfall verursacht wurde.

4.3 Vehicle

Die Klasse *Vehicle* ist auch eine Kernklasse im IDS. Dabei enthält jedes Fahrzeug eine eigene ID. Diese ID wird inkrementiert, sobald ein neues Fahrzeug erzeugt wird, und wird diesem zugewiesen. So wird sichergestellt, dass jedes Fahrzeug eine ID hat, die kein anderes haben kann. Jedes Fahrzeug enthält einen Vektor, in dem alle seine Nachbarn gespeichert werden, solange es sich auf der Autobahn befindet. Dieser Vektor ändert sich ständig während der Bewegung der Fahrzeuge weil sie zum Teil immer neue Nachbarn haben. Diese Nachbarn werden durch die Methode *getNeighbors()* gesucht und in dem Vektor gespeichert.

In der Klasse *Vehicle* werden auch die Nachrichten erzeugt und weitergeleitet. Anhand der Methode *createFakedMessage()* wird eine Nachricht erzeugt. Dabei wird eine Instanz der Klasse *Message* erzeugt und in jedem Zielfahrzeug gespeichert. Zielfahrzeuge sind Nachbarfahrzeuge vom Angreifen, an den die falsche Nachricht broadcastet wurde. Diese Nachricht enthält die ID des sendenden Fahrzeugs, und die Zone in der das Ereignis stattfindet.
Nach dem Erzeugen der Nachricht wird sie an die Nachbachknoten gebroadcastet. Dies geschieht durch die Methode *broadcast()*. Diese Methode sucht nach allen Nach-

barknoten, die in dem Vektor gespeichert sind und speichert die Nachricht in einem Nachrichtenvektor bei jedem Nachbarfahrzeug. Das Speichern der Nachrichten erfolgt durch die Methode *storeMessage()*. Nachrichten werden erst gespeichert, wenn sie nicht von demselben Absender stammen und die gleiche Ereignisregion betreffen. Doppelte Nachrichten werden ebenfalls nicht gespeichert. Triggernachrichten werden nur einmal gespeichert, andere empfangene Triggernachrichten die die gleiche Ereigniszone betreffen, werden nicht beachtet.

Die Nachbarfahrzeuge, die eine Nachricht bekommen haben, prüfen, wie im Abschnitt 3.1.2 erläutert, ob sie Nachbarknoten der Ereignisregion sind. Dafür ist die Methode *checkEvent()* zuständig. Wenn die Methode *true* zurück gibt, muss das Fahrzeug selbst feststellen, ob das Ereignis richtig ist. Dies geschieht in dem es die Position eines Nachbarfahrzeugs, welches sich innerhalb der Ereignisregion befindet, innerhalb von 2 Runden durch die Methode *proofEvent()* kontrolliert. Wenn das Ereignis tatsächlich stimmt, wird die Ereignismeldung gebroadcastet.

Wenn das Fahrzeug nach dem Abruf der Methode *checkEvent()* feststellt, dass es kein Nachbarfahrzeug von der Ereignisregion ist, erzeugt es eine Triggernachricht. Diese Triggernachricht unterscheidet sich von der normalen Nachricht, indem sie im Feld Trigger eine 1 hat. Nach dem Erzeugen dieser Triggernachricht wird sie Richtung Ereignisregion per Geocast geschickt, was durch die Methode *geocast()* geschieht. Jedes Zwischenfahrzeug, welches die Triggernachricht bekommen hat, prüft, ob es Nachbar von der Ereignisregion ist. Wenn nicht, speichert es die Triggernachricht und eine Ereignisnachricht in seinem Vektor und schickt die Triggernachricht per Geocast weiter. Dies geschieht dann durch die Methode *createmsg()*.

Die Methode *geocast()* sucht unter den Nachbarfahrzeugen des aufrufenden Fahrzeugs den Nachbarn, der am nächsten an der Ereignisregion liegt. Wenn dieser Nachbar nicht vorhanden ist, wird die Nachricht an ein Fahrzeug geschickt, das sich in gleicher Höhe oder zwischen dem sendenden Fahrzeug und der Ereignisregion befindet. Damit wird die Netzlast verringert, wie bereits im Abschnitt 2.2.3 erwähnt.

Die Methode *secondProof()* prüft für ein bestimmtes Fahrzeug ob es n Ereignismeldungen von x verschiedenen Fahrzeuge bekommen hat. Wenn das der Fall ist, wird das Ereignis bestätigt und true zurückgegeben.

Wenn die Triggernachricht ein Fahrzeug erreicht hat, welches sich in der Nähe der Ereignisregion befindet, wird sie broadcastet. Somit können alle Nachbarfahrzeuge der Ereignisregion die Ereignisprüfung durchführen.

4.4 Message

In der Klasse *Message* werden die Attribute definiert, die in jeder Nachricht gespeichert werden müssen:

- ID des sendenden Fahrzeugs
- 0 für normale Nachricht und 1 für Triggernachricht
- Einen Vektor von Coordinaten, der die Ereignisregion darstellt

Je nach dem ob ein Fahrzeug eine Triggernachricht oder eine normale Nachricht erzeugen möchte, wird eine Instanz dieser Klasse erzeugt und die Attribute entsprechend belegt. Diese Instanz wird daraufhin in den Zielfahrzeugen gespeichert, so wird der Nachrichtenaustausch gewährleiste.

4.5 Coordinate

Die Klasse *Coordinate* stellt die Koordinaten jedes Fahrzeugs als Tupel dar. In diesem Tupel kommen die Einträge des *way* Arrays rein, indem das Fahrzeug sich befindet. Für die Position der Fahrzeuge wurde die Klasse *Point* von der JAVA-API verwendet. Die Position der Fahrzeuge ist sehr wichtig um die Nachbarfahrzeuge zu finden, was eine Grundlage für das Broadcasten ist. Außerdem muss jedes Fahrzeug seine Koordinaten kennen, damit es herausfinden kann wie weit es von der Ereignisregion ist. Das Geocasting basiert auch zum größten Teil auf den Koordinaten der Nachbarfahrzeuge. So werden nur Fahrzeuge ausgesucht, dessen Koordinaten am nächsten zu den Koordinaten der Ereignisregion sind, um die Triggernachricht weiterzuleiten.

Kapitel 5

Evaluierung

In diesem Kapitel wird die Funktionalität des entwickelten IDS Anhand von Tests geprüft. Diese Tests sind von vielen Faktoren abhängig. In den folgenden Abschnitten werden diese Tests, und die daraus resultierenden Ergebnisse, erläutert.

5.1 Normalfall

Im Normalfall ist der Autobahnteil von Fahrzeugen normal besetzt. Dieser sieht dann aus, wie in der Abbildung 4.2 dargestellt. Dabei hat jedes Fahrzeug mindestens ein Nachbarfahrzeug mit dem es kommunizieren kann. So ist auch gewährleistet, dass die Triggernachrichten ordnungsgemäß an die Ereignisregion weitergeleitet werden. Das Weiterleiten der Triggernachrichten erfolgt durch Geocasting. Wenn ein Fahrzeug keine Nachbarn hat, wird diese Nachricht nicht mehr weitergeleitet. Der Normalfall stellt die Lage der Fahrzeuge in Runde 0 dar.

So wie die Fahrzeuge angeordnet sind wird das Geocasting 100% gewährleistet. Deshalb wurde die Runde 0 als Basis Runde für den folgenden Test gewählt.
Innerhalb von Runde 0 wird ein falscher Stau gemeldet, und dann im Laufe der Runden wurde die Zahl der Fahrzeuge, die diesen Stau erkannt haben, berechnet. Die Berechnung erfolgt in Abhängigkeit von der Anzahl der Staunachrichten, die in jedem Fahrzeug gespeichert sind. Abbildung 5.1 zeigt die Anzahl der Fahrzeuge die das Ereignis im Laufe der Runden entdeckt haben.

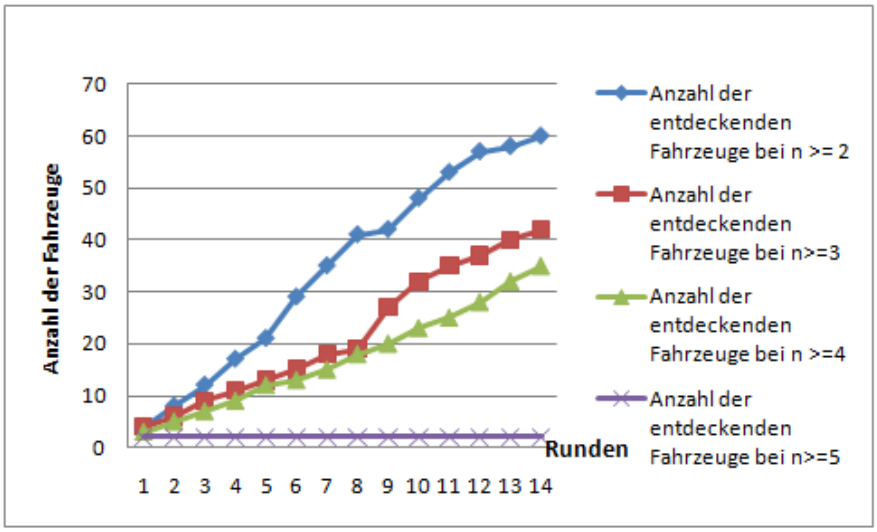

Abbildung 5.1: Anzahl der Fahrzeuge die ein Ereignis entdeckt haben, bei Variation der gespeicherten Nachrichten

Wie in Grafik 5.1 dargestellt, ist die Anzahl der Fahrzeuge, die das Ereignis entdeckt haben, stark beeinflusst von der Mindestanzahl der empfangenen Ereignismeldungen um ein Ereignis zu entdecken. Wenn die Mindestanzahl der empfangenen Nachrichten hoch gesetzt wird, sinkt automatisch die Anzahl der entdeckenden Fahrzeuge. Bei einer sehr hochgesetzten Mindestanzahl der empfangenen Nachrichten sind das nur noch die Nachbarfahrzeuge der Ereignisregion, die das Ereignis selber festgestellt haben.

Bei sehr kritischen Situationen, wenn man ein Ereignis melden möchte, muss man entweder die Mindestanzahl der gespeicherten Nachrichten verkleinern oder andere Kriterien beachten. Eins von diesen Kriterien ist zum Beispiel die Dichte des Verkehrs. Dieses Kriterium spielt beim Verbreiten der Nachrichten und der Entdeckung von Ereignissen eine sehr große Rolle.

5.2 Variierende Autodichte

Wie bereits erwähnt, spielt die Dichte des Verkehrs eine sehr entscheidende Rolle bei der Prüfung der Richtigkeit der Ereignisse. Wie in dem Test im Abschnitt 5.1 gezeigt wurde, wenn die Verkehrsdichte sehr hoch ist und dabei die Mindestanzahl der empfangenen Nachrichten hochsetzt wird, dann spielt die Anzahl der empfangenen Ereignismeldungen keine große Rolle und so wird das Ereignis von vielen Fahrzeuge erkannt. Die Grafik in Abbildung 5.2 zeigt die Anzahl der Fahrzeuge die das Ereignis entdeckt haben in Abhängigkeit von der Dichte des Verkehrs im Laufe der Runden. Dabei wurde eine Ereignismeldung in der 5. Runde erzeugt.

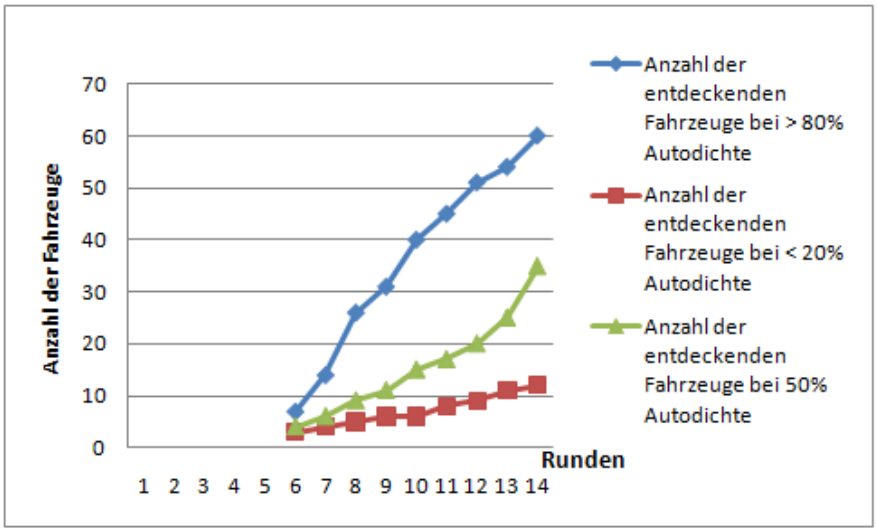

Abbildung 5.2: Anzahl der Fahrzeuge die ein Ereignis entdeckt haben mit variiernder Verkehrsdichten

Das Diagramm in Abbildung 5.2 zeigt, dass die Anzahl der Fahrzeuge, die ein Ereignis entdeckt haben sinkt, wenn die Verkehrsdichte kleiner wird. Dies hat nicht nur mit der Anzahl der vorhandenen Fahrzeugen auf der Autobahn zu tun. Hier tritt noch ein anderes Kriterium in Kraft. Bei geringer Verkehrsdichte kann manchmal das Geocasting nicht richtig funktionieren. Auch das Broadcasten funktioniert nicht, wenn ein Fahrzeug keine direkten Nachbarn besitzt. So gehen Ereignismeldungen verloren und werden von keinem Fahrzeug empfangen. Für ein IDS System ist das gleichzeitig Vorteil- und Nachteilhaft. Auf der einen Seite ist die Anzahl der Fahrzeuge, die eine falsche Ereignismeldung bekommen, sehr gering. Auf der anderen Seite können die Triggernachrichten verloren gehen. So kann nicht herausgefunden werden, ob es das Ereignis tatsächlich gibt, oder nicht.

5.3 Entfernung von der Ereignisregion

In diesem Test wird die Anzahl der Fahrzeuge ermittelt, die ein Ereignis entdeckt haben, und das in Abhängigkeit von der Entfernung des Melders von der Ereigniszone. Bei diesem Test wird eine normal belegte Autobahn angenommen, damit das Geocasting gewährleistet wird. Die Mindestanzahl der gespeicherten Nachrichten, um ein Ereignis zu entdecken, liegt in diesem Test bei 2.

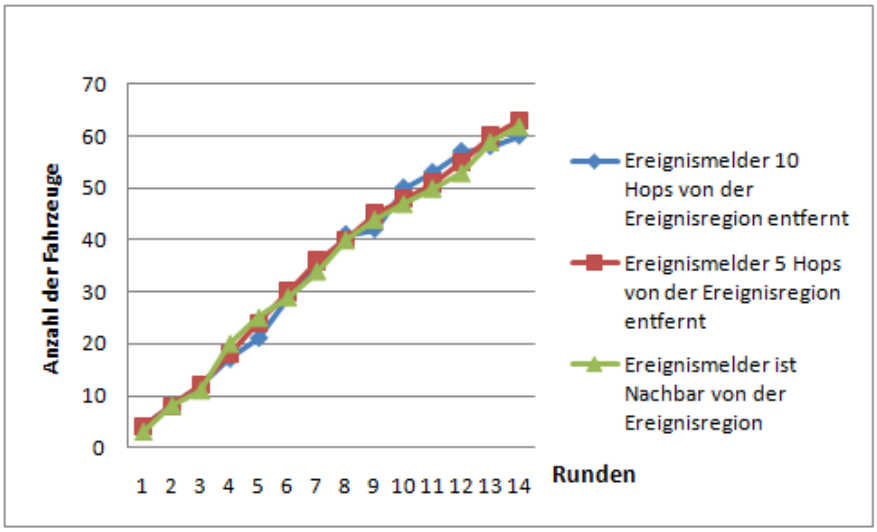

Abbildung 5.3: Anzahl der Fahrzeuge die ein Ereignis entdeckt haben mit variiernder Entfernung von der Ereignisregion

Das Diagramm in Abbildung 5.3 zeigt, dass die Anzahl der Fahrzeuge, die ein Ereignis entdeckt haben, konstant bleibt, egal wo sich der Ereignismelder befindet. So kann man feststellen, dass die Entfernung des Ereignismelders von der Ereignisregion keine Rolle bei der Ereignisentdeckung und -prüfung spielt.

Die drei Tests zeigen, dass eine erfolgreiche Ereigniserkennung nur geschehen kann, wenn die Autobahn dicht besetzt ist. Damit wird auch das Geocasting gewährleistet und kann die Triggernachrichten erfolgreich zur Ereignisregion weiterleiten. Ansonsten müsste man die Mindestanzahl der gespeicherten Nachrichten runtersetzen. So können viele Fahrzeuge das gemeldete Ereignis entdecken. Die Entfernung des Ereignismelders von der Ereignisregion spielt keine große Rolle.

Kapitel 6

Fazit

6.1 Zusammenfassung

In dieser Arbeit wurde nach einer detaillierten Erklärung der Begriffe Ad-hoc Network, MANET und VANET eine Auflistung und anschließend einen Vergleich der verschiedenen Routingverfahren, die in VANETs auftreten können, durchgeführt. Unter diesen Routing Verfahren wurde das Geocasting ausgesucht, weil es sich für die hohen Mobilität der einzelnen Knoten für geeignet gezeigt hat.

Anschließend wurde auf die Sicherheitsanforderungen eingegangen. Dabei wurden die Begriffe Datenintegrität, Informationsvertraulichkeit, Verfügbarkeit, Verbindlichkeit, Anonymisierung und Peudomisierung eingeführt und ihre Wichtigkeit bei VANETs anhand von Beispielen gezeigt.
In Abschnitt 2.4 wurden einige Angriffszenarien genannt, wobei sie in passiven und aktiven Angriffe kategorisiert wurden. Anhand eines Hybridmodells wurden jeder Schicht mögliche Angriffe zugeordnet und ausführlich diskutiert.

Die Angriffe die bereits definiert wurden, sind dann in einer Tabelle zusammengefasst worden. In dieser Tabelle wurde das Risiko berechnet, dass ein Angriff bei VANETs hervorrufen kann. Angriffe auf der Anwendungsschicht sind mit sehr hohem Risiko angesiedelt und stellen eine große Gefahr für VANETs dar. Deshalb wurden manche dieser Angriffe dann mit konkreten Beispielen gezeichnet und dargestellt.
Das Verbreiten von falschen Informationen auf Autobahnen ist das ausgewählte Szenario für das Angriffserkennungssystem. Dabei wurde ein Konzept erstellt, wie dieses IDS vorzugehen hat, wenn ein Fahrzeug eine Nachricht bekommt, um zu überprüfen, ob sie richtig oder falsch ist. Dabei wurden Randbedingungen aufgestellt, damit das IDS richtig funktioniert. Dieses Konzept wurde anhand einer Simulation realisiert, um zu überprüfen, ob es tatsächlich seine Aufgabe erfolgreich erfüllt.

Nach der Beschreibung der Simulationsimplementierung und ihren wichtigsten Kom-

ponente wurden Tests durchgeführt, um die Funktionalität der Erkennung zu testen. Dabei wurde festgestellt, dass die Dichte des Verkehrs, Entfernung des Eregnismelders von der Ereigniszone und die Anzahl der empfangenen Nachrichten in jedem Fahrzeug eine große Rolle bei der Erkennung von Ereignissen spielt.

6.2 Ausblick

Obwohl viele Teile dieser Arbeit ausführlich und detailliert erklärt wurden, gibt es noch viele offene Punkte, an denen weitgehende Forschungen nötig ist. Unter anderem kann in den Bereichen Routing und Intrusion Detection weiter geforscht werden. Man könnte die Routingalgorithmen verfeinern, um die Erreichbarkeit der Fahrzeuge zu erweitern und um das Geocasting performanter zu machen. Außerdem könnten andere Angriffe auf VANETs genauer betrachtet, und ein IDS dagegen entwickelt werden. Darauf basierend könnte ein übergreifendes IDS entwickelt werden, das eine Menge von Angriffen erkennen kann.

Weiterhin, und als praktischer Vorschlag, wäre der Test des Systems in Prototypen sinnvoll, um zu überprüfen, wie es in der Realität arbeiten würde.

Literaturverzeichnis

[1] http://www.ece.iupui.edu/~dskim/manet/. *Purdue School of Engineering and Technologie*, 14.11.2009.

[2] I. Aad and J.-P Hubaux E.Knightly. Denial of Service Resilience in Ad Hoc Networks. In: Proc. of the 10th Annual International Conference on Mobile Computing and Networking. *ACM Press, Philadelphia, PA, USA*, pages 202–215, 2004.

[3] Tannenbaum Andrew S. Computernetzwerke. 4. Auflage. *Pearson Studium, München*, 2003.

[4] Revecca Gurley Bace. Intrusion Detection. *Macmillan Technical Publishing, USA*, pages 91–110, 2000.

[5] J. Broth and B. B. Johnson. The dynamic source routing protocol for mobile ad hoc networks. *Internet draft*, 1998.

[6] Statistisches Bundesamt. http://www.destatis.de/jetspeed/portal/cms/. 12.2009.

[7] William R. Cheswick and S. Hartte. Firewall and Internet Security: Repelling the Wily Hacker. *Addison-Wesley*, 1995.

[8] Carlos de Morais Cordeiro and Dharma Prakash Agrawal. AD HOC and SENSOR NETWORKS Theory and applications. *World Scientific Publishing Co. Pte. Ldt.*, 2006.

[9] J.R. Douceur. The Sybil attack. *1st International Workshop on Peer-to-Peer Systems (IPTPS 02)*, 2002.

[10] Claudia Eckert. IT-Sicherheit: Konzepte - Verfahren - Protokolle. 6. Auflage. *Oldenburg Verlag München*, 2009.

[11] Charles E.Perkins and Pravin Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers.

[12] Mario Gerla. Ad hoc Networks. *Springer Science and Business Media, New York*, 2005.

[13] J. Hightower and G. Borriello. Location Systems for Ubiquitous Computing. *IEEE Computer, Vol. 34, No. 8*, pages 57–66, 2001.

[14] John D. Howard. An Analysis of Security Incidents on the Internet 1988-1995 . *Pittsburgh, PA, Carnegie Mellon University*, 1997.

[15] Xia Jiang and Tracy Camp. A Review of Geocasting Protocols for a Mobile Ad Hoc Network. *Grace Hopper Celebration*, 2002.

[16] Frank Kargl. Sicherheit in Mobilen Ad hoc Netzwerken. *http://medien.informatik.uni-ulm.de/ frank/research/dissertation.pdf*, 2003.

[17] C. Karlof and D. Wagner. Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. *University of California at Berkeley, USA*.

[18] klaus Plößl. Mehrseitig sichere Ad-hoc-Vernetzung von Fahrzeugen. *Gabler Edition Wissenschaft*, 2009.

[19] L. Krause and D. Borens. Das strategische Risikomanagement der ISO 31000. *zweiteilig, ZRFG 4+5*, 2009.

[20] H. LAN and U. T. NGUYEN. A study of different types of attacks on multicast in mobile ad hoc networks. *Ad Hoc Networks.,Vol.6,No.1*, 2007.

[21] Christian Maihöfer. A Survey Of Geocast Routing Protocols. *IEEE Communications*, 2004.

[22] Josiane Nzouonta, Neeraj Rajgure, and Guiling Wang. VANET Routing on City Roads using Real-Time Vehicular Traffic Information. *IEEE*.

[23] Dipl.-Ing. Klaus Oertel. Car-2Car Kommunikation und VANET. *Carl Hanser Verlag München*, pages 40–43, 2008.

[24] Charles Perkins. Ad Hoc Networking - An Introduction. *Pearson Studium, München*, 2000.

[25] Charles E. Perkins and Elizabeth M. Belding-Royer. International Journal of Network Management. *John Wiley and Sons, Inc. New York, NY, USA*, pages 97–114, 2003.

[26] K. Plößl and H. Federrath. Vorschlag für eine Sicherheitsinfrastruktur für Vehicular Ad Hoc Networks. *Universität Regensburg*, 2009.

[27] Maxim Raya and Jean-Pierre Hubaux. Security Aspects of Inter-Vehicle Communications. *Conference paper Swiss Transport Research Conference* , 2005.

[28] W.Wang and B. Bhargava. Visualization of Wormholes in Sensor Networks. *Proc. of the 2004 ACM Workshopf on Wireless Security, ACM Press, Philadelphia, PA, USA*, pages 51–60, 2004.

[29] Y. XIAO, X. SHEN, and D.-Z. Du. A Survey on Attacks and Countermeasures in Mobile Ad Hoc Networks. *WIRELESS/MOBILE NETWORK SECURITY*, 2006.