Routing in Sensornetzen - Angriffe auf ausgewählte Protokolle und Lösungsansätze

Nadine Herold
Betreuer: Alexander Klein
Seminar Sensorknoten: Betrieb, Netze und Anwendungen SS2010
Lehrstuhl Netzarchitekturen und Netzdienste, Lehrstuhl Betriebssysteme und Systemarchitektur
Fakultät für Informatik, Technische Universität München
Email: nadine.herold@onlinehome.de

KURZFASSUNG

Die Nutzung von Sensornetzwerken und deren Bedeutung nimmt immer weiter zu. Dies führt unter anderem zu vermehrten Angriffen auf diese Netzwerke. In diesem Paper sollen verschiedene Angriffe vorgestellt und deren Auswirkungen auf bestimmte Protokolle untersucht werden. Wichtige Fragestellungen, wie die Klassifizierung des Angreifers, die Möglichkeiten des Angreifers im Netz und die Erkennung des Angriffs werden betrachtet. Zudem soll ein Überblick über Gegenmaßnahmen aufzeigen, wie ein Schutz gegen diese Angriffe möglich ist und welche Protokolle anfällig für Attacken sind.

Schlüsselworte

AODV, SBR, MCFA, Wurmloch, Sybil-Attacke, Sinkhole-Attacke, Sensornetze, Routing, Angriffe, Gegenmaßnahmen

1. EINLEITUNG

Sensornetze, d.h. räumlich verteilte Netze von programmierbaren und mit Sensoren ausgestattete Recheneinheiten, gewinnen durch ihre Vielseitigkeit zunehmend an Bedeutung. Als kostengünstige Einheit lassen sie sich in vielen Bereichen leicht einsetzen. Die große Masse der Sensorknoten ist neben der Mobilität der einzelnen Knoten eine wichtige Eigenschaft, daher nimmt das Routing in einem solchen System einen hohen Stellenwert ein [5].

Die Abgrenzung zu den klassischen Ad-hoc Netzen ist fließend und daher sehr schwierig. Innerhalb eines Sensornetzes werden mehr Knoten eingesetzt als ein einem Ad-hoc Netz. Auf Grund der Beweglichkeit der Sensoren ändert sich auch die Topologie des Netzes schneller als ein einem klassischen Ad-hoc Netz. Zudem kann ein Sensorknoten nur auf beschränkte Ressourcen zurückgreifen, sowohl an Rechenleistung als auch an Speicherplatz [5].

Diese Einschränkungen haben dazu geführt, dass die herkömmlichen MANET-(mobile ad-hoc networks) Protokolle für Ad-hoc Netzwerke durch speziell für Sensornetze entwickelte Protokolle ersetzt werden sollten. Einige Studien zeigen jedoch, dass sich die MANET Protokolle im praktischen Einsatz besser eignen als spezielle Sensornetzprotokolle [5, 29]. Hier hat sich vor allem das Ad-hoc On-Demand Distance Vektor Routing Protokoll (AODV) bewährt, daher wird es in die Betrachtungen einbezogen.

In der folgenden Arbeit werden nun zunächst die einzelnen Protokolle AODV, SBR (Statistic-Based Routing) und MCFA (Minimum Cost Forwarding Algorithmus) betrachtet. Im folgenden Kapitel wird der Angreifer beschrieben und seine möglichen Attacken betrachtet. Es folgt eine Auswertung, wie sich mögliche Attacken auf die vorgestellten Protokolle auswirken. Das letzte Kapitel beschäftigt sich mit Gegenmaßnahmen zum Schutz des Sensornetzes. Eine Zusammenfassung und Auswertung der Ergebnisse bilden den Schluss der Arbeit.

2. ROUTINGPROTOKOLLE

In diesem Kapitel soll ein kurzer Überblick über die Routing-Protokolle AODV [22], SBR [12] und MCFA [28] geschaffen werden. Es werden nur die grundlegenden Mechanismen erklärt. Details werden behandelt, wenn diese für einen späteren Angriff wichtig sind.

2.1 AODV

Das AODV Routing Protokoll folgt dem reaktiven on-demand Ansatz, d.h. Routen werden erst bei Bedarf etabliert. Es gibt daher keine periodischen Updates der Routingtabellen. Eine vollständige Sicht auf die Netztopologie steht den Knoten nicht zur Verfügung [15]. Wird nun eine Route benötigt, sendet der Quellknoten einen sogenannten Route Request (RREQ) in das Netz (broadcast). Dieser enthält u.a. die Quelladresse, eine Broadcast ID und die Zieladresse. Die Quelladresse und die Broadcast ID identifizieren jeden RREQ eindeutig und verhindern Replay-Angriffe sowie unnötige Netzlast [22].

Empfängt ein Zwischenknoten, der keine Route zum Ziel kennt, einen RREQ, dann wird der RREQ und zusätzlich die eigene ID des Zwischenknotens weiter geleitet. Der Zwischenknoten merkt sich neben dem RREQ auch den Knoten, von welchem er den RREQ erhalten hat. Dieser Vorgang nennt sich Reverse Path Setup [22]. Kennt der Zwischenknoten eine Route zum Ziel oder handelt es sich um den Zielknoten selbst, wird mit einem Route Reply (RREP) geantwortet. Dieser enthält u.a. die Ziel- und Quelladresse. Der RREP wird an den Knoten gesandt, von welchem der letzte RREQ kam. Nun beginnt das Forward Path Setup. Der RREP wird an den Knoten gesandt, von welchem der ursprüngliche RREQ weitergeleitet wurde. Die Zwischenknoten konfigurieren entsprechend ihre Routingtabellen.

Der Protokollverlauf ist in Abbildung 1 dargestellt. Die erste Abbildung zeigt das Netz, der Knoten A möchte mit G

kommunizieren. A sendet einen RREQ aus, den B und C empfangen. B leitet den RREQ weiter an D und F, C leitet den RREQ weiter an E. E erhält zusätzliche RREQ von D und F, die verworfen werden und leitet den RREQ weiter an F. G sendet einen RREP über den Pfad E-C-A. Die Kommunikation zwischen A und G ist nun möglich.

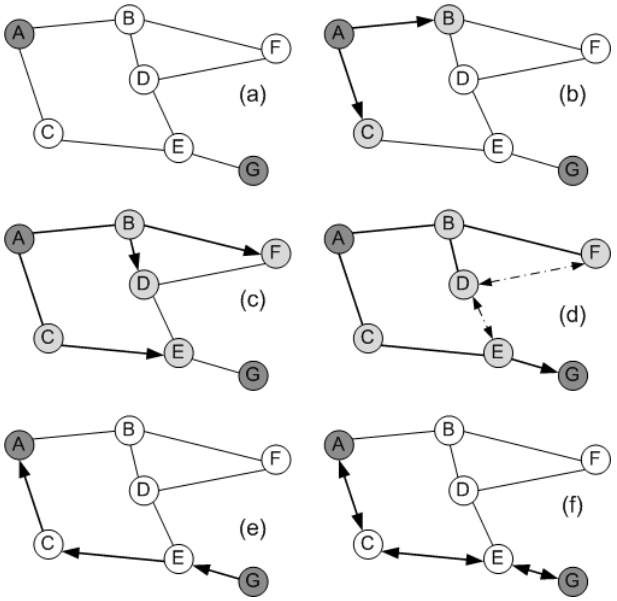

Abbildung 1: Kommunikation von A nach G

Die etablierten Routen bleiben in den Routingtabellen, solange sie aktiv sind oder bis ein Zwischenknoten ausfällt. Die Prüfung der Aktivität kann über einen Timeout geregelt werden, sodass eine Route nach einer festen Zeitspanne gelöscht wird [22]. Fällt ein Zwischenknoten aus wird eine Route Error Nachricht (RERR) zum Quellknoten gesendet [24].

Eine Erweiterung des Protokolls wird in [15] vorgestellt. AODV-BR legt während des Routenfindungsprozesses alternative Routen an, sodass im Falle eines Knotenausfalls der Datenverkehr fortgesetzt werden kann. Fällt ein Knoten aus, werden die Daten über die Alternativrouten umgeleitet. Gleichzeitig wird eine neue Route gesucht, um suboptimale Pfade zu verhindern [15].

2.2 SBR

Im Gegensatz zum AODV Protokoll wurde das SBR Protokoll speziell für den Einsatz in Sensornetzwerken entwickelt. Ziel war es hier, die Energiebeschränkungen einzuhalten und wenig Overhead und Rechenaufwand zu generieren [1].

Das SBR-Protokoll basiert auf der Verwendung von HELLO-Nachrichten. Jeder Knoten sendet periodisch HELLO--Nachrichten mit folgenden Informationen aus: Quellknoten, d.h. die eigene Adresse, eine Sequenznummer zur Identifizierung der HELLO-Nachricht, die von der Quelle bei neuen HELLO-Nachrichten inkrementiert wird, und ein Time-To-Live Feld, um zu steuern, wie lange eine HELLO-Nachricht weitergeleitet wird. In einem Intermediate-Feld wird der Knoten gespeichert, welcher die HELLO-Nachricht weitergeleitet hat [13].

Tabelle 1: Routingtabelle des Knoten A

Knoten A	empfangene HELLO's				
Quelle	B	C	D	E	F
B	50	-	-	-	-
C	-	60	-	-	-
D	9	-	-	-	-
E	-	36	-	-	-
F	40	-	-	-	-

Jeder Knoten besitzt eine Routing-Tabelle mit den ihn bekannten Sensorknoten (Quellen der HELLO-Nachrichten). Zusätzlich wird vermerkt, über welche Knoten diese Quellen zu erreichen sind. Ein generischer Wert gibt Aussage über die Linkqualität zur Quelle über den Zwischenknoten [12].

Erhält der Knoten einen HELLO-Request von einer Quelle, die noch nicht in der Routing-Tabelle auftaucht, wird eine neue Zeile und Spalte für die Quelle angelegt. Ist die Quelle der HELLO-Nachricht bekannt, werden die Sequenznummern verglichen. Falls der empfangene Wert größer ist als der gespeicherte, wird der generische Wert erneuert und das HELLO-Packet weitergeleitet. Sind beide Werte gleich, wird die TTL betrachtet. Ist die TTL des empfangenen Pakets kleiner als die gespeicherte, dann wird es weitergeleitet. In allen anderen Fällen, wird das Paket verworfen. Es ist zu beachten, dass HELLO-Nachrichten weitergeleitet werden, wenn sie über den besten Nachbarn empfangen wurden [13].

Mit Hilfe einer Decrease Routing Value Function (DRVF) und einer Increase Routing Value Function (IRVF) kann zusätzlich auf Topologieänderungen und andere Störungen reagiert werden. Die DRVF wird in regelmäßigen Intervallen, den Descrease Routing Value Interval (DRVI), angewandt und senkt die Linkqualität, sodass es nicht zu einem stetigen steigen der Linkqualität kommt. Die IRVF kommt beispielsweise bei dem Erhalt neuer Informationen über die Linkqualität zum Einsatz und steigert die Linkqualität. Im Gegensatz zur DRVF wird diese Funktion ereignisgesteuert und nicht in regelmäßigen Abständen ausgeführt. Eine Routingtabelle ist in Tabelle 1 dargestellt. Das zugehörige Netz ist das selbe wie in Abbildung 1.

2.3 MCFA

Der MCFA ist speziell für die Bedürfnisse in Sensornetzwerken entwickelt worden. Er basiert auf dem Kostenfeldkonzept, d.h. jeder Sensorknoten besitzt Informationen darüber, wie „teuer“ es ist, eine Nachricht zur Basisstation zu senden [28]. Es handelt sich um ein proaktives Protokoll, d.h. Routen sind vorhanden, bevor ein Paket gesendet werden soll [9].

Das Vorgehen teilt sich in zwei Phasen: die Initialisierung und die operative Phase [6]. Die Initialisierung beschreibt das Anlegen des Kostenfelds, d.h. alle Sensorknoten legen zunächst ein Feld mit dem Wert ∞ an, welches die Kosten zur Basis beschreibt. Die Basis sendet eine sogenannte Advertisement-Nachricht (ADV) mit dem Wert 0 aus. Unter der Verwendung des Backoff-Based Cost Field Establishment Algorithmus errechnen die Sensoren die Kosten, z.B. als Anzahl Hops, zur Basis dann wie folgt [28]:

Zuerst erhält ein Sensor eine ADV direkt von der Basis. Dann speichert er sich die Kosten C, die beim Senden dieser Nachricht entstanden sind, in seinem Kostenfeld und wartet die Zeitspanne $\alpha * C$. α ist eine Konstante die je nach Anforderungen und Infrastruktur des zu Grunde liegenden Netzes gewählt werden muss. Empfängt er in dieser Zeit keine relevanten Pakete, sendet er den ADV weiter und ändert aber den Wert von 0 auf C. Falls der Sensorknoten in der Wartezeit ein Paket von einem anderen Sensorknoten mit den Kosten C' empfängt, wird folgendes gepüft: Gilt C' zuzüglich der Kosten zwischen den Sensorknoten ist echt kleiner als das gespeicherte C, dann wird der Wert auf die angegebene Summe verändert. Eine weitere Wartezeit wird errechnet, in der auf neue Pakete gewartet wird [28].

Ist das Feld etabliert, kann nun das Routing beginnen. Ein Beispiel für ein etabliertes Kostenfeld ist in Abbildung 2 zu finden. Die Linkqualität der einzelnen Verbindungen findet sich auf den Pfeilen. Der Wert des Kostenfeldes für jeden Knoten ist neben dem Knoten angezeigt. Je höher der Wert auf den Pfeilen, desto größer sind die Kosten für die Übertragung. Niedrige Werte in den Kostenfeldern bedeuten geringere Übertragungskosten.

Beim Senden, werden zwei Felder an die Nachricht zugefügt: die minimalen und die bisher verbrauchten Kosten. Erhält ein Zwischenknoten eine solche Nachricht, prüft er, ob die bisher verbrauchten Kosten abzüglich der entstandenen Kosten für den letzten Hop immer noch größer oder gleich seiner minimalen Kosten zur Basis sind. Trifft dies zu, broadcastet der Knoten das Paket. Sonst wird es verworfen [28].

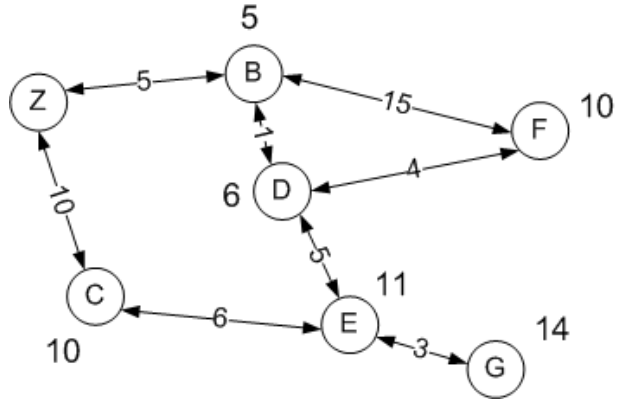

Abbildung 2: Ein etabliertes Kostenfeld für Z

Vorteilhaft bei dieser Methode ist, dass keine Routingtabellen verwendet werden müssen, die viel Speicher benötigen. Zudem sind keine ID's für die einzelnen Knoten notwendig. Durch die Verwendung der minimalen Kosten, ist auch ein optimaler Pfad zur Basis garantiert [21].

Dies zeigt jedoch einen Nachteil des Protokolls. Es kann nur für Wege von den einzelnen Sensorknoten zur Basisstation genutzt werden. Problematisch gestalten sich auch Knotenausfälle und Topologieänderungen, sowie Pfade mit gleichen Kosten [6].

3. ANGREIFER UND ANGRIFFE

Im nun folgenden Kapitel sollen zunächst unterschiedliche Angreifermodelle vorgestellt werden. Im Anschluss werden verschiedene Angriffe betrachtet.

3.1 Angreifermodelle

Man muss zunächst zwischen Insider- und Outsider-Attacken unterscheiden. Ein Outsider ist nicht Teil des Sensornetzes. Er kann aber als passiver Angreifer die Kommunikation der Knoten abhören (eavesdropping). Aktiv kann er Pakete abfangen, verändern oder neue Pakete in das Netz einschleusen und Sensorknoten physisch beschädigen [25].

Gelingt es dem Angreifer einen Knoten im Sensornetz unter seine Kontrolle zu bringen, oder einen neuen Knoten einzuschleusen, spricht man von einer Insider-Attacke. Neben den Möglichkeiten des Outsiders, kann der Angreifer nun auch an der Kommunikation im Netz teilnehmen und alle Funktionen eines normalen Knotens ausführen [25].

Eine weitere Unterscheidung ist nach den Ressourcen zu treffen. Karlof et. al. unterscheiden in [10] in mote-class und laptop-class Angreifer. Die erste Klasse beschreibt jene Angreifer, welche die gleichen Ressourcen (Rechenleistung, Speicher, etc.) zur Verfügung haben, wie auch die Sensorknoten. Ein Angreifer der laptop-class hingegen hat mehr Rechenleistung und kann somit auch mehr Angriffe auf das Sensornetz starten [10].

3.2 Direkte Angriffe auf das Routing

Die meisten Angriffe auf Sensornetze lassen sich nach [10] in eine der folgenden Kategorien einteilen:

- Manipulation der Routinginformationen
- Selektives Weiterleiten
- Manipulation mit HELLO-Paketen
- Manipulation von Infrastruktur- und Link-Layer Wissen
- Sinkhole-Attacken
- Sybil-Attacken
- Wurmlöcher

Die ersten vier Angriffsarten führen direkt zu Schäden innerhalb der Netzes. Die letzen drei wirken sich nicht direkt negativ auf das Netz aus, sondern müssen mit anderen schädigenden Aktionen verbunden werden. Ihre bloße Existenz stellt aber keine Einschränkung für das Netz dar.

Die Manipulation von Routinginformationen umfasst ändern, löschen, wiedereinspielen oder spoofen von Routingpaketen. Der Angreifer kann durch geschicktes Manipulieren Routingschleifen konstruieren, direkt den Paketfluss steuern, Routen festlegen, die Netzlast steigern oder das Netzwerk partitionieren [10].

Das selektive Weiterleiten unterscheidet sich in Grey- und Black-Holes, d.h. Nachrichten werden zufällig oder nach bestimmten Kriterien weitergeleitet oder alle Nachrichten werden gelöscht. Diese Attacke wird meist mit Angriffen wie Sinkholes, Sybil-Attacken und Wurmlöchern kombiniert [10].

HELLO-Pakete werden verwendet, um festzustellen, welche Knoten in Reichweite und somit Nachbarn sind. Dies stellt allerdings einen Angriffspunkt dar, da der Angreifer solche Pakete in das Netz fluten kann. Dies kann durch Wiedereinspielen (Replay) an gleicher oder anderer Stelle geschehen oder durch das komplette Fälschen der Pakete. Solche Angriffe bringen die gesamte Topologie durcheinander und stören zudem die Datenflusskontrolle [10].

Basieren Routingstrategien auf dem Wissen der darunterliegenden Ebenen kann dies ebenfalls von einem Angreifer ausgenutzt werden. Informationen wie Adressen der Knoten können vom Angreifer gefälscht werden und somit das Routing stören. Es kann auch versucht werden Knoten auszuschließen, indem sie als unerreichbar erscheinen oder nur eine schlechte Verbindung zu ihnen vorliegt. Solche Manipulationen können auch zur Realisierung anderer Angriffe wie selektivem Weiterleiten verwendet werden [10].

Eine Sinkhole-Attacke wird dazu verwendet den gesamten Netzverkehr über einen Knoten (Senke) lenken. Die Senke erscheint für die Sensorknoten besonders attraktiv, z.B. durch das Bereitstellen einer guten Verbindung zur Basisstation. Je nach Protokoll gibt die Senke verschiedene Eigenschaften vor, nach denen von den Sensorknoten der nächste Hop ausgewählt wird [10]. Dieser Angriff wird besonders gefährlich, wenn die Attacke mit anderen schadhaften Aktionen, wie Abhören oder selektiven Weiterleiten verbunden wird [14].

Bei einer Sybil-Attacke täuscht der Angreifer mehrere Identitäten mit nur einem physischen Knoten vor [3]. Der Angreifer kann entweder Identitäten generieren oder stehlen. Er hat auch die Möglichkeit seine Identitäten simultan oder nicht-simultan einzusetzen. Simultan bedeutet, dass es mehrere falsche Identitäten gleichzeitig im Netz gibt. Im nichtsimultanen Angriff wechseln die verschienden Identitäten. Eine Abwandlung der Sybil-Attacke ist die Replikation der gleichen Identität, der Angreifer gibt vor, an mehreren Stellen im Netz gleichzeitig zu sein. Eine solche Attacke kann sich bei Missbrauch nicht nur auf das Routing, sondern unter anderem auch auf verteilte Speichersysteme, Datenaggregation und Ressourcenverteilung auswirken [18].

Für eine Wurmloch-Attacke braucht der Angreifer zwei Knoten. Mit Hilfe des einen Knotens empfängt er Daten im Netz. Anschließend tunnelt er die gesammelten Daten an den zweiten Knoten. Besonders effektiv wird dieser Angriff, wenn der zweite Knoten in der Nähe der Basisstation liegt. Wie bereits erwähnt führt dieser Angriff nicht zu direkten Schäden, bringt aber den Angreifer in eine strategisch überlegene Position [8].

3.3 Indirekte Angriffe

Neben den bereits vorgestellten direkten Angriffe auf das Routing, die sich auf Schwachstellen in den einzelnen Routingprotokollen stützen, gibt es auch Angriffe auf Sensornetze, die nicht den Routingmechanismus an sich angreifen, ein Routing aber unmöglich machen, wie z.B. Jamming oder Denial-of-Service (DoS) Angriffe.

Beim Jamming sendet der Angreifer Radiosignale aus, welche die ausgesandeten Daten der Sensorknoten überlagern. Der Empfänger ist dann nicht mehr in der Lage, die übertragenen Informationen zu entnehmen. Der Angreifer benötigt hierfür ein stärkeres Signal als der Sender. Bei Sensornetzen stellt dies allerdings kein großes Hindernis dar [17].

Es wurden bereits verschiedene Ansätze wie das Frequency Hopping Spread Spectrum oder das Direct-Sequence--Spread-Spectrum entwickelt, die es dem Jammer erschweren das Signal zu stören und auch für Sensornetze einsetzbar sind. Zusätzlich gibt es auch für Sensornetze zugeschnittene Lösungen wie JAM oder JAID [17].

Mit Hilfe des Jammings kann eine weitere Bedrohung für das Routing in Sensornetzen realisiert werden: DoS-Angriffe. Bei einem solchen Angriff, ist das Netz nicht mehr in der Lage seine normale Aufgabe zu erledigen, sondern kommt zum Erliegen. Dies kann durch Angriffe wie Jamming, Fluten des Netzes mit Paketen oder auch Ausnutzen von Schwachstellen in Protokollen auf dem gesamten Protokollstack realisiert werden [27].

Um Protokolle resistent gegen DoS-Angriffe zu machen, werden oft kryptografische Verfahren und Authentifizierung verwendet. In Sensornetzen ist dieses Vorgehen jedoch nicht praktikabel [27].

4. ANGRIFFSSZENARIEN AUF AODV, SBR UND MCFA

Nachdem nun sowohl die Protokolle als auch die einzelnen Angriffe vorgestellt wurden, wird nun behandelt, welche Angriffe überhaupt in den Protokollen möglich sind und wie das Netz und Routingverhalten darauf reagieren.

4.1 Angriffe auf AODV

Ein wichtiger Angriffspunkt im AODV Protokoll sind die RREQ und RREP sowie RERR. Durch geschickte **Manipulation dieser Routinginformationen** lassen sich existierende Routen stören, einen manipulierten Knoten in eine Route einfügen oder auch zusätzliche Ressourcen verbrauchen [20]. Durch falsche RERR's lassen sich gezielt Ressourcen verbrauchen oder vorhandene Routen unterbrechen, da der Knoten mit einer scheinbar zerstörten Verbindung wieder eine neue Route suchen muss. Um einen Knoten in eine bestehende Route einzufügen, kann beispielsweise das Vorgehen in Abbildung 3 verwendet werden. Durch geschicktes Fälschen der Nachrichten wird die Route über den Angreifer umgeleitet. Das selektive Weiterleiten ist nun für den Angreifer möglich.

Weitere Möglichkeiten der Fälschung innerhalb der Nachrichten bietet die Broadcast-ID. Fälscht der Angreifer ein Paket eines Knotens und verwendet dabei eine sehr hohe Broadcast-ID, werden alle Pakete, die von dem betroffenen Knoten stammen ignoriert, bis dieser die gefälschte Sequenznummer erreicht hat. Auf diese Weise können Knoten isoliert, Ressourcen verbraucht und die Netzlast gesteigert werden. Erkennt ein Knoten einen zu großen Sprung in den ID's, so könnte er das Paket abweisen und eine entsprechende RERR an den betroffenen Knoten senden. Dies erhöht jedoch die Netzlast.

Durch das vorzeitige Senden eines RREP durch den An-

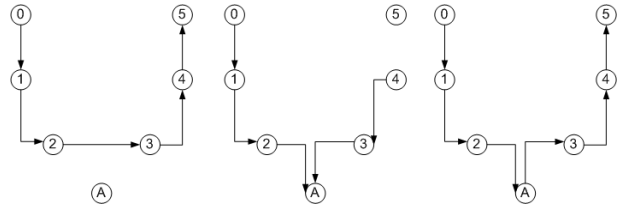

Abbildung 3: Ein Angriff auf AODV mittels Manipulation der Routinginformationen. Bild nachgezeichnet aus [20]

greifer kann zudem ein schwarzes Loch, ein Sonderfall des selektiven Weiterleitens, entstehen [16]. Dabei tarnt sich der Angreifer als Zielknoten des RREQ oder als Zwischenknoten mit einer aktiven Verbindung zum Ziel. Weitere Möglichkeiten für das **selektive Weiterleiten** innerhalb des AODV Protokolls bestehen durch die Anwendung der Wurmloch- oder Sinkhole Attacke.

Um ein **Wurmloch** zu erzeugen, wird der originale RREQ direkt zum Ziel getunnelt und dann dort an das Ziel weitergeleitet. Das Ziel antwortet nach Protokollvorschrift mit einem RREP, der dann über das Wurmloch zurück an den Initiator des RREQ getunnelt wird. Alle später eintreffenden RREP's, welche nicht über das Wurmloch laufen würden, werden dann vom Zielknoten verworfen, da dieser bereits eine Verbindung über das Wurmloch etabliert hat [8].

Da ein **Sinkhole-Angriff** auch über ein Wurmloch realisiert werden kann, ist AODV folglich auch anfällig für diesen Angriffstyp [19]. Darüber hinaus kann durch das Fälschen der Hop Counts eine Sinkhole Attacke durchgeführt werden. Hier erscheint ein Sensorknoten als besonders attraktiv für eine Route, wenn er einen geringen Hop Zähler aufweisen kann [4].

Das AODV Protokoll greift auf Identitäten zurück. Daher ist es prinzipiell möglich eine **Sybil-Attacke** zu starten. Durch das Vorspielen mehrerer Identitäten werden die Routingtabellen in den einzelnen Knoten vergrößert. Dies führt künstlich zu einem erhöhten Speicherverbrauch. Das Bilden von Alternativrouten z.B. bei der Erweiterung AODV-BR kann gestört werden, da eine scheinbare Alternativroute dann der Originalroute entspricht. Darüber hinaus wird bei Ausfall des Sybil-Knotens das Netz mit RERR-Nachrichten geflutet.

In [20] weisen Ning und Sun darauf hin, dass eine Verwendung von HELLO-Paketen bei der Nutzung von AODV nicht zwingend nötig ist. Daher ist auch ein **Fluten mit HELLO-Pakten vermeidbar**. Auch auf Infrastrukturwissen wird nicht zurückgegriffen und kann somit auch nicht als Grundlage für einen Angriff dienen.

4.2 Angriffe auf SBR

Das SBR Protokoll basiert auf dem Austausch von **HELLO-Paketen**, welche zum Austausch für **Routinginformationen** verwendet werden. Diese Informationen können allerdings vom Angreifer manipuliert werden. Durch das Abfangen und Ändern ausgewählter Nachrichten kann der Angreifer den Netzverkehr lenken. Indem er eine hohe Linkqualität vorgibt, werden Informationen an bestimmte Knoten gesandt. Eine niedrige Linkqualität führt dazu, dass ein Knoten nicht angesprochen wird.

Flutet ein Angreifer das Netz mit HELLO-Paketen, kann er den Energieverbrauch der Knoten erhöhen, da sie bei Erhalt von HELLO-Paketen ihre Routingtabellen anpassen müssen. Durch die Anwendung der DRVF und IRVF bei Erhalt der HELLO's ist der Knoten gezwungen unnötig Rechenleistung und Energie zu verbrauchen. Dieser Mechanismus, der eigentlich für eine hohe Adaption des Netzes genutzt werden soll, kann in diesem Fall das Netz erheblich schwächen.

Durch das Vorgeben einer hohen Linkqualität kann der Angreifer ein **Sinkhole** erzeugen. Er hat die Möglichkeit, seine eigenen Knoten mit hoher Linkqualität anzupreisen, sodass diese zu einer Senke werden. Er kann auch Nachrichten spoofen und somit andere Knoten im Netz zur Senke machen. Diese Knoten werden dann besonders mit Netzverkehr belastet. Dies kann zu einem Ausfall der betroffenen Knoten führen oder den Netzverkehr erheblich verlangsamen.

Eine hohe Linkqualität kann auch durch ein **Wurmloch** verursacht werden. Wird die Linkqualität beispielsweise durch die Nähe zur Basis bestimmt, werden die beiden Angreiferknoten im normalen Protokolleinsatz zur bevorzugten Route.

Durch Verwendung vom Senken oder Wurmlöchern bringt sich der Angreifer in eine sehr starke Position. Er ist nach erfolgreichem Angriff in der Lage, **selektiv Nachrichten weiterzuleiten** oder auch ein schwarzes Loch zu erzeugen. Er muss aber darauf achten, dass trotz der Verwendung der IRVF und DRVF seine Position bestehen bleibt. Daher muss er im gesamten laufenden Betrieb Maßnahmen zum Erhalt seiner Position durchführen.

Das Verändern der Routinginformationen ist vor allem bei der Sequenznummer von Bedeutung. Verwendet der Angreifer eine hohe Sequenznummer, muss der betroffene Knoten auch hier sehr viele Nachrichten versenden, bis die anderen Knoten wieder auf seine Nachrichten reagieren. Die Ressourcen des Knotens werden so verbraucht und der Angreifer ist in der Lage, bestimmte Knoten für eine gewisse Zeit aus dem Netz zu isolieren. Eine Möglichkeit wäre der Vergleich der aktuellen Sequenznummer mit der letzten erhaltenen. Ist der Unterschied zu groß, wird das Paket nicht angenommen. Eine Möglichkeit, dies dem Sender mitzuteilen besteht allerdings nicht.

Wenn bei der Berechnung der Linkqualität auf Informationen auf den unteren Schichten zurückgegriffen wird, stellt auch dieses Wissen einen Angriffspunkt dar. Eine Ausnutzung ist dann abhängig von der Nutzung der Informationen sowie der DRVF und IRVF.

Auch ein **Sybil-Attacke** ist potentiell möglich. Die Routingtabellen der Knoten beziehen sich auf die ID's anderer Knoten, d.h. die Identitätsinformation ist essentiell wichtig für das Funktionieren des Protokolls. Zusätzliche Identitäten verlängern die Routingtabellen und erhöhen so den Speicherbedarf und Rechenaufwand für die einzelnen Knoten, um diese Tabellen zu warten. Insgesamt wird so auch

der Energieverbrauch erhöht. Zudem würden real existierende Knoten aus der Tabelle verdrängt werden und somit für den Knoten nicht mehr erreichbar sein.

4.3 Angriffe auf MCFA

Die Möglichkeit **Routinginformationen zu manipulieren** ist dem Angreifer nur in der Initialisierungsphase möglich, indem er ADV-Nachrichten löscht, ändert, spooft oder wiedereinspielt. Das Wiedereinspielen beeinflusst das Routing allerdings nicht, da die Kosten nur geändert werden, wenn eine Verbesserung vorliegt. Durch das Löschen, Ändern oder Spoofen von ADV's ist es möglich, dass die minimalen Kosten für den Nachrichtenversand zu hoch kalkuliert werden und im späteren Produktivbetrieb auch Sensorknoten die Nachrichten weiterleiten, die nicht auf dem optimalen Pfad liegen. Neben einer erhöhten Netzlast hat dies auch Routingschleifen zur Folge.

Karkof et. al. beschreiben in [10] das auch eine Attacke über das Fluten des Netzes mittels **HELLO-Paketen** möglich ist. Mit einem Laptop werden Pakete in der Initialisierungsphase versandt, die Kosten von 0 zur Basis versprechen. Nun müssen die Kosten der Nachricht für jeden Knoten nur niedriger gehalten werden als dessen bisherige Kosten. Werden dann Nachrichten versandt, wird kein anderer Knoten als der Angreifer selbst, die Nachrichten weiterleiten können [10].

Befindet sich der Angreifer innerhalb des Netzes auf einem minimalen Pfad, von der Quelle zur Basis kann er auch **Nachrichten selektiv weiterleiten** oder die Kommunikation blockieren. Um dies zu erreichen könnte der Angreifer vorher eine **Sinkhole Attacke** benutzen. Hierfür ist ein Knoten im Netz ausreichend, welcher Kosten von 0 zur Basisstation propagiert. Der zugrundeliegende Algorithmus für die Routenerstellung versucht die Kosten zu minimieren und wird sich für den Knoten mit Kosten von 0 zur Basis entscheiden. Die Sinkhole Attacke muss aber schon in der Initialisierungsphase durchgeführt werden um im Produktivbetrieb genutzt werden zu können [10].

Eine weitere Möglichkeit das Netz zu stören besteht in der Anwendung eines **Wurmlochs**. Der Angreifer muss in diesem Fall einen leistungsstärkeren Knoten aufweisen, wie beispielsweise einen Laptop [10]. Auch in diesem Szenario muss der Angriff in der Initialisierungsphase geschehen. Er wird ebenfalls durch das Propagieren von Kosten von 0 zur Basis umgesetzt.

Für die einzelnen Knoten sind keine ID's nötig sind, folglich ist eine **Sybil-Attacke nicht anwendbar**. Da nicht auf Link Layer- oder Infrastrukturwissen zurückgegriffen wird, führt auch hier eine Manipulation nicht zu einer erfolgreichen Störung des Netzes.

5. GEGENMAßNAHMEN

Die Nutzung von Kryptografie und global verteilten Schlüsseln ist eine Möglichkeit Gegenmaßnahmen zu ergreifen [10]. Durch die geringe Leistungsfähigkeit sind diese Ansätze im Bereich Sensornetze jedoch nicht vorteilhaft. Im Folgenden sollen nun einige weiterführende Ansätze besprochen werden. Zu den vorgestellten Angriffen existieren einige Frameworks zur Behandlung der Probleme. Hinzu kommen Ansätze für Protokolle, die speziell für Sensornetzwerke mit Hinblick auf Sicherheit entwickelt wurden. Auf Grund der Menge an vorhandenen Lösungsansätzen wird hier nur eine kleine Auswahl der wichtigsten Ansätze vorgestellt.

5.1 Frameworks für spezielle Angriffe

Eine Möglichkeit um Wurmlöcher zu verhindern sind sog. Packet leashes, d.h. zusätzliche Informationen, die die Sendedistanz einschränken [8]. Man unterscheidet zwischen temporalen, hier wird die Sendezeit angefügt, und geografischen, zusätzlich wird die eigene Position übertragen, Leashes. Die emfangenen Werte werden mit den eigenen Daten verglichen und bestimmt, ob sich das Paket zu weit bewegt hat [7]. Einen weiteren Ansatz bietet das WOMEROS Framework von Vu et. al. in [26]. In der Verdachtsphase misst jeder Knoten die Round Trip Time (RTT) zu allen seinen Nachbarn. Neben einer erhöhten RTT bringt auch die Auswertung der Nachbarn Hinweise auf ein Wurmloch. In der Bestätigungsphase wird versucht durch bestimmte Tests den Verdacht auf ein Wurmloch zu bestätigen [26]. Leider werden keine Maßnahmen genannt, das Wurmloch aus dem Netz zu entfernen.

Neben Wurmlöchern wurden auch Sinkhole-Angriffe besonders untersucht. In [19] wird ein solcher Ansatz vorgestellt. Es wird davon ausgegangen, dass eine häufige Anwendung des Sinkholes das selektive Weiterleiten von Daten ist. Daher kann das Fehlen von Daten aus immer wieder den gleichen Bereichen auf ein Sinkhole hindeuten. Mit statistischen Methoden können solche Inkonsistenzen aufgedeckt werden. Durch die Analyse der Routing-Mustern kann der Angreifer identifiziert und isoliert werden. Das genaue Verfahren zur Musteranalyse und das Adaptieren des Vorgehensmusters sind in [19] eingehend beschrieben.

Auch Maßnahmen zur Eindämmung der Sybil-Attacke wurden eingehend in [18] untersucht. Das Radio Resource Testing geht davon aus, dass jeder Knoten nur eine Antenne hat, mit der er auf einem Kanal entweder senden oder empfangen kann. Spricht man einen benachbarten Knoten auf einer bestimmten Frequenz an und dieser antwortet nicht, deutet dies auf einen Sybil-Angriff hin. Das genaue Verfahren wird in [18] erläutert. Ein weiteres Verfahren ist das Random Key Predistribution. Hierbei werden Schlüssel verwendet, die mit der Knoten-Identität verbunden werden. Jeder Knoten kann dann die Schlüssel validieren. Ein Generieren von gültigen Identitäten ist nicht mehr möglich. [18]

5.2 Sicherheitsprotokolle

Ein speziell für Sensornetzwerke entwickeltes Sicherheitssystem ist SPINS (Security Protocols for Sensor Networks) [23]. Es besteht aus zwei unabhängigen Protokollen. Das Sensor Network Encryption Protocol (SNEP) bietet Vertraulichkeit, wechselseitige Datenauthentifikation, Integrität und Datenaktualität (freshness). μTESLA wird für die Authentifikation von Datenbroadcasts verwendet.

Nötig ist hier allerdings ein vorab festgelegter Schlüssel zwischen jedem Knoten und der Basis [23]. Bezüglich Angriffe auf das Routing bietet SNEP einen Schutz gegen Replay Angriffe und Manipulation von Routinginformationen. Problematisch ist jedoch die durch kryptografische Funktionen benötigte zusätzliche Rechenleistung [23]. Der zweite Sicherheitsblock besteht aus einer Adaptierung des TESLA Pro-

tokolls μTESLA. Die vorgenommenen Anpassungen sollen Einschränkungen im Sensorbereich verglichen mit Ad-hoc Netzwerken ausgleichen [23].

Dieses Verfahren kann nun auf Ad-hoc Routing Strategien angewandt werden. Die Routingnachrichten können so authentifiziert werden, was eine Manipulation erschwert [23]. Wurmlöcher, Sinkholes und Sybil-Attacken sind weiterhin möglich, und somit auch das selektive Weiterleiten. Das Fluten mit HELLO-Paketen wird erschwert, wenn der Angreifer Pakete abfangen und verändern muss. Neue, gültige Pakete zu generieren ist nicht möglich, solange er die Schlüsselkette nicht kennt.

5.3 Intrusion Detection Lösungen

Der Ansatz von Eik et. al. in [4] basiert auf der Einbruchserkennung, d.h. es werden keine Maßnahmen zur Vermeidung von Angriffen getroffen, sondern lediglich Angriffe erkannt. Ziel ist es hier, nicht nur einen bestimmten Angriffstyp zu erkennen, sondern alle Angriffsmuster zu erfassen.

Jeder einzelne Knoten benötigt sein eigenes Intrusion Detection System (IDS). Es werden unabhängig voneinander und lokal traffic und nicht-traffic bezogene Daten gesammelt, die einen Eigenschaftsvektor bilen. Da jeder Knoten zusätzliche Daten verwalten muss, steigt der Speicherbedarf und durch die zusätzlich benötigte Rechenleistung wird auch mehr Energie benötigt. Alternativ können sog. monitoring nodes verwendet werden, deren einizige Aufgabe die Netzüberwachung ist [4]. Ein Einbruch wird dann erkannt, wenn das aktuelle Netzverhalten nicht mit dem normalen, vorher erlernten Netzverhalten übereinstimmt [4].

Ein weiterer ganzheitlicher Ansatz ist das DICAS (Detection, Diagnosis and Isolation of Control Attacks in Sensor Networks) Framework. Es soll u.a. Wurmlöcher, Sinkholes, Sybil-Attacken und das Fluten mit HELLO-Paketen erkennen und betroffene Knoten isolieren [11]. Ein Knoten sucht zunächst alle seine Nachbarn. Dann beobachten die Knoten die Kommunikation der anderen Knoten, wenn sie beide Kommunikationspartner als Nachbarn identifiziert haben. Werden verdächtige Pakete ausgetauscht, wird ein Zähler für die jeweilige Kommunikation erhöht. Beim Überschreiten eines Schwellwertes, wird von einem Angriff ausgegangen und ein Alarm ausgelöst [11].

6. VERWANDTE ARBEITEN

Karlof et. al. lieferten bereits in [10] einen Überblick über Routingangriffe. In dieser Arbeit wurden allerdings drei andere Protokolle (AODV, SBR und MCFA) fokussiert, wobei AODV und SBR nicht in den Betrachtungen von [10] liegen. Zudem wurden in diesem Paper auch die möglichen Gegenmaßnahmen genauer betrachtet.

In [1] wird einen sehr genauer Überblick über bestehende Protokolle für den Einsatz in Sensornetzen geboten. Es werden allerdings keine möglichen Angriffen und zugehörige Gegenmaßnahmen besprochen. Auch in [9] legen die Autoren den Fokus auf die Betrachtung und den Vergleich der Routingprotokolle für Sensornetzwerke.

In den Quellen [2] und [16] wird versucht speziell das AODV Protokoll gegen Angriffe abzusichern. Diese Methoden sind allerdings nicht zwangsläufig auf andere Protokolle übertragbar und bieten so keine umfassende Lösung. Umfassende Lösungen werden in [4] und [11] beschrieben. Diese Lösungsschemata sollen einen allgemeinen Schutz gegen Angriffe bieten. Es wird jedoch keine genaue Implementierung dargestellt.

Die Lösungen aus [7] und [26] bieten wiederum nur einen Schutz vor einem bestimmten Angriffstyp und keine umfassende Betrachtung der Probleme. [23] bezieht sich nicht direkt auf mögliche Probleme durch Angriffe auf das Routing, sondern stellt allgemein ein Sicherheitsprotokoll vor.

7. ZUSAMMENFASSUNG UND AUSBLICK

In dieser Arbeit wurde dargelegt, welche Angriffe auf ausgewählte Protokolle möglich sind. Dabei hat sich gezeigt, dass kein Protokoll so konzipiert wurde, dass es allen vorgestellten Angriffsszenarien standhalten kann.

Zu AODV gibt es Verbesserungsvorschläge um konkrete Angriffe zu unterbinden. Auf SBR oder MCFA zugeschnittene Lösungen gibt es aber nicht. Allgemeine Sicherheitsprotokolle für Sensornetze können auch nur teilweise Angriffe unterbinden. In Kombination mit Routingprotokollen wirken sie vor allem der Manipulation von Routinginformationen entgegen. Dies ist zwar schon ein Schritt in die richtige Richtung, löst aber das Problem nicht vollständig.

Umfassendere Lösungen wurden ebenfalls betrachtet, doch sind mit solchen universellen Lösungen neue Probleme verbunden. Erhöhter Speicherbedarf und zusätzliche Rechenleistung, die den Energiebedarf erhöht, stellen neue Forderungen an die Hardware von Sensorknoten und das Energiemanagement.

Diese Ansätze sind zwar vielversprechend, müssen aber noch auf die Bedürfnisse von Sensorknoten angepasst werden. Eine weitere Möglichkeit ist die Entwicklung spezieller Knoten, die im Netz nur die Aufgabe übernehmen, nach Angreifern zu suchen, zu finden und dann aus dem Netz zu entfernen. Solche zusätzlichen Komponenten treiben jedoch auch die Kosten des Netzes in die Höhe, was nicht bei allen Einsatzgebieten akzeptabel ist.

Es konnte keine Lösung gefunden werden, die alle Probleme abdeckt ohne dadurch neue Herausforderungen zu schaffen. Dies zeigt, dass auch im Bereich der Sensornetzwerke noch ein erhöhter Forschungsbedarf besteht.

8. LITERATUR

[1] J. N. Al-karaki, T. H. University, A. E. Kamal, and I. S. University. Routing techniques in wireless sensor networks: A survey. *IEEE Wireless Communications Magazine*, 11(6):6 – 28, December 2004.

[2] S. Bhargava and D. P. Agrawal. Security enhancements in AODV protocol for wireless ad hoc networks. In *Vehicular Technology Conference*, pages 7–11, Atlantic City, New York, October 2001.

[3] J. Douceur. The sybil attack. In *Proceedings of the 1st International Workshop on Peer-to-Peer Systems*, pages 251–260, 2002.

[4] C. Eik, L. Mun, Y. Ng, C. Leckie, and

M. Palaniswami. Intrusion detection for routing attacks in sensor networks. *International Journal of Distributed Sensor Networks*, 2(4):313 – 332, December 2006.

[5] Y. Gadallah. A comparative study of routing strategies for wireless sensor networks: Are MANET protocols good fit? In *Ad-Hoc, Mobile, and Wireless Networks*, pages 5–18, 2006.

[6] W. D. Henderson and S. Tron. Verification of the minimum cost forwarding protocol for wireless sensor networks. *Emerging Technologies and Factory Automation, 2006. ETFA '06. IEEE Conference on*, pages 194 – 201, May 2007.

[7] Y. Hu, A. Perrig, and D. Johnson. Packet leashes: A defense against wormhole attacks in wireless networks. In *INFOCOM*, 2003.

[8] Y. C. Hu, A. Perrig, and D. B. Johnson. Wormhole attacks in wireless networks. *IEEE Journal on Selected Areas in Communications (JSAC), pp*, 24(2):370 – 380, February 2006.

[9] Q. Jiang and D. Manivannan. Routing protocols for sensor networks. *Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE*, pages 93 – 98, April 2004.

[10] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. In *First IEEE International Workshop on Sensor Network Protocols and Applications, SPNA*, 2003.

[11] I. Khalil, S. Bagchi, and C. Nita-Rotaru. DICAS: Detection, diagnosis and isolation of control attacks in sensor networks. In *IEEE Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm)*, 2005.

[12] A. Klein. Statistic-based routing SBR. Experimental 453, University of Wuerzburg, October 2008.

[13] A. Klein and P. Tran-Gia. Handover in sensor networks using statistic-based routing. *6th ITG Wireless Sensor Networks*, July 2007.

[14] I. Krontiris, T. Giannetsos, and T. Dimitriou. Launching a sinkhole attack in wireless sensor networks; the intruder side. In *WIMOB '08: Proceedings of the 2008 IEEE International Conference on Wireless & Mobile Computing, Networking & Communication*, pages 526–531, Washington, DC, USA, 2008. IEEE Computer Society.

[15] S. J. Lee and M. Gerla. AODV-BR: Backup routing in ad hoc networks. In *Proceedings of IEEE WCNC 2000, Chicago IL*, 2000.

[16] N. Mistry, D. C. Jinwala, and M. Zaveri. Improving AODV protocol against blackhole attacks. In *Proccedings of International MultiConference of Engineers and Computer Scientist, IMECS*, volume 2, Hong Kong, March 2010.

[17] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou. A survey on jamming attacks and countermeasures in WSN.

[18] J. Newsome, E. Shi, D. Song, and A. Perrig. The sybil attack in sensor networks: analysis & defenses. In *Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on*, pages 259–268, 2004.

[19] E. C. H. Ngai, J. Liu, and M. R. Lyu. On the intruder detection for sinkhole attack in wireless sensor networks. In *Communications, 2006 IEEE International Conference on*, volume 8, pages 3383–3389, 2006.

[20] P. Ning and S. K. How. How to misuse AODV: a case study of insider attacks against mobile ad-hoc routing protocols. Technical report, Computer Science Departement, North Carolina State University, March 2003.

[21] T. Padmavthy, G. Divya, and T. Jayashree. Extending network lifetime in wireless sensor networks using modified minimum cost forwarding protocol - MMCFP. *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1 – 4, October 2009.

[22] C. E. Perkins and E. M. Royer. Ad hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computer Systems and Applications*, pages 90–100, 1999.

[23] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. In *Proceedings of the Annual International Conference on Mobile Computing and Networks (MOBICOM) 2001*, pages 189–199, 2001.

[24] E. M. Royer and C. E. Perkins. An implementation study of the AODV routing protocol. In *IEEE WCNC*, 2000.

[25] E. Shi and A. Perrig. Designing secure sensor networks. *Wireless Communications, IEEE [see also IEEE Personal Communications]*, 11(6):38–43, 2004.

[26] H. Vu, A. Kulkarni, K. Sarac, and N. Mittal. WORMEROS: A new framework for defending against wormhole attacks on wireless ad hoc networks.

[27] A. Wood and J. Stankovic. Denial of service in sensor networks. *IEEE Comp*, 35(10):54–62, 2002 2002.

[28] F. Ye, A. Chen, S. Lu, and L. Zhang. A scalable solution to minimum cost forwarding in large sensor networks. In *Proc. of the IEEE Tenth International Conference on Computer Communications and Networks*, pages 304–309, 2001.

[29] Z. Zhang, H. Zhou, and J. Gao. Scrutinizing performance of ad hoc routing protocols on wireless sensor networks. *Intelligent Information and Database Systems, Asian Conference on*, 0:459–464, 2009.