TRUST AND PRIVACY IN WIRELESS SENSOR NETWORKS

PhD thesis

2009

Efthimia Aivaloglou
Department of Information and Communication Systems Engineering
University of the Aegean
Συμβουλευτική Επιτροπή

Στέφανος Γκρίτζαλης — Επιβλέπων
Καθηγητής Πανεπιστημίου Αιγαίου

Σπύρος Κοκολάκης — Μέλος
Επίκουρος Καθηγητής Πανεπιστημίου Αιγαίου

Κωνσταντίνος Λαμπρινουδάκης — Μέλος
Επίκουρος Καθηγητής Πανεπιστημίου Πειραιώς
Εξεταστική Επιτροπή

Στέφανος Γκρίτζαλης
Καθηγητής Πανεπιστημίου Αιγαίου

Σωκράτης Κάτσικας
Καθηγητής Πανεπιστημίου Πειραιώς

Κωνσταντίνος Λαμπρινουδάκης
Επίκουρος Καθηγητής Πανεπιστημίου Πειραιώς

Σπύρος Κοκολάκης
Επίκουρος Καθηγητής Πανεπιστημίου Αιγαίου

Χαράλαμπος Σκιάνης
Επίκουρος Καθηγητής Πανεπιστημίου Αιγαίου

Ιωάννης Μαριάς
Λέκτορας Οικονομικού Πανεπιστημίου Αθηνών

Εμμανουήλ Μάγκος
Λέκτορας Ιόνιου Πανεπιστημίου
# Contents

<table>
<thead>
<tr>
<th>Περιλήψη</th>
<th>xiii</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>xvii</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>xix</td>
</tr>
</tbody>
</table>

## 1 Introduction

1.1 Research Field ........................................ 2
1.2 Motivation and Research Approach ................. 3
1.3 Contribution .......................................... 5
  1.3.1 Trust Management .................................. 6
  1.3.2 Privacy Protection ................................ 6
  1.3.3 Security Management .............................. 8
1.4 Work Done in Collaboration ........................ 8
1.5 Organisation of the Thesis .......................... 9

## 2 Wireless Sensor Networks & Security

2.1 Characteristics of WSNs .............................. 11
2.2 Applications and Scenarios ......................... 14
2.3 Concepts, Requirements and Challenges ............ 15
  2.3.1 Trust ............................................. 16
  2.3.2 Privacy .......................................... 18
  2.3.3 Security ......................................... 24
2.4 Summary .............................................. 26

## 3 Related Work

3.1 Trust Management ...................................... 29
  3.1.1 Trust Properties .................................. 30
  3.1.2 Trust Management Approaches .................... 32
  3.1.3 Certificate-Based Trust Establishment ......... 35
    3.1.3.1 Hierarchical Trust Models .................. 36
    3.1.3.2 Distributed Trust Models ................... 37
    3.1.3.3 Distributed Certification Authority Models 38
  3.1.4 Behaviour-Based Trust Establishment ........... 39
    3.1.4.1 Linear Trust Computation .................... 40
    3.1.4.2 Bayesian Approaches for Trust Evaluation 42
    3.1.4.3 Belief Theory Approaches .................. 44
### 3.1.4.4 Other Approaches for Trust Evaluation

<table>
<thead>
<tr>
<th>3.1.5 Hybrid Trust Management Models</th>
<th>46</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.6 Analysis and Discussion</td>
<td>48</td>
</tr>
</tbody>
</table>

### 3.2 Privacy Protection

<table>
<thead>
<tr>
<th>3.2.1 Privacy Protection Approaches</th>
<th>52</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2.2 Protection of the Communication Context</td>
<td>54</td>
</tr>
<tr>
<td>3.2.2.1 Source Location Privacy</td>
<td>55</td>
</tr>
<tr>
<td>3.2.2.2 Network Identity Privacy</td>
<td>59</td>
</tr>
<tr>
<td>3.2.3 Privacy Sensitive Information Disclosure</td>
<td>63</td>
</tr>
<tr>
<td>3.2.3.1 Privacy Policies and Preferences for Access Control</td>
<td>63</td>
</tr>
<tr>
<td>3.2.3.2 Information Granularity Control</td>
<td>66</td>
</tr>
<tr>
<td>3.2.3.3 Protection Against Location and Identity Inference</td>
<td>67</td>
</tr>
<tr>
<td>3.2.4 Privacy Sensitive Information Gathering</td>
<td>69</td>
</tr>
<tr>
<td>3.2.4.1 Restricted Data Gathering</td>
<td>69</td>
</tr>
<tr>
<td>3.2.4.2 User Notice and Choice through User Agents</td>
<td>71</td>
</tr>
<tr>
<td>3.2.5 Analysis and Discussion</td>
<td>73</td>
</tr>
</tbody>
</table>

### 3.3 Security Management

<table>
<thead>
<tr>
<th>3.3.1 Security Primitives and Mechanisms for WSNs</th>
<th>75</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1.1 Cryptographic Key Distribution and Data Encryption</td>
<td>75</td>
</tr>
<tr>
<td>3.3.1.2 Secure Data Aggregation</td>
<td>77</td>
</tr>
<tr>
<td>3.3.1.3 Applicability of Security Primitives</td>
<td>77</td>
</tr>
<tr>
<td>3.3.2 Security Services Integration and Management</td>
<td>78</td>
</tr>
<tr>
<td>3.3.3 Discussion</td>
<td>80</td>
</tr>
</tbody>
</table>

### 3.4 Summary

| 3.4 Summary                                      | 80 |

### 4 A Hybrid Trust Management Model

<table>
<thead>
<tr>
<th>4.1 Scope and Objectives</th>
<th>85</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1.1 Trust Management Requirements and Approach</td>
<td>85</td>
</tr>
<tr>
<td>4.1.1.1 Application to WSN Deployments</td>
<td>85</td>
</tr>
<tr>
<td>4.1.1.2 Support for the Heterogeneity in the Trust Evaluation Needs</td>
<td>85</td>
</tr>
<tr>
<td>4.1.1.3 Utilisation of the Pre-Deployment Knowledge</td>
<td>86</td>
</tr>
<tr>
<td>4.1.1.4 Trust Revocation due to Malicious Behaviour</td>
<td>87</td>
</tr>
<tr>
<td>4.1.1.5 Robustness Against Attacks on the Trust Management Model</td>
<td>87</td>
</tr>
<tr>
<td>4.1.6 Generic Scope and Utilisable Results</td>
<td>88</td>
</tr>
<tr>
<td>4.1.2 Relation to Other Approaches</td>
<td>89</td>
</tr>
<tr>
<td>4.1.3 Assumptions and Limitations</td>
<td>89</td>
</tr>
<tr>
<td>4.2 Working Scenario</td>
<td>90</td>
</tr>
<tr>
<td>4.3 Hybrid Trust Management Overview</td>
<td>91</td>
</tr>
<tr>
<td>4.3.1 Trust Associations and Metrics</td>
<td>92</td>
</tr>
<tr>
<td>4.3.2 Trust Evidence</td>
<td>95</td>
</tr>
<tr>
<td>4.3.3 Configuration Parameters</td>
<td>97</td>
</tr>
<tr>
<td>4.3.3.1 Trust Thresholds</td>
<td>97</td>
</tr>
<tr>
<td>4.3.3.2 Trust Degradation Parameter</td>
<td>99</td>
</tr>
<tr>
<td>4.3.3.3 Total Referral Trust</td>
<td>100</td>
</tr>
<tr>
<td>4.3.3.4 Propagation Set</td>
<td>101</td>
</tr>
</tbody>
</table>
4.4 Structure, Components and Interfaces ............................................. 103
4.5 Trust Management Processes ......................................................... 106
   4.5.1 Trust Associations Resolution ............................................. 107
   4.5.2 Trust Establishment ............................................................. 108
   4.5.3 Trust Re-evaluation and Revocation ...................................... 111
4.6 Trust Evaluation Metrics ............................................................. 113
   4.6.1 Functional Trust Metric ....................................................... 114
   4.6.2 Referral Trust Metric ........................................................... 116
   4.6.3 Trust Re-evaluation ............................................................... 117
4.7 Analysis of the Model .................................................................. 118
   4.7.1 Evaluation Against Requirements ......................................... 118
   4.7.2 Security and Trust Properties ................................................. 120
   4.7.3 Robustness Against Attacks .................................................. 121
   4.7.4 Resource Requirements and Overheads .................................. 123
4.8 Experimental Evaluation ............................................................... 124
   4.8.1 Simulation Setup .................................................................. 125
   4.8.2 Established Trust Relationships ............................................. 126
   4.8.3 Required Trust Evaluation Operations .................................... 129
   4.8.4 Trust Revocation Operations and the Impact of Colluding Malicious Recommenders ......................................................... 130
   4.8.5 The Effect of the Configuration Parameters ............................ 131
4.9 Summary .................................................................................... 135

5 Trust–Based Data Disclosure .......................................................... 137
   5.1 Scope and Objectives ................................................................. 139
      5.1.1 Data Disclosure Requirements and Approach ...................... 139
      5.1.2 Relation to Other Approaches ............................................. 140
      5.1.3 Assumptions and Limitations .............................................. 141
   5.2 Structure, Components and Interfaces ....................................... 141
   5.3 Data Disclosure Control ............................................................ 143
   5.4 Data Cloaking Operations ......................................................... 145
      5.4.1 Negative Surveys ............................................................... 146
      5.4.2 Reduction of Data Accuracy .............................................. 147
   5.5 Experimental Evaluation ........................................................... 147
   5.6 Summary ................................................................................... 151

6 Security Management for WSNs ...................................................... 153
   6.1 Scope and Objectives ................................................................. 154
      6.1.1 Security Management Requirements and Approach ............ 154
      6.1.2 Relation to Other Approaches ............................................ 156
      6.1.3 Assumptions and Limitations .............................................. 156
   6.2 Structure, Components and Interfaces ....................................... 157
   6.3 Flexibility and Adaptability ....................................................... 159
      6.3.1 Security Agent ................................................................. 159
      6.3.2 Controlled Data Disclosure Agent .................................... 161
      6.3.3 Hybrid Trust Management Agent ..................................... 161
   6.4 Security Framework Configuration .......................................... 162
7 Conclusions and Future Work

7.1 Summary of Results

7.1.1 Trust Management

7.1.2 Privacy Protection and Controlled Data Disclosure

7.1.3 Security Management

7.2 Future Work

7.3 Concluding Remarks

Bibliography
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Contributions to the field of WSNs security</td>
<td>7</td>
</tr>
<tr>
<td>2.1</td>
<td>Example WSNs applications and scenarios</td>
<td>15</td>
</tr>
<tr>
<td>3.1</td>
<td>Characteristics of certificate-based trust models</td>
<td>36</td>
</tr>
<tr>
<td>3.2</td>
<td>Trust evaluation parameters of behaviour-based trust models</td>
<td>41</td>
</tr>
<tr>
<td>3.3</td>
<td>Comparison of the supported trust characteristics of the trust establishement models for ad hoc and sensor networks</td>
<td>50</td>
</tr>
<tr>
<td>3.4</td>
<td>Energy costs of cryptographic operations on sensor nodes</td>
<td>78</td>
</tr>
<tr>
<td>4.1</td>
<td>Types of trust evidence combined in the proposed model</td>
<td>92</td>
</tr>
<tr>
<td>4.2</td>
<td>Elements of the trust associations of node $i$ with any node $j$</td>
<td>93</td>
</tr>
<tr>
<td>4.3</td>
<td>Trust establishment evidence and validation for trust relationship between any $i$, $j$</td>
<td>96</td>
</tr>
<tr>
<td>4.4</td>
<td>Notations for the configuration parameters of sensor node $i$</td>
<td>97</td>
</tr>
<tr>
<td>4.5</td>
<td>Analysis of the model against the trust management requirements</td>
<td>119</td>
</tr>
<tr>
<td>6.1</td>
<td>Indicative list of mechanisms for the security levels</td>
<td>160</td>
</tr>
<tr>
<td>6.2</td>
<td>Security management framework configuration issues and affecting parameters</td>
<td>162</td>
</tr>
</tbody>
</table>
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Trust properties: Diversity</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>Trust properties: Transitivity</td>
<td>31</td>
</tr>
<tr>
<td>3.3</td>
<td>Basic elements in behaviour-based trust establishment systems</td>
<td>39</td>
</tr>
<tr>
<td>3.4</td>
<td>The phantom flooding strategy for source location privacy in the panda-hunter problem</td>
<td>57</td>
</tr>
<tr>
<td>3.5</td>
<td>Example setting of distributed privacy policies enforcement for location based services</td>
<td>64</td>
</tr>
<tr>
<td>3.6</td>
<td>Sample mix zone arrangement with two application zones and two users</td>
<td>68</td>
</tr>
<tr>
<td>3.7</td>
<td>Privacy preserving data aggregation through blurred node identifiers in a building sensor network</td>
<td>70</td>
</tr>
<tr>
<td>4.1</td>
<td>Working WSN scenario</td>
<td>90</td>
</tr>
<tr>
<td>4.2</td>
<td>Adding pre-deployment knowledge on trust associations</td>
<td>94</td>
</tr>
<tr>
<td>4.3</td>
<td>Working WSN scenario: Nodes configuration</td>
<td>98</td>
</tr>
<tr>
<td>4.4</td>
<td>Classification of the nodes according to their functional and referral trust</td>
<td>98</td>
</tr>
<tr>
<td>4.5</td>
<td>Working WSN scenario: Varying the degradation parameter</td>
<td>99</td>
</tr>
<tr>
<td>4.6</td>
<td>The effect of the total referral trust parameter on the trust associations</td>
<td>101</td>
</tr>
<tr>
<td>4.7</td>
<td>The effect of the propagation set on the required operations</td>
<td>102</td>
</tr>
<tr>
<td>4.8</td>
<td>Basic components of the hybrid trust management system</td>
<td>103</td>
</tr>
<tr>
<td>4.9</td>
<td>Trust management simulation setup: Network view</td>
<td>125</td>
</tr>
<tr>
<td>4.10</td>
<td>Types of trust relationships of the WSN nodes per round</td>
<td>127</td>
</tr>
<tr>
<td>4.11</td>
<td>Trust relationships of node WSN&lt;sub&gt;0&lt;/sub&gt;</td>
<td>128</td>
</tr>
<tr>
<td>4.12</td>
<td>Operations per round by WSN nodes</td>
<td>129</td>
</tr>
<tr>
<td>4.13</td>
<td>Trust re-evaluations per round by WSN nodes</td>
<td>130</td>
</tr>
<tr>
<td>4.14</td>
<td>Trust relationships at the end of the second phase in the clustered scenario</td>
<td>132</td>
</tr>
<tr>
<td>4.15</td>
<td>Operations per round by WSN nodes in the clustered scenario</td>
<td>133</td>
</tr>
<tr>
<td>4.16</td>
<td>Trust re-evaluations per round by WSN nodes in the clustered scenario</td>
<td>134</td>
</tr>
<tr>
<td>5.1</td>
<td>Basic components of the trust–based data disclosure system</td>
<td>142</td>
</tr>
<tr>
<td>5.2</td>
<td>Data disclosure simulation setup: Network view</td>
<td>148</td>
</tr>
<tr>
<td>5.3</td>
<td>Types of trust relationships of VSN nodes per round</td>
<td>149</td>
</tr>
<tr>
<td>5.4</td>
<td>Trust relationships of node VSN&lt;sub&gt;2&lt;/sub&gt; on round 300</td>
<td>149</td>
</tr>
<tr>
<td>5.5</td>
<td>Types of replies of VSN nodes to data requests per round</td>
<td>150</td>
</tr>
<tr>
<td>6.1</td>
<td>The security framework — scaled-down and extended versions</td>
<td>158</td>
</tr>
<tr>
<td>6.2</td>
<td>Node power consumption with and without adaptive security</td>
<td>164</td>
</tr>
<tr>
<td>6.3</td>
<td>Estimated power consumption for different scenarios</td>
<td>165</td>
</tr>
</tbody>
</table>
Περίληψη

Η διατριβή αυτή επικεντρώνεται στην ερευνητική περιοχή της ασφάλειας σε ασύρματα δίκτυα αισθητών, και ειδικότερα στους τομείς της διαχείρισης των σχέσεων εμπιστοσύνης μεταξύ των κόμβων, του ελέγχου προσπέλασης δεδομένων και της διαχείρισης των υπηρεσιών ασφάλειας. Η κύρια συμβολή της διατριβής είναι ένα μοντέλο διαχείρισης εμπιστοσύνης, το οποίο εφαρμόζεται για την ελεγχόμενη προσπέλαση και απόκρυψη των δεδομένων που παράγονται από τους ασθητήρες, και στη συνέχεια ενσωματώνεται σε μια ολοκληρωμένη λύση ασφάλειας.

Το ασύρματο δίκτυο ασθητήρων χαρακτηρίζεται από ιδιότητες που θέτουν σημαντικούς περιορισμούς και απαιτήσεις στο σχεδιασμό εφαρμόσιμων λύσεων ασφάλειας, ιδιωτικότητας και εμπιστοσύνης: Σε επίπεδο κόμβων, οι ασθητήρες διαθέτουν περιορισμένες υπολογιστικές ικανότητες, μνήμη και ενεργειακά αποθέματα. Σε επίπεδο δικτύων, είναι ετερογενείς, δυναμικά, και χωρίς σταθερά κεντρικά σημεία διαχείρισης. Σε επίπεδο δεδομένων, γίνεται επεξεργασία και συνάθροιση των μετρήσεων των ασθητήρων από ενδιάμεσους κόμβους, γεγονός που αυξάνει την πολυπλοκότητα στη ροή πληροφοριών.

Η διαχείριση των σχέσεων εμπιστοσύνης μεταξύ των κόμβων είναι μια σημαντική υπηρεσία ασφαλείας στα δίκτυα αισθητών, λόγω του ότι είναι από τη φύση τους συνεργατικά δίκτυα, ενώ ταυτόχρονα είναι ευάλωτα σε εσωτερικές επιθέσεις από κακόβουλους κόμβους. Στη διατριβή αναλύουμε τις έννοιες και τις ιδιότητες που σχετίζονται με την εμπιστοσύνη και παρουσιάζουμε τα μοντέλα διαχείρισης εμπιστοσύνης που έχουν προταθεί κατηγοριοποιημένα ανάλογα με τον σκοπό τους και τους τύπους αποδεικτικών στοιχείων που χρησιμοποιούν σε αυτά που βασίζονται σε πιστοποιητικά και αυτά που βασίζονται στη συμπεριφορά των κόμβων. Επιπλέον, προσδιορίζουμε τους περιορισμούς και το μειονεκτηματικά της κάθε προσέγγισης όσον αφορά τις ιδιότητες της εμπιστοσύνης που υποστηρίζει, τους υπολογιστικούς πόρους που απαιτεί, καθώς και τη δυνατότητα εφαρμογής της σε δίκτυα ασθητήρων, και συμπεραίνουμε ότι δεν είναι εναλλακτικές προσεγγίσεις, αλλά συμπληρωματικές. Παρουσιάζουμε ξεχωριστά τις υβριδικές λύσεις, οι οποίες συνδυάζουν διαφορετικούς τύπους αποδεικτικών στοιχείων για την αξιολόγηση της εμπιστοσύνης, ώστε να προσφέρουν τις ιδιότητες της εμπιστοσύνης που υποστηρίζουν και τους δύο προσεγγίσεις ταυτόχρονα. Από την ανάλυση προκύπτει όμως ότι τα υπάρχοντα υβριδικά μοντέλα έχουν προταθεί για άλλα δίκτυα και περιβάλλοντα, και δεν λαμβάνουν υπόψη τις ειδικές απαιτήσεις και τους περιορισμούς των δικτύων ασθητήρων.

Ο ερευνητικός στόχος μας στόχος είναι ο σχεδιασμός μιας γενικής και ασφαλούς λύσης για τη δυναμική διαχείριση των σχέσεων εμπιστοσύνης εντός και μεταξύ ετερογενών δικτύων ασθητήρων σύμφωνα με την προϋπάρχουσα γνώση για τους ρόλους και τις σχέσεις των κόμβων, του σκοπό των δικτύων, και τις διαθέσιμες πληροφορίες για κακόβουλες ενέργειες ή συμπεριφορές των κόμβων. Το μοντέλο διαχείρισης εμπιστοσύνης που προτείνουμε
είναι υβριδικό. Χρησιμοποιεί ταυτόχρονα αξιολόγησης της συμπεριφοράς των κόμβων ως αποδεικτικά στοιχεία, και τα συνδυάζει σε κοινές διαδικασίες αξιολόγησης και μετρικές. Στο μοντέλο περιγράφοντα μια διάφορα είδη των αποδεικτικών στοιχείων και των σχέσεων εμπιστοσύνης που υποστηρίζονται, οι παράμετροι προσαρμογής του, οι διαδικασίες διαχείρισης και οι μετρικές αξιολόγησης της εμπιστοσύνης. Η αξιολόγηση του μοντέλου γίνεται μέσω των αποδεικτικών στοιχείων και εμπιστοσύνης που υποστηρίζει, της ανθεκτικότητας του σε διάφορους τύπους επιθέσεων, και των απαιτήσεων του σε ενεργειακούς πόρους. Επιπλέον, γίνονται πειράματα αξιολόγησης και μετρικές προσαρμογής της λειτουργίας του μοντέλου σε διαφορετικές συνθήκες και παραμετροποιήσεις.

Στην περιοχή της διαχείρισης της ιδιωτικότητας, προσδιορίζουμε τις απαιτήσεις για το σχεδιασμό δικτύων αισθητήρων, όσο και σε επίπεδο εφαρμογών. Από την ανάλυση προκύπτει ότι η προστασία της ιδιωτικότητας εντοπίζεται σε διάφορα σημεία της στοιβάδας πρωτοκόλλων. Παρουσιάζουμε τις λύσεις που έχουν προταθεί κατηγοριοποιμένον στους μηχανισμούς προστασίας του πλαίσιος των επικοινωνιών για την απόκρυψη της θέσης ή της ταυτότητας των κόμβων, οι μηχανισμοί για εγκληματία επικοινωνίας και για προσαρμογή του επιπέδου λεπτομερείας των δεδομένων, και στις λύσεις για ελεγχόμενη συλλογή δεδομένων. Η κατηγοριοποίηση αυτή αναδεικνύει τις διαφορετικές στιγμές της ιδιωτικότητας, και τις διαφορετικές λύσεις που προτείνονται στους μηχανισμούς για εκφυλίση ή ελέγχομενη συλλογή δεδομένων. Η κατηγοριοποίηση αυτή αναδεικνύει τις διαφορετικές της ιδιωτικότητας, και τις διαφορετικές λύσεις ως λύσεις απαλλαγής από την απόκρυψη και την ελεγχόμενη συλλογή δεδομένων.

Ως συνέπεια, επισημαίνεται ότι η προστασία της ιδιωτικότητας επικοινωνίας απαιτεί εκπληρώματα διαδικασιών και παρακολούθησης των δεδομένων, και στις λύσεις για ελεγχόμενη συλλογή δεδομένων. Η κατηγοριοποίηση αυτή αναδεικνύει τις διαφορετικές στιγμές της ιδιωτικότητας, και τις διαφορετικές λύσεις που προτείνονται στους μηχανισμούς για ελεγχόμενη συλλογή δεδομένων.

Στην περιοχή της διαχείρισης της ιδιωτικότητας, προσδιορίζουμε τις απαιτήσεις για το σχεδιασμό δικτύων αισθητήρων, όσο και σε επίπεδο εφαρμογών. Από την ανάλυση προκύπτει ότι η προστασία της ιδιωτικότητας εντοπίζεται σε διάφορα σημεία της στοιβάδας πρωτοκόλλων. Παρουσιάζουμε τις λύσεις που έχουν προταθεί κατηγοριοποιμένον στους μηχανισμούς προστασίας του πλαίσιος των επικοινωνιών για την απόκρυψη της θέσης ή της ταυτότητας των κόμβων, οι μηχανισμοί για εγκληματία επικοινωνίας και για προσαρμογή του επιπέδου λεπτομερείας των δεδομένων, και στις λύσεις για ελεγχόμενη συλλογή δεδομένων. Η κατηγοριοποίηση αυτή αναδεικνύει τις διαφορετικές στιγμές της ιδιωτικότητας, και τις διαφορετικές λύσεις που προτείνονται στους μηχανισμούς για ελεγχόμενη συλλογή δεδομένων.
τις λύσεις αυτές που είναι σχεδιασμένες για δίκτυα αισθητήρων δεν παρέχει ευελιξία και 
προσαρμοστικότητα στην διαχείριση υπηρεσιών ασφάλειας, ιδιωτικότητας και εμπιστοσύνης 
tαυτόχρονα.

Ο ερευνητικός μας στόχος στο πεδίο της διαχείρισης υπηρεσιών ασφάλειας είναι ο σχε-
dιασμός μιας ολοκληρωμένης λύσης που να εφαρμόζεται τοπικά στους αισθητήρες ετερο-
γενών και διασυνδεδεμένων δικτύων και ενσωματώνει το μοντέλο διαχείρισης εμπιστοσύνης
και το μηχανισμό ελέγχου προσπέλασης δεδομένων μαζί με βασικές υπηρεσίες ασφάλειας.

Καθορίζουμε τη δομή, τις εσωτερικές υπηρεσίες και τις διεπαφές της λύσης, τις αλληλεοξ-
αρτήσεις μεταξύ τους, τους μηχανισμούς επίτευξης ευελιξίας και προσαρμοστικότητας τόσο
στην ολοκληρωμένη λύση όσο και σε κάθε μία από τις υπηρεσίες που περιλαμβάνει, καθώς
και διαφορετικές εκδόσεις της με τους αντίστοιχους μηχανισμούς και υπηρεσίες ασφάλειας.

Από την ανάλυση και αξιολόγηση της προτεινόμενης λύσης προκύπτει ότι μπορεί να υποσ-
tηρεί την ετερογένεια στις δυνατότητες και στις απαιτήσεις ασφάλειας των κόμβων και
των δικτύων και είναι προσαρμόσιμη σε δίκτυα αισθητήρων διαφόρων πεδίων εφαρμογής.
Abstract

This thesis contributes to the fields of trust, data disclosure control and security management in Wireless Sensor Networks (WSNs). It describes a novel trust management model, applies it for controlling data disclosure operations on the sensor nodes, and integrates it in a security management framework.

WSNs possess a number of unique characteristics, which make it challenging to devise security, trust and privacy solutions that can be applied uniformly in the deployments: At the node level, the sensor nodes are constrained in computation capabilities and energy supplies. At the network level, they are heterogeneous, dynamic, and lack centralised management points. At the data level, in-network aggregation of the sensor readings is applied.

Trust management is an important security service for WSNs, because of their cooperative nature and their vulnerability to node compromise and misbehaviour. In our work, we analyse the concepts and properties of trust, discuss the related work on trust management, and classify the solutions into certificate-based and behaviour-based, according to their scope, purpose and admissible types of evidence. We present the limitations and drawbacks of each approach in terms of the properties of trust that it supports, of its resource requirements, and of its applicability on WSNs, and show that they are not alternative but supplementary. We also discuss hybrid solutions, which combine different types of trust evidence to benefit from the properties of trust that each individual approach offers, and find that the hybrid models in the related bibliography are all proposed for other environments, and do not take into account the special requirements and limitations of WSNs.

The objective of our work on trust management is to design a generic solution for dynamically managing the trust relationships within and between heterogeneous WSN deployments based on the pre-deployment knowledge, the network purpose, and the available feedback on malicious behaviours. The trust management model we introduce is hybrid, utilising all role-base trust associations, certificates and behaviour evaluation results as trust evidence, and combining them on common evaluation processes and metrics. The novel aspects of the model include the different types of trust evidence and trust associations that it supports, its configuration parameters, the trust management processes and the trust evaluation metrics. The model is evaluated by analysing its security and trust properties, its robustness against various types of attacks, and its resource requirements and overheads. Moreover, simulation experiments with varying parameters demonstrate the effectiveness of the model in managing the trust relationships between nodes and clusters, while distributing the computational cost of trust evaluation operations.

In the field of privacy protection, we analyse the requirements and the related work...
and find that privacy issues in WSNs can be addressed at multiple levels of the network stack and at different points of the information flow. We categorise the solutions into those for protecting the communication context, those for privacy sensitive information disclosure, and those for privacy sensitive information gathering. Most solutions for data disclosure and data granularity control are found to have limited applicability, being either too application-specific or requiring the existence of trusted intermediaries and assuming that all devices up to the level where the privacy operations are performed are trusted.

The objective of our work on data disclosure control is to devise a solution for performing data access and data granularity control at the points of data capture in distributed and dynamic WSN deployments according to the deployment needs, the network purpose, and the context of the data requests. We position trust as the facilitator of the data disclosure decisions of the sensor nodes, utilising the hybrid trust management model we developed, to benefit from the flexibility that it offers in handling unknown or compromised data requestors. The solution provides alternative options for data cloaking, including negative surveys and data granularity control mechanisms, for disclosure to partially trusted requestors. Simulations demonstrate that the hybrid trust management model provides adequate information for identifying how data requests should be handled at each sensor node, while respecting the deployment needs and the network configuration.

In the field of security management, we analyse the requirements of WSNs and the related work and we find that, while there exist solutions to address most of the core security requirements of WSNs, few works exist on the integration and management of the security services, none of which is targeted to WSNs and offering flexibility, adaptability and context-awareness properties and including all security, privacy and trust services.

The objective of our work on security management is to integrate the trust management and the data disclosure control mechanisms, along with core security functionality, in a localised security management framework for heterogeneous and integrated WSN deployments. The specification of the framework includes its structure and components, the dependencies between them, the mechanisms for providing adaptability and flexibility in the framework and in each component, and the framework versions with their corresponding sets of security mechanisms. We provide analysis and evaluation data that demonstrate that the framework enables the customisation of WSNs to a diverse set of application spaces and supports the heterogeneity in the capabilities and the security requirements of the nodes and the deployments.
Acknowledgements

This thesis would not have been possible without the motivation, support and guidance I received throughout the four years of my studies from my supervisor Stefanos Gritzalis. I would like to thank him for giving me the opportunity to work in research and pursue my PhD, for suggesting the topic of the thesis, which proved to be both very interesting and timely, for trusting me to be involved in a number of challenging tasks, and for putting just the right amount of pressure and, at the same time, allowing for just the right amount of independence in my research work.

I am also grateful to all the other members of the Laboratory of Information & Communication Systems Security and the Department of Information and Communications Systems Engineering of the University of the Aegean who pleasantly involved themselves in assisting me, cooperating with me, and providing me valuable feedback on my work during the last years. I would especially like to thank Charalabos Skianis for supporting me when entering the funded research world and for all the advice he provided me during the critical initial stages of my work.

During the course of the e-SENSE project, I had the opportunity to work with some brilliant researchers in the field of wireless sensor networks. I want to thank the members of the consortium who helped me broaden my research horizons and place my work into perspective. Through numerous discussions, which sometimes were like negotiations for splitting the resources of the sensor nodes among the services each member was involved with, they made me realise the need for a realistic security solution that can actually be integrated to the deployments. I am particularly grateful to Neeli R. Prasad, Anelia Mitseva and Adrian Waller for our successful cooperation during the project, for all the fruitful discussions that we had, and for willingly sharing their expertise and experience on the field with me.

Moral support proved to be a great deal more important that I used to think when I started working on my PhD. I am immensely fortunate in having parents, siblings and friends who are caring, encouraging and patient. I missed major events in their lives in the course of the last few years, and I hope to compensate in the days to come.

My final word of thanks goes to Georgios Gousios, for much more than I can express. Thank you for the inspiration and motivation to re-enter the academic world, for making home the most productive and pleasant research environment, for all the technical assistance, for sharing your expertise in research tools and methods, for helping me sustain my balance and strength during these years and, most of all, for being my companion.
The work presented in this thesis was partially supported by the European 6th Framework Programme under the Integrated Project e-SENSE ‘Capturing Ambient Intelligence for Mobile Communications through Wireless Sensor Networks’ (www.ist-e-SENSE.org), contract number: 027227.
Chapter 1

Introduction

In the ubiquitous computing era, continuous connectivity and context awareness are key requirements for the provision of smart services. Wireless Sensor Networks (WSNs) are expected to hold an essential role by providing timely and accurate context data to ambient intelligence applications. A wide range of commercial, scientific and military applications can benefit from the increased data collection capabilities that large scale WSN deployments offer.

WSNs are composed of numerous sensor nodes, spatially distributed over sensing fields, that cooperatively capture diverse types of contextual information related to their environment and make it available to applications and services in other networks and application platforms. WSNs became an active research topic during the last years due to the advances in Micro-Electro-Mechanical Systems (MEMS) technology, that enabled the development of smart, low-cost and small sensor devices with wireless communication capabilities. Sensor devices usually consist of an energy source, a processor with limited computation capabilities, communication components, and sensing components for properties like temperature, sound, pressure, or motion.

There is a wide range of application spaces for WSNs. Although both the research community and the industry initially focused on their use for military applications, interest was soon shifted to community and commercial services. These include health care applications for patient monitoring, traffic control, habitat monitoring and enhanced collaboration. In the industrial application space, production monitoring, quality control and smart farms are some of the scenarios where WSNs are used.

However, the deployment of pervasive sensing environments that contain numerous, almost invisible, sensors that constantly monitor their surroundings and communicate a variety of information, inevitably causes concerns related to their potential of abuse and the risks they impose to the privacy of individuals. The potential risks are aggravated by the fact that different types of WSNs may be deployed for different purposes, others being trusted, for example subscription based sensor networks offering health services, others being partially trusted, such as these deployed for customer assistance in shopping malls, and others untrusted, such as surveillance networks the users might be completely unaware of. Even sensor networks initially deployed for legitimate purposes may be violated or abused. Historically, it is believed that as surveillance technology has become cheaper and more effective, it has been increasingly used for privacy abuses [PSW04].
CHAPTER 1. INTRODUCTION

As for any ubiquitous computing technology, the security of the data and the communications are essential requirements for the end applications and services to be reliable, while the protection of the privacy of the end users is essential for their adoption. The design of secure WSNs is an active research area. Within the topics that it includes are the protection of the confidentiality and the integrity of the data communicated, node authentication, access control, and the provision of lower level security services like secure routing and node grouping.

Within the research field of security and privacy in WSNs, that is briefly examined in the next section, this thesis places trust as the cornerstone for the provision of security services—the motivation for this decision is analysed in Section 1.2. We introduce a trust management model for handling the in-network trust relationships of the sensor nodes, and apply it for controlling their data disclosure operations. The trust management model and the data disclosure control mechanism are integrated, along with core security services, in a security management framework. The contributions of this thesis in the research areas of trust management, privacy protection and security management are summarised in Section 1.3. The chapter concludes with clarifications on the work that has been done in collaboration with other researchers.

1.1 Research Field

The field of security and privacy in WSNs is relatively new—very few works were published before 2000. At the same time, the research field is wide, with no standard security issue being left unexplored. During the last few years, security issues of WSNs attracted great interest in the security research community. International security conferences quickly included WSN-specific topics, a number of specialised workshops are organised each year, and most well established security research teams have included WSNs in their topics of interest.

Security in WSNs is a very active research area and a challenging research topic. The main reason is that the characteristics of WSNs render most traditional security solutions inapplicable. The sensor nodes are constrained in terms of processing, memory, communication and, most importantly, energy resources. They are distributed and dynamic networks, and connectivity with base stations can not be taken for granted. Security management solutions for WSNs should thus be distributed, and utilize only a small subset of the available security primitives—the ones with acceptable resource requirements. Another issue that must be taken into consideration during the design of realistic security mechanisms is that security services do not provide core network functionality and, being supportive services, they should impose reasonable computational overhead and resource consumption. Moreover, security solutions for WSNs have to cater for in-network data aggregation operations, data with varying sensitivity being communicated, multi-hop routing, intermediaries with varying levels of trust, and integration with different networks and platforms.

For secure WSNs, everything had to be reinvented. Core security aspects were the first to be explored by the research community. A wealth of lightweight cooperative key management solutions have been proposed [CGPM05, CY05, XRS+07]. Cryptographic primitives have been evaluated regarding their applicability on sensors [GSSK05, RAL07]. Application-dependent security management schemes have been
developed for a variety of WSN application fields. Secure routing and secure data aggregation solutions have been examined [WAR06, CMYP09].

The resource constraints of the sensor nodes also stimulated interest towards the design of novel privacy protection solutions. Most existing privacy solutions, like those used in internet communications, are too complex computationally and too expensive in terms of energy requirements for the resource-constrained sensors. Moreover, since they mostly focus on protecting the contextual aspects of the communications, they can not fulfill the complete spectrum of the sensor networks’ privacy requirements. Privacy issues of WSNs can be addressed at multiple levels of the network stack and at different points of the information flow. Privacy protection mechanisms have been proposed for the protection of the communications context [PMR+08], for privacy sensitive information gathering and aggregation [GSJ+03, HGPE07], and for privacy sensitive information disclosure [Sne01, MWP08].

A significant amount of research has also been carried out on trust and reputation management [GS04, BXEK07, PK07, CP07]. The cooperative nature of WSNs, along with the susceptibility of the sensor nodes to a variety of misbehaviours, make it valuable to assess the in-network trust relationships. The relationships established between network nodes can be used for the provision of higher level security solutions, such as secure routing [GBS08], trusted key exchange [LF07], and data disclosure decisions [AG09b]. In most trust management models that have been proposed for WSNs, trust is formulated as a combination of the direct trust value to the target node, which is evaluated by the trust issuer based on previous interactions and network traffic monitoring metrics, and the indirect trust value derived from the recommendations of neighboring nodes. Trust evaluation is thus behaviour-based, since the direct estimates are based on the observed behaviour of the nodes.

Trust management is the research field that is the mostly relevant to this thesis. Overall, the trust management field includes topics like trust representation, establishment, maintenance, and distribution. Regarding privacy protection, this thesis relates to the research topic of controlled information disclosure. Security management is also a topic of interest, since an application-independent security management framework is developed to incorporate the proposed trust management and controlled information disclosure solutions.

1.2 Motivation and Research Approach

The overall objective of our work is to develop a security and privacy solution for WSNs. During the initial stages of our research, we focused on studying background material and related work. The problem area was decomposed into elements, and security and privacy requirements were derived. In order to obtain a practical view of the problem, we searched the literature and selected a broad range of scenarios where WSNs can be applied. The existing security and privacy approaches for WSNs were then analysed, classified, and evaluated both on the security properties that they offer, and on their applicability to the WSNs of the selected scenarios.

While studying the research field, we broadened the explored area and came across solutions on trust establishment for ad hoc networks. It was then that we recognised the potential for a trust-based security solution for WSNs. The main motivation for
using trust is that, like ad hoc networks, WSNs depend highly on the distributed cooperation among network nodes. The network membership changes and the dynamic topology require the cooperation of nodes that may have been unknown during the pre-deployment phase. At the same time, the sensor nodes are susceptible to a variety of misbehaviours. From compromised nodes acting as internal attackers to legitimate nodes that act selfishly, internal misbehaving nodes are a vulnerability that cannot be tackled using authentication and access control mechanisms alone.

Most trust establishment schemes that have been proposed for WSNs, inspired by the ones that had been proposed for ad hoc networks, perform behaviour-based trust evaluation. Sensor networks, however, are substantially different from the traditional case of ad hoc networks. Our analysis and scenario-based evaluation of these approaches revealed that behaviour-based trust evaluation cannot be directly applied to WSNs. The first reason is that behaviour-based trust evaluation requires the continuous observation of the behaviour of neighbouring nodes, which would drain the energy supplies of the sensors—in most practical WSN deployments, the sensors’ radio is not even constantly on, in order to prolong battery lifetime. The second reason is that the semantics assigned to trust by behaviour-based trust models are related to node behaviour; therefore, the metrics produced may have high value for detecting compromised or selfish nodes, but low value for other purposes, for example controlled data disclosure.

This problem arises because, unlike ad hoc networks, WSNs are purpose-specific: their ownership and purpose are known before deployment. We found that the models that are based solely on behaviour monitoring and evaluation are not entirely suitable for data disclosure decisions. They do not exploit pre-deployment knowledge, that is the most important factor for data access decisions, which should be taken based on the deployment needs. Moreover, we believe that good node behaviour should not compensate for the distrust that the deployment needs may dictate.

The motivation for the trust management model that we developed was to combine the results of behaviour monitoring with types of trust evidence that represent the deployment needs and purpose. One of the main objectives that was set for the model was to be generic, independent of the semantics assigned to trust. It should be targeted to purpose-specific networks, and to enable the network designer to control trust evolution according to the pre-deployment knowledge. Moreover, since scenario analysis had shown that diversity exists between the nodes of real-world integrated WSNs, the model should uniformly support the needs of nodes with highly diverse network roles and capabilities, and distribute the cost of trust evaluation operations. Other objectives for the model were to support distributed trust evaluation and to allow controlled trust revocation.

In addition to trust management, we studied the area of security for WSNs and worked towards a security management solution that would integrate the trust management model. Very few works focusing the integration and management of security services were found in the related literature. Because of the diversity that was identified between the nodes of WSNs and between the security requirements of the data collected and communicated, we set as our objective to construct a security management framework that would integrate lightweight security services, including trust establishment, in a flexible and adaptable way.

Having developed both the trust management model and the security framework that would include it, we continued our research on the field of privacy protection.
and, specifically, on the topic of controlled data disclosure. Data access control is an essential requirement for WSNs, since they can be deployed in settings with multiple data collectors, with varying levels of trust associated to them, that offer a variety of services in exchange of raw or aggregated information of varying levels of detail. Most existing solutions use privacy policies and privacy tags for each type of data, and introduce intermediaries between the data provider and the data requestor to perform data disclosure control operations. However, they assume that the entities and devices are trusted up to the level where the privacy operations are performed.

Our idea was to exclude intermediaries and to use trust as the facilitator for data disclosure decisions that would be made locally at each sensor node. Since controlled data disclosure includes data access and data granularity control, we examined different options for reducing data accuracy. The objective was to utilise the metrics produced by the trust management model, while sustaining the semantic consistency of the controlled information disclosure scheme.

In summary, the objectives that we set for our work in the fields of trust, privacy and security in WSNs are to provide:

1. A trust management model for WSNs that is generic, enables controlling trust evolution according to the pre-deployment knowledge and the network purpose, enables distributing the cost of trust management operations, and utilises the results of behaviour-based trust evaluation.

2. A controlled data disclosure solution that utilises the trust assigned to data requestors for performing localised data access and data granularity control.

3. A flexible and adaptable security management framework that facilitates the provision of core security functionality and integrates the trust management model and the controlled data disclosure mechanisms.

For each of these objectives, we formulate a set of detailed requirements for our solution by examining the characteristics of WSNs, their trust, privacy, and security requirements, the properties and characteristics of each associated security concept, the approaches that exist for addressing trust and privacy requirements in other types of systems and networks, and the related work on trust, privacy, and security management in WSNs. We then specify our solution, justify our main design decisions, and analyse and evaluate it both in terms of how it addresses the requirements, and through experimental evaluation results. We developed a simulation environment to facilitate testing of both the trust management model and of the controlled information disclosure solution for different WSN scenarios. The simulation results have proved valuable for assessing the effectiveness of the trust metrics in controlling trust evolution, and for introducing refinements to the trust management processes.

1.3 Contribution

The core contribution of this thesis to the research field of secure WSNs is a generic trust management model, that is applied for controlling data disclosure operations on the sensor nodes, and is integrated in a security management solution. Except from the core research contribution, the reported work includes intermediate results on the
CHAPTER 1. INTRODUCTION

definition of security requirements and the review, analysis and evaluation of related work on the field. The contributions of the thesis are summarised in Table 1.1, while the following sections briefly describe the contribution and the results obtained in each of the related research areas.

1.3.1 Trust Management

Our contribution to trust management is a generic trust management model. The model is hybrid, utilising all role-base trust associations, certificates and behaviour evaluation results as trust evidence, and combining them on common evaluation processes and metrics. It exploits the pre-deployment knowledge on the network topology and the information flows, and allows controlled trust evolution based on the network pre-configuration. Moreover, it enables controlled trust revocation through the propagation of behaviour evaluation results made available by supervision networks.

Our initial research results on the solution have been reported in reference [AGS07], while the complete model was later described in reference [AG09a]. The specification of the model includes its structure and interfaces, the different types of trust evidence and trust associations that it supports, the trust establishment and re-evaluation processes, the functional and referral trust evaluation metrics, the trust configuration parameters, and the controlled trust revocation mechanisms that determine how the results of behaviour based evaluation are propagated. Within the novel features of the model is the representation of the distrust that nodes should exhibit through an explicit trust degradation parameter. Moreover, to the best of our knowledge, it is the first model to explicitly use valid signed certificates as quantified recommendations.

We have studied the security properties of the model, its robustness against various types of attacks, and its resource requirements and overheads. Through simulation experiments with varying parameters, we have also validated the proposed model and the trust metrics and processes it includes. The results have demonstrated its effectiveness in managing the trust relationships between nodes and clusters, while distributing the computational cost of trust evaluation operations.

Before devising the trust management model, we had analysed, evaluated, and reported on the related work on trust establishment in ad hoc and sensor networks. In references [AGS06b] and [AGS08] we classify the trust establishment solutions into behaviour-based and certificate-based approaches, according to their scope, purpose and admissible types of evidence, and analyse them in terms of their supported trust characteristics, their complexity and requirements, and their deployment complexity and flexibility. The results of the analysis demonstrated that the two approaches adopt different perspectives of trust and that each has its own limitations and drawbacks, and provided our motivation to form a hybrid approach to leverage their disadvantages.

1.3.2 Privacy Protection

Our contribution to privacy protection is an application level scheme for controlling information disclosure at the points of data capture or generation. The scheme utilises the trust management model for facilitating in-network privacy decisions. The main characteristic of the model that makes it suitable for privacy decisions is that it enables the exploitation of pre-deployment knowledge in order to control trust evolution
Table 1.1: Contributions to the field of WSNs security

<table>
<thead>
<tr>
<th>Research Area</th>
<th>Publications</th>
<th>Contribution</th>
</tr>
</thead>
</table>
| Trust Management   | [AG09a, AGS08, AGS07, AGS06b] | Classification of trust establishment solutions for wireless sensor and ad hoc networks into behaviour-based, certificate-based and hybrid, analysis and evaluation on trust characteristics, complexity, and suitability for WSNs.  
Definition of generic hybrid trust management model, analysis of security properties and experimental evaluation.  
Specification of trust metrics and trust establishment processes.  
Definition of parameters for controlling trust evolution and distribution of trust evaluation overheads. |
| Privacy Protection | [AG09b, AGK09, AGK08, AGS06a] | Definition of set of privacy requirements for privacy protection solutions for WSNs, classification of solutions according to their position in the network stack and their interference with the information flow.  
Definition of trust-based data disclosure scheme, scenario analysis and experimental evaluation.  
Integration of trust metrics in privacy protection mechanisms.  
Specification of node classification processes and of alternative data cloaking operations. |
| Security Management| [MAM+08, AMS+07b, AMS+07a] | Definition of adaptive security management framework, its analysis, evaluation, and application on medical scenarios.  
Integration of trust establishment and privacy protection components in the security architecture. |
according to the deployment needs and the network purpose. The trust assigned to each data requestor is used to determine whether the data or only a sample of it will be disclosed, or if the request will be rejected. The scheme allows the use of various mechanisms, including negative surveys and data granularity control mechanisms, for publishing samples of data to partially trusted requestors.

The results on the trust-based controlled information disclosure scheme were presented in reference [AG09b]. The specification of the scheme includes its structure and interfaces, the definition of distinct trust classes for the nodes requesting data, and the specification of how the trust status dictates the data access and granularity control operations to be performed. The scheme has been validated through simulation experiments on scenarios with both trusted and partially trusted nodes.

Before devising the controlled information disclosure scheme, we had specified the requirements for privacy preserving WSN deployments and analysed the related work. In references [AGS06a] and [AGK08] we categorise the privacy requirements and the mechanisms that address them into privacy sensitive information gathering schemes, controlled information disclosure approaches, and mechanisms for the protection of the communications context. Through the separate discussion of the approaches we highlight the diverse privacy aspects that have been studied in the literature, and show how the approaches can be viewed as complementary to fulfil the complete spectrum of sensor networks’ privacy needs.

1.3.3 Security Management

Our contribution to security management is a generic security management framework for heterogeneous WSNs that integrates that trust management model and the controlled information disclosure mechanisms. To allow for customisation of the security framework to a diverse set of WSN scenarios, it is designed to be adaptable and flexible. These properties were attained through following a modular approach for the integration of various configurable protocol and control elements.

The security framework has been described in reference [MAM+08], while reference [AMS+07b] applies the framework on medical scenarios by specifying its application-dependent components. The specification of the framework includes its generic architecture, the mechanisms for providing adaptability and flexibility, the components and their interfaces, and the framework versions with their corresponding sets of suitable security protocols and mechanisms that are combined to form a complete toolbox solution. Performance evaluation results have demonstrated the feasibility of the solution and have approximated the benefits of the security framework, mainly in terms of resource consumption, for a variety of scenarios.

1.4 Work Done in Collaboration

The security management framework described in Chapter 6 and published in references [AMS+07b], [MAM+08], and [AMS+07a] has been developed in conjunction with Aelia Mitseva, Neeli R. Prasad, Maria A. Marchitti (all from Center For TeleInfrastruktur, Aalborg University, Denmark), Charalabos Skianis (University of the Aegean, Greece), and Adrian Waller (Thales Research and Technology Ltd, UK). The adaptability property of the framework was inspired by previous research of Neeli R. Prasad.
prototype implementation of the security framework, the results of which were later used for its evaluation, was performed by Adrian Waller, Timothy Baugé, and Sarah Pennington (all from Thales Research and Technology Ltd, UK).

1.5 Organisation of the Thesis

Chapter 2 includes background information on WSNs, on their characteristics and applications, and defines their trust, privacy, and security requirements. The solutions that have been proposed to address the requirements are then discussed in Chapter 3.

The trust management model, the motivation for our design decisions, the algorithms and metrics that it includes, and its analysis and evaluation are presented in Chapter 4. Chapter 5 applies the model for controlled data disclosure, specifies the processes and mechanisms for data access and granularity control, and analyses and evaluates the scheme. Chapter 6 describes the adaptive security management framework that integrates the trust management and data disclosure solutions.

Finally, Chapter 7 concludes the thesis and provides some directions for future work. Each chapter of the thesis, except the current and the final ones, is finalised with a separate summary section that briefly describes the core issues that the chapter included.
Chapter 2

Wireless Sensor Networks & Security

WSNs are composed of numerous sensor nodes, spatially distributed over sensing fields, that capture, process and communicate various types of contextual information related to their environment. They possess a number of unique characteristics that affect the design of security solutions and are discussed in the next section. They are used in a wide variety of scenarios; examples are provided in Section 2.2. In Section 2.3 we relate WSNs to security by explaining the concepts and notions, analysing their security, trust and privacy requirements, and discussing the challenges in fulfilling them.

2.1 Characteristics of WSNs

The unique characteristics of WSNs have inspired research in a number of fields, including security. Their characteristics at node, network and data level are separately presented:

Sensor node characteristics A sensor node consists of an energy source, a processing unit, communication components and sensing units, while optional components include localisation devices and secondary power supplies like solar panels. Sensing units may include a variety of mechanical, thermal, biological, chemical, optical, and magnetic sensors that measure physical properties of the environment such as temperature, sound, pressure, light, or motion. They are designed to be small and inexpensive, and are thus constrained regarding their energy, memory, computation and communication capabilities. Current commercially available sensor devices, like Micaz, have 7.3 MHz microcontrollers (ATmega128L) with 128KB of program memory and 4KB Random-Access Memory (RAM), while their data rate is 250 kbps. TelosB nodes have 16-bit, 8 MHz microcontrollers (Texas Instruments MSP430), 48K program memory, 10KB RAM, and 1024K flash storage.

The most important constraint of sensor nodes is energy, since it affects the lifetime of the network. The common energy source for sensor nodes are AA batteries. In many WSN applications, regular maintenance of the sensor nodes in not possible, so

\[1^{http://www.xbow.com/} \]
the batteries are not replaced or recharged after deployment. To prolong the lifetime of each node and of the entire network, energy consumption must be restrained. One of the most straightforward methods to achieve this involves dynamic power management. Sensor nodes enter sleep state when not needed and are woken up when necessary. When in active state, the power consumption can be minimised by adapting the power supply and operating frequency to match the workload \cite{SC01}. Other techniques that have been proposed include power aware computation and communication components, low-energy signalling and networking, and power aware software infrastructure \cite{MBC01}.

Regarding sensor node operating systems, TinyOS\footnote{http://www.tinyos.net/} is the one used by most commercially available sensor devices and is considered to be the industry standard \cite{LMP05}. Initially developed by U.C. Berkeley, it is a small (the base OS is about 400 bytes), open-source operating system that handles task scheduling, I/O processing, radio communication etc. It provides a programming framework and set of reusable components which can be assembled for the development of application-specific systems. Its component-based programming model is codified by the NesC language, a dialect of C \cite{GLvB03}. Apart from TinyOS, other lightweight operating systems for sensors include SOS \cite{HKS05} and Contiki \cite{DG04}.

Network organisation and communications The heterogeneity of the nodes is one of the main characteristics of WSNs. Heterogeneity can be identified in the roles of the nodes, which range from simple sensors to dedicated data aggregators, cluster heads and gateways to other networks, in their computational capabilities, in the type of information that they collect, in their mobility and the possibility of their regular maintenance. The number of the nodes may vary, depending on the application and the sensing field, from a few nodes forming a body sensor network, to thousands of nodes monitoring an environmental phenomenon on a large terrain. The position of the nodes may be predetermined, in which case they are placed one by one in the sensing field, or their topology can be random, with the nodes thrown in the sensing field as a mass \cite{ASSC02b}.

The nodes may be organised in completely distributed flat structures or in, possibly hierarchical, clusters. In the latter case, the cluster heads may be responsible both for making decisions regarding the network operation, like message routing and data aggregation, and for enforcing them. After deployment, the topology of the network may change due to the mobility of the nodes or the entities they are adjusted on, to the addition of new nodes, or to the unreachability of existing ones. There exist from static WSNs, where the nodes are manually placed in the sensing field, are attached to static objects, are organised in static clusters, and receive regular maintenance, to highly dynamic WSNs, where the nodes are deployed in an ad hoc manner, operate unattended, and are self-organised.

Regarding the communications between nodes, messages usually follow multihop routes through a wireless medium, which can be radio or optical. The most common type of transceiver that sensor nodes are equipped with is IEEE 802.15.4/ZigBee compliant direct sequence spread spectrum radio transceiver, operating at 2.4 GHz frequencies, although there exist sensor nodes, like the BTNode\footnote{http://www.btnode.ethz.ch/} equipped with Bluetooth
2.1. CHARACTERISTICS OF WSNS

frequency hopping spread spectrum radios. IEEE 802.15.4 [IEE06] is the standard that is the most widely used for sensor nodes’ communications, mainly due to its low energy requirements in comparison to Bluetooth. The standard was developed for low rate wireless personal area networks, and specifies the physical and media access control layers.

Other specifications, like ZigBee [All05], build on IEEE 802.15.4 to cover the higher-level layers of the network stack while being lightweight, enabling multihop message transmission, and supporting devices with different network roles. ZigBee was developed as a mesh networking standard whose protocol stack includes a suite of communication protocols for the network and application layers, intending to provide a framework for distributed application development. Apart from ZigBee, several other standards have been developed for WSNs, and a great amount of research has been carried out on optimising the message routing strategy, and thus reducing the communication overheads. A number of flat and hierarchical routing protocols have been proposed, including geographical routing, low-energy adaptive clustering hierarchy, and directed diffusion. We refer the reader to reference [AKK04] for more information on routing strategies, to reference [BPC+07] for IEEE 802.15.4 and ZigBee, and to reference [YMG08] for other communications standards that have been developed.

Data collection and processing The role of the sensor nodes is to monitor their environment and make the sensed data available. When and what data is made available depends on the application. The purpose of the network may be to monitor a sensing field, to alert on the occurrence of certain phenomena, or to respond to requests for information. The sensor nodes may thus make the accumulated data available periodically, as alerts when they identify predefined circumstances, or on demand. The data that is made available can be either raw sensed data or processed data, depending on the information needs of the application.

Sensed data is processed not only at the node, but also at the network level. A simple information flow, where all sensors send data to a base station, would not be suitable for WSNs. Firstly, it is rare for an application to require all the raw sensed data. Secondly, this would be too resource consuming for the intermediate nodes in the multihop message routes. In order to minimise communication overheads, in-network data aggregation is performed. Large streams of data are converted to aggregated information within the sensor network either by the sensors or by other dedicated aggregator nodes. The optimisation of the information flow and of the data aggregation operations is important to preserve resources and prolong the lifetime of the network. A number of data aggregation protocols have been proposed, both for flat and for clustered sensor networks [RV06]. However, the data aggregation operations that can be performed depend on the type of the sensed data and the information needs of the application, so practically it is difficult to define generic data aggregation techniques. Moreover, since with data aggregation the accuracy of the data may degrade, aggregation entails a tradeoff between communication costs and data quality.

Having studied the characteristics of WSNs, it becomes apparent that they feature ad hoc networking properties, like self-organisation and distributed operation. They are not, however, pure ad hoc networks. Unlike ad hoc network nodes, the sensor nodes
have limited computational and energy resources, and their number can be significantly higher than the nodes of an ad hoc network. The sensor nodes may not have global identifiers, similar to the IP addresses that ad hoc network nodes can be identified by. WSNs may not be entirely distributed, but organised into hierarchies or clusters of nodes. More importantly, unlike ad hoc networks, WSNs have an ownership and are deployed for pre-determined, specific purposes. The sensor nodes may be deployed in an ad hoc manner, but each has a role, exists to fulfil specific information requirements, and to collaborate with the other nodes in order to serve the information needs of a specific application.

### 2.2 Applications and Scenarios

We identify two broad classes of WSNs with respect to the monitored subjects: those that are related or attached to entities, and those deployed for monitoring sensing fields. The entities related to the first class include people, animals, plants, objects, equipment and vehicles. Examples are the body sensor networks that are deployed for monitoring the vital functions of individuals, like patients in hospitals or athletes during their training. The sensing fields of the second class include locations and buildings. Examples are environmental sensor networks deployed for monitoring industrial facilities, forests, rivers or volcanoes. WSNs may include clusters of nodes of both types. An example WSN in the medical field may include both environmental sensor nodes for monitoring the temperature and humidity in the patient recovery areas, and body sensor clusters attached to patients. Another classification that is made based on the sensing environment of WSNs includes terrestrial, underground, underwater, multi-media, and mobile WSNs [YMG08].

In reference [YMG08], the WSN applications are classified into two categories: monitoring and tracking, with the first category including applications for indoor/outdoor environmental monitoring, health and wellness monitoring, power monitoring, inventory location monitoring, factory and process automation, and seismic and structural monitoring, and the latter category including tracking objects, animals, humans, and vehicles. A more detailed classification is made in reference [ALM05], where various WSN deployments are surveyed, and categorised into military applications, environmental monitoring applications, applications providing support for logistics, human-centric applications, and applications to robotics. A similar classification is made in reference [ASSC02a], according to which WSNs can be used in:

- Military applications for monitoring friendly forces, equipment and ammunition, battlefield surveillance, reconnaissance of opposing forces and terrain, targeting, and battle damage assessment.
- Environmental applications, for forest fire detection, biocomplexity mapping of the environment, flood detection, and precision agriculture.
- Health applications, for telemonitoring of human physiological data, tracking and monitoring doctors and patients inside a hospital, and drug administration in hospitals.
- Home applications, home automation and smart environment.
2.3. CONCEPTS, REQUIREMENTS AND CHALLENGES

Table 2.1: Example WSNs applications and scenarios

<table>
<thead>
<tr>
<th>Example scenario</th>
<th>Personal services</th>
<th>Community services</th>
<th>Industrial services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use cases</td>
<td>Lifestyle assistant</td>
<td>Wireless healthcare</td>
<td>Remote asset monitoring</td>
</tr>
<tr>
<td>Use cases</td>
<td>Mood based services, entertainment, nutrition, training</td>
<td>Wireless hospital, residential health monitoring, emergency coordination</td>
<td>Store of the future, food processing tracking</td>
</tr>
<tr>
<td>Types of sensor networks</td>
<td>Body sensor nodes for capturing user-related context, environmental sensors for capturing physical phenomena in the user’s surroundings</td>
<td>Body sensor nodes for monitoring vital functions, environmental sensors for capturing information about the patients’ close surrounding area</td>
<td>Sensors attached to products and equipment, environmental sensors for acquiring information about goods position, transportation conditions and environmental conditions</td>
</tr>
<tr>
<td>Types of information</td>
<td>Physiological, vital functions, movement, location, social presence</td>
<td>Condition and position of equipment and products</td>
<td>—</td>
</tr>
</tbody>
</table>

- Other commercial applications, like environmental control in office buildings, interactive museums, detecting and monitoring car thefts, managing inventory control, and vehicle tracking and detection.

In Table 2.1, we provide some examples of WSN scenarios from different applications spaces, as identified and classified in reference [D1206], and used for our work in reference [MAM+08]. The table includes examples of the types of sensor networks utilised and of the information captured, to highlight that the application space that WSNs are designated for influences not only the services that are provided to the end users, but also the contexts and types of the sensed data, and essentially the security requirements and the sensitivity of the information collected and communicated.

2.3 Concepts, Requirements and Challenges

The wide range of applications and the unique characteristics of WSNs at node, network and data level pose unique challenges in the design of security solutions. The resource constraints of the sensor nodes is not the only challenge. The diversity in the size of the deployments, in the organisation of the nodes in distributed, clustered or hierarchical, static or dynamic topologies, and in their supervised or completely autonomous operation, are also issues that need to be taken into consideration. Moreover, the purpose-specific nature of WSN deployments sets different requirements related to trust, privacy and security in comparison to ad hoc networks. In the following sections we discuss these requirements, outline the challenges in fulfilling them, and explain the notions that are used for trust, privacy, and security-related concepts.
2.3.1 Trust

WSNs depend highly on the distributed cooperation among the nodes for both network and data related operations. The network membership changes and the dynamic topology require the cooperation of nodes that may have been unknown during the pre-deployment phase. It is not trivial for a node to assess if another node is a legitimate network member, authorised to receive its sensor readings or to route its messages. Even legitimate network members may act selfishly when their energy supplies are limited, omitting to forward messages in order to preserve resources. Moreover, since WSNs may be deployed in unsupervised environments, and the nodes may not be tamper-resistant due to the high costs this would incur, sensor nodes can be physically compromised and exhibit malicious behaviour, acting as internal attackers.

Sensor networks are therefore susceptible to a variety of node misbehaviours. From compromised nodes to legitimate nodes that act selfishly, internal misbehaving nodes are a vulnerability that cannot be tackled by conventional security mechanisms for confidentiality, integrity, authorisation and access control. This vulnerability, along with the cooperative nature of WSNs, impose the need for assessing the trust relationships among network nodes, and using them as the basis for higher level security services, like data access control or trusted key exchange.

Trust concepts and notions Trust establishment models provide the means for representing, evaluating, maintaining and distributing trust within communities and networks. Trust and reputation systems are regarded as security mechanisms but, unlike the traditional hard security mechanisms, such as those used for authentication and access control, they are characterised as soft security mechanisms [JIB07]. As a notion, trust pre-exists security—as a natural phenomenon, it existed before the concept of security was invented [BH07]. The concept of trust, as used in different research areas like trusted computing, trusted platforms, trusted code and trust management, has received various interpretations [Gol06]. Throughout our work, we study the in-network trust relationships that can exist between network entities. We use the definition of trust as “The quantified belief by a trustor with respect to the competence, honesty, security and dependability of a trustee within a specified context” [Gra03]. Reputation is defined as a perception that a party creates through past actions about its intentions and norms within a community which is observing its members, while a recommendation is an attempt at communicating a party’s reputation from one community context to another [RRK05]. The main difference between trust and reputation systems is that trust systems produce a score that reflects the relying party’s subjective view of an entity’s trustworthiness, whereas reputation systems produce an entity’s public reputation score as seen by the whole community [JIB07].

A trust relationship is established by two parties, the trustor and the trustee, also referred to as the trust issuer and the trust target. The trust establishment process includes the specification of valid types of evidence and its generation, distribution, collection and evaluation [TB04]. The trust evidence, which form the basis for establishing trust relations, may be uncertain or incomplete, stable and long-term [EGB02].

---

4A misbehaviour is selfish if it aims at obtaining an advantage that can be quantitatively expressed in the units of wireless networking or in a related incentive system, while any other misbehaviour is considered to be malicious [BH07].
Trust evaluation is performed by applying context-specific rules, metrics and policies on the trust evidence. The result of the process is the trust relationship between the trustor and the trustee, usually represented as a certificate or as a numeric value, either discrete or in a continuous range. Trust relationships can be revoked on the basis of newly obtained evidence. Trust is transitive if it can be extended beyond the two parties between which it was established, allowing for the build-up of trust paths between entities that have not directly participated in a process of trust evaluation.

In general, the problem of formulating evaluation rules and policies, representing trust evidence, and evaluating and managing trust relationships is collectively referred to as trust management [BFL96].

Trust management requirements in WSNs Any given trust management solution should sustain semantic coherence of the trust evidence it utilises and respect a set of trust properties, that are discussed in Section 3.1.1. Apart from the generic trust requirements that apply to all types of networks, there exist some WSN-specific ones.

Trust management solutions for WSNs should be able to support the heterogeneity of the deployments and the nodes. Depending on the application domain of each WSN deployment, diversity can be identified in the trust evaluation requirements of the nodes, with some being required to cooperate only with a limited, known set of other parties (e.g. with their cluster head), and some regularly having to evaluate and cooperate with unknown nodes. Moreover, diversity may exist in the level of distrust that the nodes should exhibit during the network lifetime towards unknown parties. Trust management solutions should be able to support from simple nodes that have very restricted role, computational capabilities and should only trust the nodes they are pre-configured to trust, to highly adaptive nodes and gateways to other networks. They should provide the means for controlling trust evolution according to the level of distrust that the nodes should exhibit, and be flexible in managing trust between nodes of heterogeneous deployments.

Moreover, pre-deployment knowledge on the roles of the sensor nodes and their trust associations may be available, and trust management solutions for WSNs should be able to utilise it. Some sensor nodes may be clustered by deployment so that the trust relationships within the cluster can be preconfigured and be long-term and stable [RGB12]. Within predefined clusters like these of body sensor networks, the trust relationships between the nodes do not need to be continuously evaluated. Trust establishment models can exploit the pre-deployment knowledge that may exist, enable the pre-configuration of trust relationships and allow for some control on their evolution.

A requirement that is crucial for trust management solutions for WSNs is to provide support for controlled trust revocation. They should provide the mechanisms to inform the network nodes about trusted nodes that are found to be exhibiting malicious behaviour, and to revoke trust and isolated the malicious nodes. However, the susceptibility of WSNs to node misbehaviour can affect not only network operations, but also the trust management model itself. Malicious nodes may perform defaming attacks against legitimate nodes to spread bad reputation, either by directly spreading false evidence or by pretending to be victims of defaming attacks themselves to make legitimate nodes
look malicious [SP04]. Trust revocation should thus be controlled, which can be accomplished if some mechanism exists to protect from defaming attacks.

**Trust management challenges and constraints in WSNs** The resource constraints of the sensor nodes render most of the techniques commonly utilised for trust management inapplicable: asymmetric cryptographic operations, node behaviour monitoring and utilisation of past behaviour records are too complex or resource consuming for the sensor nodes. Asymmetric cryptography is considered expensive for sensor nodes in terms of both computational costs and communication requirements [SP04, AEAQ05]. Behaviour monitoring solutions require the nodes to keep their radio constantly on in order to monitor their neighbours, and to continuously re-evaluate their trust values. The trust management solutions for WSNs need to utilise only a subset of the available mechanisms and primitives, the ones that are the most lightweight in terms of resource requirements.

The wireless nature of communications, the dynamically changing topology and membership, and the lack of fixed infrastructure are also parameters that affect the design of trust management models for WSNs. The lack of stable centralised monitoring and management points preclude the use of trusted intermediaries, such as trusted third parties or certification authorities (CAs) as intermediaries for trust establishment. Each node needs to manage trust relationships with other nodes individually. Due to the vulnerability of the wireless links and the frequent topology changes, connectivity can not be guaranteed, and thus stable hierarchies of trust relations can not be supported. Trust relationships may change frequently due to changes in the network topology. Trust management solutions for WSNs should support distributed and cooperative trust evaluation, and each node should compute and maintain its own local trust values.

The availability of trust evidence is another challenge for trust management. As a result of the varying connectivity and the dynamic topology, the available evidence may be uncertain and incomplete, since it can only be sporadically collected and exchanged for each node under evaluation [TB04, EGB02]. Trust management solutions should thus provide the means to represent the quality of the evidence that is maintained, exchanged and utilised cooperatively by the nodes.

### 2.3.2 Privacy

The application scenarios of WSNs, as discussed in Section 2.2, include scenarios with sensors that are directly or indirectly related to people. Example sensor networks with a direct relation to people are the body sensor networks. The second case includes both environmental sensor networks where people are within the sensing environment, like WSNs deployed in workplaces, and sensor networks that monitor objects, like vehicles, that are related to people. Through such WSNs, context-rich, timely and, more importantly, sensitive information may be collected or could be inferred; for example, the information collected from an environmental WSN deployed in an office building may enable inferring the time each employee spends in his office through the data reported from noise or motion detection sensors.

As for any ubiquitous computing technology, privacy concerns need to be addressed in order to gain user acceptance in the long term. In order to preserve privacy, a system primarily has to ensure that sensed information is confined to the sensor network and is
accessible only to authorised parties, with the individuals being empowered to control how their personal information is obtained, processed, distributed, and used by any other entity. However, the narrow perception of privacy as secrecy and access control can not cover the wide range of issues that fall under the notion, which related more to controlling the disclosure of personal information in exchange of some perceived benefit, than to ensuring complete secrecy or anonymity. The fact that the perceived benefit and the purpose of data collection are the main determinants of personal data disclosure decisions has been verified by experimental studies on the value that individuals assign to location information [CKMD06].

Privacy can be addressed at multiple levels of the network stack and at different points of the sensed information flow, depending on issues such as which entities can be perceived as trusted by the users, and what the architectural and technological limitations of each deployment are. In the following paragraphs, the privacy requirements of WSNs are separately analysed in two dimensions: the requirements as they apply to the design of privacy preserving sensor networks, and the application and service privacy requirements as they are identified from the user’s perspective, which are mainly related to anonymity and location privacy.

**Privacy concepts and notions**  The most commonly used notion in the area of privacy protection is anonymity. Anonymity is defined as the state of not being identifiable within a set of subjects with potentially similar attributes, called the anonymity set, which is the set of all possible subjects who might cause an action [PH06]. The size of the anonymity set can thus be used as a naive degree of anonymity, i.e., the degree of how indistinguishable the users remain within their anonymity set. It is however been argued that “Anonymity is the stronger, the larger the respective anonymity set is and the more evenly distributed the sending or receiving, respectively, of the subjects within that set is” [PK00]. The degree of anonymity depends not only on the size of the anonymity set, but also on the distribution of probabilities that an observer can assign to the entities within the set for causing particular actions or for assuming particular roles. For example, an adversary observing anonymised network traffic could assign probabilities to each sender as being the originator of the message, based on the information that the system leaks, through traffic analysis, timing or message length attacks [DSCP02]. In order to accurately quantify anonymity, metrics based on entropy have been proposed [DSCP02, SD02], that take into account the probabilities that observers can assign to different members of the anonymity set.

Especially for the identities of the communicating parties, multiple levels of protection can be identified: sender, recipient, or mutual anonymity pertains to protecting the identity of the source, the destination, or both nodes. Sender and recipient unlinkability protects the relation of the nodes from inference by third parties. Unlinkability of two or more items of interest, like subjects or messages, means that within the system, from the attacker’s perspective, these items of interest are no more and no less related after his observation than they are related concerning his a-priori knowledge [PK00].

A concept that most privacy protection solutions utilise is pseudonymity, i.e. the use of pseudonyms instead of the identities of the subjects. Pfitzmann and Hansen observe that whereas anonymity and accountability are the extremes with respect to linkability of subjects, pseudonymity is the entire field between and including these extremes.
The scope of pseudonyms may differ according to the context of their use, providing different strength of long-term linkability. A static person pseudonym would be the weakest, while the establishment of transaction pseudonyms can be used to achieve strong unlinkability. A role-relationship pseudonym, different for each service provider and user role, is an intermediate case that would protect against identity information correlation on the service provider’s side. The same authors define identity management as the management of various partial identities of the individual, usually denoted by pseudonyms, the administration and design of identity attributes, as well as the choice of the partial identity and pseudonym to be re-used in a specific context or role.

Requirements for the design of privacy preserving WSNs Specific design level privacy requirements depend both on the application space of a WSN, which determines the sensitivity of the collected and communicated information, and the users’ preferences, roles and level of trust to the service provider. For WSN deployments to be privacy preserving, it primarily has be to ensured that the sensed data is protected from disclosure to unauthorised parties or adversaries and to illegitimate service providers. The fist two requirements below address the issue of how the data should be communicated in the network so that it is protected from eavesdroppers, while the next two address what data should be collected and released to data requestors so that the privacy of the individuals is preserved.

Confidentiality of the sensed and the aggregated data As in the case of all privacy preserving systems, data confidentiality must be ensured through message encryption. Encryption is the typical defence against eavesdropping, and the only method for preserving the confidentiality of the content of exchanged messages. For WSNs this requirement becomes even more crucial, because an outsider can induce information by correlating the results reported from multiple sensors surrounding an individual. Moreover, because of the in-network data aggregation operations, data of different granularity and sensitivity with respect to the user’s privacy is being communicated and needs to be protected.

Protection of the communications context Ensuring the confidentiality of the messages’ content does not always suffice, since an adversary might induce sensitive information by observing the communications’ contextual data, especially since they can be correlated with prior information about the people and the physical locations that are being monitored. For example, the disclosure of both spatial and temporal data through traffic analysis, may allow tracking the relative or actual (through correlation with prior knowledge) location of the mobile sensor nodes that might be carried by users, which would constitute a serious privacy breach. Independently of what encryption scheme is being used, the cipher texts should not allow induction of any information related to their context. Protecting the communication’s context, also referred to as transactional confidentiality by Pai et al., includes mechanisms to protect against identity, location or event inference due to the carrier frequency, the message rate, the message size or the routing behaviour. Techniques like timestamping, padding, using serial numbers, or frequent key redistribution can be used so that the communicated cipher
texts do not reveal information through their similarity or size. However, protecting the traffic patterns within the network and the identities of the nodes is not trivial, as it requires interference with the routing protocol.

**Privacy sensitive information disclosure** In settings with multiple data collectors, with varying levels of trust associated to them by each user, that offer a variety of services in exchange of information, users need to be empowered to reject, accept or negotiate the release of private data. Empowering sensor network users to control the level of information privacy according to the context, their role and communication partner mainly entails two actions: First, mechanisms should be provided to allow users to define their privacy preferences and to inform them what privacy policies are being announced by data requestors. Second, having provided the users with access to information about the service provider, the service offered and the purpose of data collection, mechanisms that enable the enforcement of their preferences would allow them to control the disclosure of their personal data, its level of detail, and the pseudonym utilised. The issues that thus mainly need to be addressed are related to data access control, data granularity control, and protection from inference through information correlation. The issue of context awareness for the release decisions is also crucial. It has been argued, for example, that the system needs to support special exceptions for emergencies in crisis situations, where safety outweighs privacy needs [HL04]. Moreover, the enforcement of privacy preferences should be made transparent and with minimal user interaction demands for the system to be non-intrusive. The controlled information disclosure requirement can be addressed in the middleware and application layer, since it is related to the content of the messages, as opposed to the previous requirement about the context of the communications. For this requirement to be satisfied, one has to assume that users trust the entities and devices up to the level where the privacy preferences can be enforced.

**Privacy sensitive information gathering** WSN deployments can not be considered privacy sensitive unless two conditions are fulfilled: First, mechanisms should be provided for the notification of individuals within the sensing areas. Second, data collection should be restricted to the minimum required level for the services to be provided. User’s notice and choice mechanisms address the first issue by providing awareness of data collection and requiring user consent. A first step towards addressing the second issue is restricting the network’s ability to gather data at a detail level that could compromise privacy [CP03], for example through depersonalising the results reported by sensors or through applying discrete information flows instead of continuous, when continuous data is not required by the applications. This requirement is more proactive than the previous one, in the sense that it protects user privacy at the point of information capture, before any data release decisions are made. For this requirement to be satisfied, it has to be assumed that users trust only the WSN that collects the data.

**Application and service level privacy requirements** While many applications that WSNs support shall require the identification of users in order to provide some services, others will only require anonymised or pseudonymised data. Examples of
the first case are medical applications gathering data from body sensor networks that monitor the vital functions of patients in hospitals, while examples of the second case are applications for traffic monitoring that gather data from environmental and vehicular sensor networks in order to extract traffic statistics. Providing support for anonymous data is an essential requirement in the application and service dimension, that can be addressed in multiple levels of the design requirements categorisation: The context of the communications needs to be protected in order to avoid the disclosure of the network identities of the user-carried devices, since this would allow adversaries monitoring network traffic and having some prior knowledge to correlate the network identities with the actual user identities. Privacy sensitive information disclosure mechanisms need to be provided for the data to be pseudonymised or anonymised before being released to applications that do not require user identification. Finally, privacy sensitive information gathering schemes could be applied to depersonalise the results reported by sensors during the data aggregation process.

Independently of how it is addressed, the anonymity requirement is set to allow individuals to use the network services, while protecting their identity both from adversaries and from data requestors. For legitimate network services that require user identification, the identity of users can be secretly shared between them and the service provider through the use of pseudonyms. However, even if data is anonymised or pseudonyms are being used, each user’s anonymity depends on his anonymity set: if it shrinks, his identity can be disclosed. A solution would be to use, together with the anonymity mechanisms, dummy traffic or background noise [PPW91].

Another requirement in the application and service dimension is the one of location privacy. Location is considered sensitive information both because the unauthorised access to information related to the actual or relative locations of users constitutes a serious privacy breach, and because accurate location information enables the correlation of network with actual user identities. Location information can be captured either by the located object itself by using some positioning technology like GPS, or by some external entity like a wireless network operator, using techniques such as signal triangulation. In the case of smart vehicles, for example, location information can be captured using the roadside infrastructure through distance bounding and multilateration [HCL04]. Hong et al. used the term mobile anonymity to describe the privacy issues that arise because of mobile nodes, and identified that mobile anonymity has to address venue anonymity, privacy of network topology, and privacy of motion pattern [HKG06].

Location information can also be captured through environmental sensor networks that monitor the presence of individuals in particular locations. Independently of how it is captured, location information is reported by the entity that captured it, called location information provider, to some provider of location based services. These services can either be user-initiated, like tourist information services that report the areas of interest around the user, or continuous location tracking services, where the users’ location is frequently reported, as in the case of traffic monitoring applications. Location privacy concerns are more related to the case of location tracking services, where location information discloses the user’s movements.

Similarly to the anonymity requirements, location privacy can be addressed at multiple levels: The context of the communications can be protected in order to avoid location tracking from adversaries that monitor the network traffic and can identify
the messages transmitted by user-carried devices. Privacy sensitive information disclosure mechanisms can ensure that the external service providers receive data of an accuracy level that is in accordance with the user’s preferences. Finally, privacy sensitive information gathering schemes can be used both to notify the users within sensing areas, and to ensure that the level of detail of the captured location information is the minimum required.

Privacy protection challenges and constraints in WSNs  While several privacy mechanisms, tools and applications have been proposed to enhance privacy and anonymity in traditional networks, these solutions can not be directly applied to the case of WSNs. Due to the energy constraints of the sensor nodes, routing schemes for WSNs are designed to optimise the multihop path selection, as opposed to the mix-based approaches that randomise the path from the source to the destination to achieve untraceability. Moreover, the capabilities of the nodes pose limitations on the range of cryptographic primitives they can support. Techniques like onion routing, which relies on extensive asymmetric cryptographic operations for the construction of the onion using layers of encryption that intermediate nodes can strip off using their private key, would impose high computational overhead to all nodes in each path. The use of dummy traffic or background noise, that has been proposed as an extension to the MIX approach in order to achieve indistinguishability [PPW91], would exhaust the energy supplies of the sensor nodes. Overall, the privacy mechanisms that require either extensive encryption operations or additional network traffic because of the message routing strategies they include, incur overheads that are prohibitive for use in resource constrained sensors.

The wireless nature of sensor network communications and the standardised communication technologies that are employed make it even more challenging to satisfy most of the requirements, as the wireless medium exposes information about the network traffic. WSNs are vulnerable to eavesdropping and traffic analysis attacks, since an adversary does not need to gain physical access to the networking infrastructure. Moreover, in the general case, WSNs are infrastructureless and dynamic. The lack of central servers and static base stations, the dynamically changing network topology, the possibility of addition or deletion of sensor nodes through all stages of their life cycle, all combined with the scale of the deployments (hundreds or thousands of sensor nodes), set strict requirements on the privacy schemes that can be used: they need to be distributed, flexible, scalable, and cooperative. Privacy schemes that require some designated proxy to protect the anonymity of the source nodes, like Anonymizer, can not be directly applied. Privacy policies negotiation schemes like P3P require the existence of network nodes that make privacy reference files and privacy policies available, but for WSNs the availability of such nodes can not be guaranteed. The lack of centralised host relationships and the transmission range limitations also affect the routing mechanisms of the data packets, that will typically follow multi hop routes before arriving at their final destination. The trustworthiness of the intermediate nodes of each path can not be guaranteed a priori.

The fact that in-network data processing and aggregation is performed sets additional security requirements for sensor-to-aggregator communication. In traditional
networks, where each node communicates with some base station, most data and identity confidentiality requirements can be satisfied by end to end encryption and the use of pseudonyms. In WSNs, however, for data aggregation purposes, intermediate node authentication is required. Moreover, possibly untrusted aggregator nodes gain access to large streams of data that may be sensitive. The increased complexity of the information the flow makes it challenging to enforce privacy sensitive data gathering mechanisms. Issues related to how user pseudonyms can be handled in the presence of aggregator nodes, and which raw or aggregated data can be anonymised at each point within the network also need to be resolved.

2.3.3 Security

All applications of WSNs, including those discussed in Section 2.2, involve information and assets that are valuable and need to be protected. The security of the network nodes, of the data and of the communications is an essential requirement for the end applications to be reliable. At a lower level than trust management and privacy protection, WSNs require the provision and management of core security services to ensure the authenticity, the confidentiality and the integrity of the data generated and communicated through the network. However, the characteristics of WSNs make it challenging to apply traditional security mechanisms. At the same time, their integration with other networks and platforms and the heterogeneity that exists between the capabilities and the security requirements of the nodes and the deployments necessitate the application of flexible and adaptable security solutions.

Security concepts and notions  Network security includes the provisions to protect the network and the network-related resources from unauthorized access or tampering. Secure systems are those that remain dependable in the face of malice, error, or mischance, while a vulnerability is a system property which, in conjunction with an internal or external threat, can lead to a security failure [And01]. In the networks security domain, internal or external threats include the passive or active attacks that may be launched, where the active attacks aim to disrupt or tamper with the network operations.

Secure systems are expected to meet certain requirements [BH07]. These are summarised to authentication of the entities participating in the system operations, which entails providing the means for the verification of their identities, authorisation and access control to resources and information, confidentiality and integrity of the information generated and communicated, which entails providing assurance that the information processed by a system is protected against intentional or unintentional unauthorised access, modification, insertion, substitution or deletion, and availability of the system’s resources. The authenticity of the information communicated in a system entails not only the authentication of its origin and the protection of its integrity, but also of its freshness. Forward and backward secrecy are notions that are related to the time domain and are commonly used in networks security; they require that network entities do not have access to information that was communicated before or will be communicated after their authorisation period.
2.3. CONCEPTS, REQUIREMENTS AND CHALLENGES

Security requirements of WSNs  Any security solution for WSNs should provide the means for authentication, encryption and integrity protection of sensor node communications. Data freshness is also an important security requirement for WSNs, since sensor nodes may produce and communicate time-critical data, and adversaries should be prohibited from replaying old messages. Moreover, since new sensors may be deployed at any point in the network lifetime and old sensors may fail or be compromised, forward and backward secrecy are also important to the security of WSNs [CMYP09]. Defence mechanisms should be provided to increase the resilience of the network against active internal attacks launched through compromised nodes, that may include reporting false readings, tampering with the data aggregation operations, and interfering with the routing protocol. Compromised nodes may defame legitimate nodes or collude for increasing the effectiveness of the attack. The security requirements of WSNs therefore include the detection of compromised nodes, secure routing and secure data aggregation.

The requirement for flexible and adaptable security management stems from the diversity between the nodes and the deployments in terms of their capabilities and their security requirements. The diversity in the application fields of WSNs influences the risks and consequently the security needs of the nodes. In any WSN there may exist from simple nodes that have very restricted role, computational capabilities and security requirements, to critical nodes and gateways to other networks. Some nodes may generate information whose correctness and freshness is crucial, like the sensor nodes used for the remote asset monitoring applications of Table 2.1, thus requiring strong integrity protection, while others may generate information that has high confidentiality needs, like the sensors utilised in wireless healthcare applications. This imposes the need for security management solutions that can support and exploit this diversity, and can facilitate the secure integration of WSNs with other networks.

The requirement for self-organization is commonly acknowledged [Rom08, WLSC07, CMYP09]. WSNs may need to operate unattended, without intervention, and to adapt to changing contexts and environment settings. The security solution should offer the required scalability and flexibility, while enabling the enforcement of pre-defined settings and policies in each node. Moreover, as with any solution designed for WSNs, energy preservation is a crucial requirement. In order to minimise the resource overheads, the security services that are available on each node should be energy efficient and the security mechanisms conservatively selected and applied.

Security challenges and constraints in WSNs  Chen et al. summarised the security challenges in sensor networks into minimizing resource consumption and maximizing security performance, increased vulnerability to link attacks ranging from passive eavesdropping to active interfering, application of in-network processing involving intermediate nodes in end-to-end information transfer, unsuitability of traditional wired-based security schemes, increased complexity due to large scale and node mobility, and dynamic network topology [CMYP09]. The unattended operation, the exposure to physical attacks, and the lack of centralised management points are also obstacles in securing WSNs [WLSC07].

Due to the inherent properties of WSNs, security mechanisms and primitives used in traditional networks can not be applied. Especially for cryptographic key distribution
and management, the applicable solutions have to respect the resource constraints of the nodes, the infrastructureless and dynamic nature of the networks, and the possibility of node compromise. Asymmetric cryptography can not be extensively applied by the sensor nodes [SP04, AEAQ05], while they are capable of holding a limited amount of symmetric keys. As a result, the key management solutions need to be based mainly on symmetric cryptographic operations, without requiring the nodes to store keys for all the nodes that they communicate with and, at the same time, ensure that the effects of node compromise are localised and do not violate the forward and backward secrecy requirements.

The vulnerability of WSNs to node compromise makes it challenging to secure them against a range of active insider attacks. This could be avoided if tamper-resistant hardware was used on the sensor nodes. Tamper proof modules would enable the secure storage of the security-critical information (like the cryptographic keys) on the nodes, preventing attackers from retrieving it when they are captured. They would, however, also increase the cost of the sensors, and are therefore not considered to be a viable solution. To defend against node compromise, other techniques that have been proposed include code attestation and code obfuscation techniques. The former aim to detect whether the code executed by the sensors is legitimate, in order to identify compromised nodes, while the latter aim to increase the complexity of analysing the memory of a node [Rom08].

2.4 Summary

WSNs are composed of numerous sensor nodes, spatially distributed over sensing fields, that capture, process and communicate various types of contextual information related to their environment. They are used in a wide variety of scenarios and applications. They possess a number of unique characteristics: At the node level, the sensor nodes are constrained in computation capabilities and energy supplies. At the network level, they are heterogeneous, dynamic, and lack centralised management points. At the data level, in-network aggregation of the sensor readings is applied. As a result, it is challenging to devise security, trust and privacy solutions that can be applied uniformly in WSN deployments.

Trust management is an important security service for WSNs, because of their cooperative nature and their vulnerability to node compromise and misbehaviour. Trust management solutions for WSNs should be able to support ad exploit the heterogeneity of the deployments and the nodes in terms of their capabilities and trust evaluation requirements. They should also utilise the pre-deployment knowledge on the roles of the sensor network nodes and their trust associations, support distributed, cooperative trust evaluation and revocation, and entail acceptable resource consumption.

WSNs collect and communicate information of varying sensitivity with respect to the privacy of the monitored subjects. Privacy in WSNs can be addressed at multiple levels of the network stack and at different points of the sensed information flow, depending on issues such as which entities can be perceived as trusted by the users, and what the architectural and technological limitations of each deployment are. The privacy requirements that were discussed include the protection of the communications
context, privacy sensitive information disclosure and privacy sensitive information gathering.

The provision of core security services for authentication, encryption and integrity protection is a prerequisite for secure WSNs. Other required security services include the detection of compromised nodes, secure routing and secure data aggregation. Moreover, WSNs require flexible and adaptable security management to address the need for self-organisation and to support the heterogeneity between the nodes and the deployments in the capabilities of the nodes and their security needs.
Chapter 3

Related Work on Trust, Privacy, and Security Management for WSNs

This chapter presents the solutions that have been proposed to address the trust, privacy and security requirements of WSNs. It organised in three sections: Section 3.1 discusses the properties of trust, the trust management approaches in traditional networks, and the trust management models that have been proposed for WSNs, with the hybrid trust management models analysed in Section 3.1.5 being the most related to our work; Section 3.2 discusses the privacy protection approaches, categorised according to the privacy requirements in the design dimension, with the privacy sensitive information disclosure approaches of Section 3.2.3 being the most related to our work; finally, Section 3.3 discusses the related work on core security services and security management. Each section concludes with an analysis of the solutions that were presented and a discussion on their applicability on WSNs.

3.1 Trust Management

Trust management models provide the means for representing, evaluating, maintaining and distributing trust within the network nodes. Trust management is a relatively new term, introduced in reference [BFL96] to distinguish a new and more general decentralized approach to access control from traditional methods [Gol06]. Trust, as a notion, is however well defined. Generic trust properties have been specified to govern the way trust relationships are represented, established and maintained. This chapter begins with a discussion on these trust properties, that any trust management solution needs to respect.

A number of solutions have been proposed for different types of networks, including traditional wired networks, ad hoc, peer to peer and sensor networks, and there exist various approaches for trust evaluation and establishment. In Section 3.1.2 we discuss these different approaches, and then in Sections 3.1.3, 3.1.4 and 3.1.5 provide a state-of-the-art review of trust establishment models that have been proposed for ad hoc and sensor networks. The models are categorised into certificate-based, behaviour-based
and hybrid models, according to their scope, purpose and admissible types of evidence. A similar distinction was made by Bonatti et al. [BDOS05], who identified that two different perspectives exist on trust management: in the strong and crisp approach decisions are founded on logical rules and verifiable properties encoded in digital credentials, while in the soft and social approach decisions are based on reputation measures gathered and shared by a distributed community. A distinction between hard and soft trust relationships was also made by Lin et al. [LV07], where hard trust relationships are defined as those that can be derived from underlying cryptography based security mechanisms, such as digital certificates, and soft trust is based on trust relationships derived from localized and external observations of system entity behaviour.

Moreover, hierarchical and distributed models are discussed, based on the type of ad hoc and sensor networks they are designed for. Hierarchical models assume the existence of a hierarchy among the nodes, based on their capabilities or level of trust. These models may specify, for example, that certification authorities or trusted third parties provide on-line or off-line evidence. Distributed models assume that there is no fixed infrastructure, and the responsibility of acquiring, maintaining and spreading trust evidence is equally spread among the network nodes. This distinction mainly applies for certificate-based models, since the behaviour-based ones are all designed for distributed networks. The models are analysed both on criteria specific to each category and on common criteria, including the fulfilment of the trust management requirements of Section 2.3.1 and their applicability on WSNs.

3.1.1 Trust Properties

The basic properties of trust are derived from its definition. Focusing on the in-network trust relationships that can exist between network entities, we use the notion of trust as “The quantified belief by a trustor with respect to the competence, honesty, security and dependability of a trustee within a specified context” [Gra03]. A definition which involves the concept of risk is that “Trust is the extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible” [JGK06].

Trust is context-specific; it applies to the honesty, reputation and reliability of other parties for a specific purpose [JGK06]. A trust relationship is not absolute: a trustor trusts a trustee with respect to its ability to perform a specific action or provide a specific service within a context [Gra03]. The terms trust context, trust purpose, and trust scope are often used interchangeably to express the semantic content of an instantiation of trust. Generic classes of trust contexts include service provision trust, resource access trust, delegation trust, certification trust, and infrastructure trust [GS00]. An example instantiation of service provision trust includes trust in message routing, while resource access trust could refer to data disclosure. This context or purpose may be predetermined for the application of a trust management model. A common property for all trust management models is the semantic coherence of trust values and trust evidence with respect to the context or purpose that the model is designed or applied for.

Trust relationships are asymmetric and subjective. A’s trust in B is not necessarily the same as B’s trust in A, while different entities can assign different types and levels of trust towards B [GS00]. Trust diversity in terms of trustors A, trustees B and
3.1. TRUST MANAGEMENT

Figure 3.1: Trust properties: Diversity

Figure 3.2: Trust properties: Transitivity

Trust scopes $\sigma$ have been expressed with basic diversity attributes, as illustrated in Figure 3.1 [JGK06]. Except from these attributes, a measure of the level of trust is associated with each trust relationship, which can be discrete or continuous in some form, like probability or percentage. A fifth attribute of a trust relationship is its time component, since trust relationships can change with the interactions that take place between the trustor and the trustee or with time passing.

Trust can be transitive, under certain semantic constraints that have been defined in reference [JP05]. Figure 3.2 (a) illustrates the basic trust transitivity concept: if entity $A$ trusts entity $B$ for referring other entities for a particular function $\sigma$, and entity $B$ trusts entity $C$ for $\sigma$ then, by transitivity, $A$ trusts $C$ for $\sigma$. Trust for referring other entities is called referral trust, represented by $r$ in the figure, while trust for a particular function is called functional trust, represented by $f$. Functional trust is direct when it is derived through direct experience that an entity has for the competence, honesty, security or dependability of another entity.

Figure 3.2 (b) illustrates a trust path for trust derivation through transitivity. The types of trust that it depicts in the trust path are set according to the Functional Trust Derivation Criterion of Jøsang et al., which defines that “Derivation of functional trust through referral trust, requires that the last trust arc represents functional trust, and all previous trust arcs represent referral trust” [JGK06]. The same authors have defined the Trust Scope Consistency Criterion, according to which “A valid transitive trust path requires that the trust scope of the functional/last arc in the path be a subset of all previous arcs in the path”. In the simplest case of trust paths, every arc can have the same trust scope $\sigma$.

The measure of the indirect trust derived from the construction of a trust path is computed using a link operator $\otimes$, which must be associative in order to allow computing trust along complex paths without considering the sub-paths evaluation order.
An example associative link operator is the product operator. An additional requirement that applies to the link operator is that the concatenation propagation of trust does not increase trust, since uncertainty increases through propagation. For the trust relationships of Figure 3.2 (a) and (b), this implies that the computed $f_{A,C}$ should be less than the functional and referral trust values in the trust path that was used for its computation, i.e., $f_{A,C} \leq \min(|r_{A,B1}|,|r_{A,B1}|,|f_{B2,C}|)$.

Parallel trust paths can be constructed when referrals from several trusted entities are collected and aggregated for the derivation of indirect functional trust, as illustrated in Figure 3.2 (c). The measure of the indirect trust derived from combining parallel trust paths is computed using an aggregation operator $\odot$ on the measures that are computed separately from each trust path. Like the link operator, the aggregation operator should also be associative and, in addition, it is desirable to be idempotent. This last property, meaning that multiple applications of the aggregation operation should not change the result, is especially important for complex trust networks that rely on trust transitivity, since there may exist overlapping on the trust paths. For example, the value $f_{B2,C}$ in the figure may also be indirect, having been derived using a recommendation of $f_{B1,C}$ from $B1$. Using an idempotent operator, like the maximum, would not allow the overlapping of paths to affect the result.

The existence of parallel trust paths can increase the confidence on the indirect functional trust that is derived. This was also expressed by Sun et al. as a requirement that multipath propagation of trust does not reduce trust. In Figure 3.2 (c), if the same recommendations are received for $C$ ($f_{B1,C} = f_{B2,C}$) by equally trusted recommenders ($r_{A,B1} = r_{A,B2}$), then $f_{A,C} \geq f_{A,C\text{through}B1} = f_{A,C\text{through}B2}$. This requirement is essentially set because multipath recommendations should not increase uncertainty on the computed indirect trust measure. It is complemented by another observation of the same authors: that the recommendations from independent sources can reduce uncertainty more effectively than recommendations from correlated sources, and thus trust based on multiple recommendations from a single source should not be higher than that from independent sources.

### 3.1.2 Trust Management Approaches

Trust management is closely related to authentication and access control. The roots of trust management in conventional systems, such as databases and operating systems, are the identity-based access control mechanisms, which require from each subject to be uniquely identified through some authentication mechanism for proving its trustworthiness. The means for authentication in networked applications may be digital certificates, which are used as a proof of either identity or membership in a trusted group. These may be X.509 or Pretty Good Privacy (PGP) public-key certificates. The first are issued and managed through hierarchically organised certification authorities, while the latter though a web-of-trust formed in a distributed way by users that act as certification authorities.

PGP and X.509 certificates do not enable the definition of the authorisation rights of their holders. The first trust management systems were inspired by the observation that in open decentralised environments authorisation decisions can not be based solely on the proof of a subject’s identity. The trust management approach to distributed
systems security was developed as an answer to the inadequacy of traditional authorization mechanisms in terms of decentralisation, scalability, expressibility and extensibility [BFIK99]. Blaze et al. were the first to explicitly define the term trust management as a unified approach to specifying and interpreting security policies, credentials and relationships which allow direct authorisation of security-critical actions [BFL96]. The PolicyMaker system [BFL96] and its successor, KeyNote [BFK98], were the first decentralised trust management systems to implement this definition. Instead of using certificates for binding identities to public keys, these systems use trust management credentials, which subsume the role of certificates, and enable binding public keys to predicates that describe the actions that their holders are trusted to perform. Since PolicyMaker, a number of systems have been proposed for distributed trust management through security policies and trust management credentials [GS00, AG07].

A common characteristic of all trust management systems is that they require the disclosure of credentials and trust evidence in order to establish trust. This however may lead to loss of privacy in open systems, such as the Internet, where trust may need to established among entities with no prior knowledge of each other. Trust negotiation is a trust establishment approach for protecting the privacy of both the trustor and the trustee. It treats trust evidence as potentially sensitive resources, and regulates their exchange during trust establishment [WL06].

A trust negotiation between two entities consists of iteratively disclosing trust evidence, that may be in the form of certified digital credentials, so that trust is built incrementally according the disclosure policies. The negotiating strategy and the disclosure policies of the two entities define which evidence to disclose and which conditions need to be fulfilled at each step of the negotiation. A number of strategies have been proposed for trust negotiation [BFS04], offering different levels of privacy protection and varying efficiency with respect to the delays and the communication and computational costs that the negotiations incur.

Apart from trust management through security policies, rules and credentials, a number of different approaches to trust representation and establishment have been proposed. Logic-based approaches use forms of first order predicate logic or modified modal logic to represent trust and its associated concepts in environments with distributed agents [GS00]. Each logic-based formalism extends its primitive constructs to express trust rules and include features such as temporal constraints and predicate arguments. In the heuristic formalism proposed by Marsh for artificial trusting agents, general trust of $x$ in $y$ is modelled with statements of the form $T_x(y)$, extended with arguments to represent context $\alpha$ and time $t$ as $T_x(y, \alpha)^t$, and formulas are devised for calculating situational trust based on the agent general trust, cooperation, risk, competence, and reciprocation [Mar94]. Other formalisms are based on modal logic and utilise the modal operators of necessity $\Box$ and possibility $\Diamond$ to express belief, knowledge, temporal progression and other modalities [GS00]. In the modal logic-based model proposed by Rangan, trust statements are defined in the form of $B_i p$, denoting belief of agent $i$ in proposition $p$. Any sentences in the logic may be added to the logic as axioms, and these axiomatic sentences are considered as trust specifications [Ran88].

An application field in which trust has been extensively studied is autonomous agents and open multi-agent systems. A number of solutions have been proposed for modelling the interactions of trusting agents, some of which follow a game theoretic
approach to model and measure uncertainty, confidence, trust, dependency and collaboration [AG07]. Ramchurn et al. classified the approaches on trust in multi-agent systems into individual-level trust, whereby an agent has some beliefs about the honesty or reciprocative nature of its interaction partners, and system-level trust, whereby the actors in the system are forced to be trustworthy by the rules of encounter that regulate the system [RHJ04]. The trust models at the individual level are classified as either learning based, reputation based, or socio-cognitive based, according to the types of evidence that they require. At the system level, the solutions are subdivided into truth-eliciting interaction protocols, reputation mechanisms that foster trustworthy behaviour, and security mechanisms that ensure new entrants can be trusted. Other studies have found that the utilisation of trust and reputation mechanisms by autonomous agents handling electronic transactions has a significant positive effect on the honesty of the agents and the overall reliability of the transactions [JHF03].

Apart from electronic communities, the semantic web is another application field for trust. As autonomous agents need to make judgements when alternative sources of information are available, concepts of trust can be utilised for evaluating the degree of belief in, possibly contradicting, statements from different sources. A number of trust models have been proposed for filtering and selecting information from the semantic web [AG07]. Richardson et al. assume that the content on the Semantic Web is in the form of logical assertions that are not believed with certainty [RAD03]. They propose a solution for evaluating the degree of belief in a statement that is explicitly asserted by one or more sources on the Semantic Web, as a function of the trust in the sources providing it. Once the agent’s belief in the statements of each source are computed through the combination functions that are supported, then the belief in the derived statements can be calculated.

Trust management has also been applied to peer to peer communities, with EigenTrust [KSGM03] being one of the most commonly cited works in the area. The aim of the EigenTrust algorithm was to decrease the number of downloads of inauthentic files in a peer-to-peer networks by assigning each peer a unique global trust value based on his history of uploads, and having peers use these global trust values to choose the peers from whom they download. Each peer interacting with any node $i$ rates the transaction as positive or negative, and uses this history of transactions to compute its local trust value. EigenTrust computes a global trust value for $i$ in a distributed manner, by calculating the left principal eigenvector of a matrix of normalized local trust values, thus taking into consideration the entire system’s history with each single peer.

A number of commercial and commonly available trust and reputation systems are currently operating for various communities. These are reviewed and discussed by Jøsang et al. in reference [JIB07], and include reputation systems for electronic markets and auction websites (such as eBay’s feedback system, the Epinions Web of Trust, and BizRate), for knowledge sharing and expert websites (such as the rating system of AllExperts and the reputation engine of the Advogato community of open-source programmers), and for discussion forums (such as the Slashdot reputation system).

Most of the trust management approaches that were discussed in this section cannot, however, be directly applied to WSNs due to their inherent characteristics, analysed in Section 2.3.1. The approaches that are based on asymmetric cryptographic operations
are too resource consuming for the sensor nodes, especially in terms of energy. The approaches that assume the existence of trusted third parties can not be applied to WSNs due to the lack of fixed infrastructure, the varying connectivity and the dynamically changing topology and membership. While in WSNs the general aim is to minimise the number of message transmissions, the trust negotiation approaches increase the number of messages that need to be exchanged for any trust establishment. Approaches that utilise past behaviour records for trust evaluation require too much memory to be applied by the sensor nodes. The ones that require rich trust establishment evidence to be accumulated are inapplicable both due to their increased communication requirements and due to the limited availability of trust evidence in WSNs.

Considering these limitations and the characteristics and trust evaluation requirements of the sensor nodes, the application field for trust that is the mostly relevant to WSNs is the field of ad hoc networks. Due to the similarities that exist in the structure and the trust evaluation requirements of the ad hoc network nodes with the ones of the sensor nodes, the approaches that have been proposed for trust management are similar. These approaches include certificate-based approaches, discussed in Section 3.1.3 and behaviour-based approaches, discussed in Section 3.1.4. The latter include solutions that are based on the Bayesian theory, on subjective logic, on statistical evaluations, etc. The solutions that we have identified as the ones most related to our work are the ones that combine different types of trust evidence, and are separately discussed in Section 3.1.5.

3.1.3 Certificate-Based Trust Establishment

Certificate-based approaches aim to define mechanisms for the knowledge on the trust relationships within the network, usually represented by certificates, to be spread, maintained and managed either independently or cooperatively by the nodes. Trust decisions are mainly based on the provision of a valid certificate, that proves that the target node is considered trusted either by a certification authority or by other nodes that the issuer trusts. It is generally outside the scope of certificate-based models to evaluate the behaviour of nodes and base trust decisions on that evaluation. The approaches discussed in this section are equivalent to the hard trust approaches of [LV07], defined as those that can be derived from underlying cryptography based security mechanisms such as digital certificates, and the policy-based approaches of [BDOS05], defined as those where trust relies strong security mechanisms and verifiable properties encoded in digital credentials.

The most widely used approach for certificate-based trust establishment is the traditional, hierarchical, public key infrastructure model formed as an organisation of certification authorities. The use of on-line certification authorities for ad hoc networks, however, is problematic for connectivity and service availability reasons. Three generic approaches for certificate-based trust establishment have been proposed, two of which are hierarchical and one is distributed. In the first hierarchical approach, trust is represented by certificates signed by offline trusted third parties, whose public keys the trustors need to possess to verify the signatures. The second is the fully distributed self-organised public key management approach that follows the web-of-trust model, where trust is evaluated using certificate chains. The third one utilises secret sharing mechanisms to distribute trust to an aggregation of nodes that collaboratively provide
Table 3.1: Characteristics of certificate-based trust models. For each model, the type of evidence that is required for trust evaluation by node \(i\) for node \(j\) is categorised as: (C)-Certificate/public key, (RI)-Trust revocation information like certificate revocation lists (CRLs) or similar structures, (CF)-Confidence factor on evidence/recommendations. The Evidence Provision column outlines the input required by the evaluation mechanism performed by \(i\) from each of the parties involved in the evaluation. The Pre-Configuration column includes the information each node \(x\) in the network must possess before entering the network. The representations used are: (\(K_x\))-Private key of node \(x\), (\(C_y^x\))-Certificate issued for \(x\) by \(y\). \(A\) represents the certification authority. The set \(N\) represents all nodes in the network.

<table>
<thead>
<tr>
<th>Trust Model</th>
<th>Required Evidence</th>
<th>Parties Involved</th>
<th>Evidence Provision</th>
<th>Pre-Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchical Trust Models</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[VOT01]</td>
<td>+ +</td>
<td>(i,j,n) CAs</td>
<td>(i,C_A^i&amp;\text{CRLs}, j,C_A^j)</td>
<td>(C_A^i,K_x&amp;nC_A^i)</td>
</tr>
<tr>
<td>[Dav04]</td>
<td>+ +</td>
<td>(i,j,n) offline CAs</td>
<td>(i,C_A^i&amp;RI, j,C_A^j)</td>
<td>(C_A^i,K_x&amp;nC_A^i)</td>
</tr>
<tr>
<td>Distributed Trust Models</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[HBC01]</td>
<td>+</td>
<td>(i,j)</td>
<td>(i,\text{REP}_i&amp;\text{REP}_j)</td>
<td>(K_x,\text{REP}_x&amp;nC_A^y&amp;nC_A^y)</td>
</tr>
<tr>
<td>[EGB02]</td>
<td>+ +</td>
<td>(i,j) any other</td>
<td>(j,\text{any other})</td>
<td>Keys, Policy, Metrics</td>
</tr>
<tr>
<td>Distributed Certification Authority Models</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[ZH99]</td>
<td>+</td>
<td>(i,j,t+1) partial CAs</td>
<td>(i,C_A^i, C_A^j&amp;C_A^t)</td>
<td>(x,K_x, C_A^x&amp;C_A^{t+1})</td>
</tr>
<tr>
<td>[YK03]</td>
<td>+ +</td>
<td>(i,j,t+1) partial CAs</td>
<td>(i,C_A^i&amp;RI, C_A^j&amp;C_A^t)</td>
<td>(x,K_x, C_A^x&amp;C_A^{t+1})</td>
</tr>
</tbody>
</table>

certification authority services. This is considered to be a hierarchical approach, since trust is distributed among a subset of network nodes, that are designated to represent a certification authority. Table 3.1 summarises the characteristics of the certificate-based models that are discussed in the following sections.

3.1.3.1 Hierarchical Trust Models

The hierarchical trust approaches that have been proposed for ad hoc networks are based on and introduce modifications to the X.509 [X.505] trust model. The model is hierarchical, requiring the existence of certification authorities which are organised into a certification authority tree, so that all public-key certificates in the network are issued and managed by some certification authority that can be linked to this tree.

A hierarchical progressive trust negotiation scheme for ad hoc networks is introduced by Verma et al. [VOT01]. Off-line trusted third parties are set responsible both for issuing the certificates required for each node, including a network address certificate and at least one identity certificate, and for issuing certificate revocation lists. The model includes the notion of certificate release policies that are used to enforce a negotiating strategy for each node, in order for the disclosure of information to be controlled during trust negotiation. Each node in the network stores the certificates of the third parties and the certificate revocation lists they have issued, along with the local certificates to be used in trust negotiation. Trust negotiation is carried out by incrementally exchanging certificates.

In reference [Dav04], Davis proposes a scheme where trust is similarly represented by certificates signed by offline certification authorities, whose public keys are maintained.
locally by the trustors to verify the signatures. The scheme also enables the explicit revocation of certificates without input from trusted third parties. Any node \( j \) is considered trusted by any node \( i \) once it presents a certificate that has not expired, has not been revoked, and \( i \) can verify using the public key of a third party. Nodes have to maintain locally their private keys and the public keys of the third parties.

To handle certificate revocation without input from third parties, nodes maintain certificate status tables and profile tables which are used to determine whether or not a given certificate should be revoked. The profile tables kept by all nodes in the network should be consistent. In case inconsistencies are found by any node, accusations are broadcasted for the nodes that sent the inconsistent data. The two tables of all nodes are updated when an accusation is broadcasted, thus the accused node's certificate is revoked and network access is denied. In order to defend against bad mouthing attacks, the authors propose the final decision on certificate revocation to be based on a sum of weighted accusations from independent nodes.

Trusted authorities are also used by the scheme proposed by Boukerch et al. [BXEK07]. Instead of managing certificates and trust evidence, these authorities are responsible for generating and launching mobile agents on the network nodes. The scheme is based on a clustered WSN with backbone, and its core is the mobile agent system. The mobile agents provide trust and reputation management services on the hosting nodes, including the administration of their trust information and their reputation certificates, so that each node's trust evidence is stored locally.

### 3.1.3.2 Distributed Trust Models

In contrast to the hierarchical models, where certificates are issued by trusted third parties, distributed models provide mechanisms for trust evaluation between network nodes in a cooperative, self-organised manner. The PGP model [Gar95] was the first to enable users to act as independent certification authorities, expressing their trust on other users (the confidence on their identity) by signing their public keys. The public key certificates of this so-called “web of trust” approach are assigned with trust levels and confidence levels. However, although certificates are issued by the users, publicly accessible certificate directories are required for their distribution, which makes the model inapplicable for ad hoc networks.

A model that uses the web of trust approach of the PGP model, without requiring certificate directories for the distribution of certificates, is proposed by Hubaux et al. [HBC01]. The relationships between users are modelled as a directed graph, called trust graph, whose edges represent public key certificates. Each user maintains a subset of the trust graph as a local repository of certificates issued by himself or other users in the system. A subgraph selection algorithm is proposed, which is called Shortcut Hunter Algorithm. When a user \( i \) wants to obtain the public key of user \( j \), they merge their subsets of trust graph stored in their repositories and \( i \) tries to find a trust route in the form of a certificate chain from \( i \) to \( j \) in the merged repository.

To deal with dishonest users issuing false certificates, an authentication metric is introduced as a function that takes two users \( i \) and \( j \) and a trust graph as inputs and returns a value that represents the assurance with which \( i \) can obtain the authentic public key of \( j \) using the trust graph. This model is considered practically inapplicable for ad hoc networks because it requires extensive public-key operations for constructing
The distributed trust establishment model proposed by Eschenauer et al. \cite{Eschenauer2002} takes a broader view on the inputs required for node trust decisions by accepting as trust evidence not only certificates and public keys, but also information like identities, locations, or independent security assessments. The type of information required depends on the policy and the evaluation metric each node uses to establish trust. Trust metrics are used to assign confidence values to available pieces of evidence that may be uncertain or incomplete, while policy decisions are defined as a local procedures that, based on the evidence and the confidence assigned to it, output a trust decision.

The model is fully distributed. Any node can generate trust evidence about any other node and make it available to others through the network, as long as it signs it with its private key and specifies its lifetime. Evidence revocation is supported through revocation certificates and by the generation and distribution of contradictory evidence. To protect against bad mouthing attacks, when evidence revocation occurs, it is proposed that the policy decisions require redundant pieces of evidence from independent sources to proceed to the evaluation.

3.1.3.3 Distributed Certification Authority Models

The use of secret sharing to distribute the CA functionality among a set of nodes in ad hoc networks was first proposed by Zhou and Haas \cite{Zhou1999}. Their Distributed Public Key Model takes advantage of redundancies in the network topology to achieve availability of the CA service, that is provided by an aggregation of nodes that trust is distributed to. The model uses threshold cryptography \cite{Shamir1979} to distribute the private key of the CA over a number of network nodes $n$, that share the ability to perform cryptographic operations. The scheme allows for any $t + 1$ out of $n$ nodes to combine their partial keys to collaboratively generate the secret key of the service and sign certificates, whereas this would be unfeasible for any $t$ nodes.

For an adversary to acquire the secret key, at least $t + 1$ of the designated nodes must be compromised. In order to tolerate mobile adversaries, the authors make their threshold cryptography scheme proactive by using share refreshing. This enables the designated nodes to derive new partial keys from the old ones in collaboration, without having service secret key disclosed to any of them.

A number of models based on secret sharing have been proposed since. A review of the current state of the art can be found in reference \cite{Oechslin2008}. One of the most notable works on the field of ad hoc networks is the Mobile Certificate Authority model (MOCA), presented by Yi and Kravets \cite{Yi2003}. MOCA similarly uses secret sharing mechanisms to distribute trust to an aggregation of nodes that can collaboratively provide certification authority services. Provided that heterogeneity is expected to exist among ad hoc network nodes, the nodes that are assigned with CA functionality, called MOCAs, are selected according to criteria like computational power, physical security or risk of compromise. The model includes a communication protocol that client nodes are equipped with in order to correspond with MOCAs for certification services, by contacting at least $t + 1$ MOCAs and receiving at least $t + 1$ replies.

The model deals with trust revocation through certificate revocation lists, stored at each node, at the MOCAs, or at a set of specially designated nodes. For a certificate to be revoked, each MOCA signs a revocation certificate with its partial key and broadcasts.
3.1. TRUST MANAGEMENT

3.1.4 Behaviour-Based Trust Establishment

The behaviour-based trust models view trust as the level of positive cooperation between neighbouring nodes in a network. The basic aim of the behaviour-based models proposed for ad hoc and sensor networks is to identify the most cooperative nodes and to isolate the ones that either act maliciously because they have been compromised, or selfishly in order to preserve resources, by assigning and recommending the appropriate levels of trust. The approaches discussed in this section are equivalent to the soft trust approaches of [LV07], where trust relationships are derived from localized and external observations of system entity behaviour, and the reputation-based approaches of [BDOS05], where trust decisions are based on reputation measures gathered and shared by a distributed community.

Trust in behaviour-based approaches is evaluated both independently by each node based on observations, previous interactions and network traffic monitoring metrics, and cooperatively through sharing recommendations and spreading reputation. The basic elements of a behaviour-based trust management system are shown in Figure 3.3 [SHL08]. The trust record stores the trust values and related information for each of the established trust relationships. At the core of the system are the direct and indirect trust calculation elements.

Direct trust is the result of the independent behaviour evaluation, and is based on the direct experience the trustor node may have on the trustee node. The direct trust value is evaluated using mechanisms similar to the watchdog [MGLB00], which was originally proposed for identifying misbehaving nodes in ad hoc networks through monitoring the neighbouring node’s transmissions. The evidence collection mechanisms are usually placed below the application layer, in order to evaluate routing behaviours and information integrity. In the context of WSNs, even the raw data communicated
can be evaluated for consistency among neighbouring nodes [GS04]. Yan et al. [YZV03] proposed the use of a trust evaluation matrix for each network node to store the knowledge derived through network traffic monitoring, and a linear function to compute the trust value based on the evaluation parameters in the trust matrix and pre-defined factor rates for weighting. Pirzada and McDonald [PM04] use independent trust agents that reside on network nodes, to gather network traffic information in passive mode by applying appropriate taps at different protocol layers. The information gathered from these events is classified into trust categories, so that the situational trust $T_S(i,j,x)$ for node $j$ can be computed using the information of trust category $x$. Moreover, weights are assigned according to the utility and importance of each trust category. The direct trust is thus computed as the trust that the trustor node $i$ assigns to the trustee node $j$ based upon all previous transactions in all situations, according to their significance.

*Indirect trust* is derived using recommendations from other nodes, which usually are their trust values for the target node. Selection criteria may be applied for the neighbouring nodes that provide the recommendations. The indirect trust derivation process may include weighting the recommendations of other nodes using their referral trust values, or providing confidence values along with the recommendations. The result of the recommendations exchange for computing indirect trust is that node reputation is spread through the network, enabling the formation of a connected trust graph. The most important factor that could hinder this process is node selfishness and unwillingness to spread reputation information. Including node cooperation on reputation spreading for the calculation of direct trust is one of the countermeasures.

The methods that are specified in most behaviour-based trust models in order to evaluate the trust value of the trustor node $i$ for the trustee node $j$ are:

- A method for calculating and updating the direct trust value, based on previous interactions and network traffic monitoring metrics.
- A method for calculating the indirect trust value based on recommendations from selected neighbouring nodes.
- A method for calculating the final trust decision $T(i,j)$ through balancing the relationship between direct and indirect trust. The result of this calculation is compared against a trust threshold to reach the final decision on node cooperation. Models like [YZV03] also include context and action specific metrics for computing $T$.

A number of approaches have been proposed for modelling and evaluating behaviour-based trust, and the methods for calculating direct and indirect trust vary. Table 3.2 summarises the trust evaluation parameters of the main behaviour-based models that have been proposed for ad hoc and sensor networks, and are described in the following sections categorised according to how they model and maintain trust and reputation into linear, Bayesian, belief theory, and other approaches.

### 3.1.4.1 Linear Trust Computation

A trust-domain based security architecture for mobile ad-hoc networks was proposed by Virendra et al. [VJCU05]. It includes a behaviour-based trust evaluation model...
Table 3.2: Trust evaluation parameters of behaviour-based trust models. The parameters are: (NTM)-Network traffic monitoring, (WCE)-Weighted combination on event significance, (WFE)-Freshness as a weight factor for the events, (R)-Recommendations from neighbouring nodes, (RCF)-Explicit confidence factor on recommendations, (WCR)-Weighted combination of recommendations using the referral trust of the recommenders.

<table>
<thead>
<tr>
<th>Trust Model</th>
<th>Direct Trust Evaluation</th>
<th>Indirect Trust Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NTM</td>
<td>WCE</td>
</tr>
<tr>
<td>Linear Approaches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[VJCUL05]</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>[PK07]</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Bayesian Approaches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[BLB03]</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>[GS04]</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>[ZMHT06]</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Belief Theory Approaches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[JGK06]</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>[KB06]</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Other Approaches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[TB06]</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>[SYHL06]</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

that is used both as the basis for key establishment decisions and for secure node grouping. Trust evaluation is based both on direct and on indirect knowledge. For computing direct trust, network monitoring parameters related to traffic volumes and information integrity are listed, and a traffic statistics function is presented but not precisely defined. Four schemes are proposed for combining indirect trust information, the most sophisticated of which is the double weighted approach:

\[
T_{\text{indirect}}(i, j) = \frac{\sum_{k \in O} T(k, j) / \sum_{m \in O} T(m, j) \ast T(i, k)}{\sum_{k \in O} T(i, k)}
\] (3.1)

The set \(O\) appearing in the equation is the set of nodes in the range of both \(i\) and \(j\), that \(i\) trusts above a certain threshold. Function \(T(i, j)\) for calculating the final trust decision balances the relationship between direct and indirect trust through utilising weighting factors.

The method proposed by Probst and Kasera enables computing both statistical trust and a confidence interval around the trust based on direct and indirect experiences of sensor node behaviour [PK07]. The experiences that are accumulated are of four types: the correctness of the nearby sensor readings, the experience generation accuracy, the observed data propagation accuracy of the data forwarding nodes, and the observed accuracy of data aggregation of the aggregator nodes. Recognising that these types of experiences refer to different contexts, they are not combined in trust evaluation. Instead, for the evaluation of the direct trust and the confidence interval for a specific context, only the corresponding experience records are utilised. Each experience record includes a rating \(x_i\) of trustworthiness that the observer assigns the node being observed for the particular experience \(i\), and a context specific weight \(w^*_i\) indicating the amount
CHAPTER 3. RELATED WORK

of observation for generating the experience record. For trust evaluation, a weight $W_i$ is calculated for each experience record as $W_i = w_c w_t$, where $w_t$ is based on the age of the experience record. The evaluation of the trust and confidence intervals using the experience records is performed by calculating the weighted mean $\bar{\pi}_w$ of all ratings $x_i$:

$$\bar{\pi}_w = \sum \left( \frac{W_i}{\sum W_i} x_i \right)$$  \hfill (3.2)

Using this weighted mean, the unweighted variance $\sigma^2$ and, consecutively, the weighted variance $\sigma_w^2$ are calculated as:

$$\sigma^2 = \frac{\sum (x_i - \bar{x}_w)^2}{n-1}, \sigma_w^2 = \frac{\sigma^2 \sum W_i^2}{(\sum W_i)^2}$$  \hfill (3.3)

The confidence interval about the weighted mean is then calculated using the weighted variance using the Student’s t-distribution. The mechanism enables experiences to be shared between neighbouring nodes that are trusted in the context of experience generation accuracy. The weight $W_i$ of the experiences from third parties is discounted according to the trust in the third party in the context of accurately generating experiences.

3.1.4.2 Bayesian Approaches for Trust Evaluation

The Bayesian approaches maintain trust and reputation as probabilistic distributions. Buchegger and Le Boudec were the first to use Bayesian statistics for representing node reputation and trust evolution [BLB03]. The aim of their CONFIDANT (Cooperation Of Nodes, Fairness In Dynamic Ad-hoc NeTworks) [BLB02] scheme was to detect and isolate malicious nodes by means of observation or reports about attacks, and to allow nodes to route around misbehaving nodes. Their trust model uses the data obtained by direct and indirect observations to make an estimation of the probability of a node to act maliciously, represented by $\theta$. The Bayes Theorem is used to calculate the probability of the random variable given an observation.

$$P(B_i|A) = \frac{P(A|B_i)P(B_i)}{\sum_{i=1}^n P(A|B_i)P(B_i)}$$  \hfill (3.4)

The likelihood $P(x)$ is assumed as $P(x) = \theta^n (1 - \theta)^{1-n}$. Using a prior distribution to represent the initial belief, with each direct or indirect observation, the information available can be updated to reflect the added knowledge and to increase the certainty for the a belief. The Beta probability density function, and specifically $Beta(1,1)$, is used to reflect the prior belief.

$$f(\theta) = Beta(\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta^{\alpha-1}(1-\theta)^{\beta-1}$$  \hfill (3.5)

$$\Gamma(x + 1) = x\Gamma(x), \Gamma(1) = 1$$  \hfill (3.6)

The Beta function is also used to calculate the posterior distribution and update each observation. Using $s$ to represent the number of successes and $f$ the number of failures, $Beta(\alpha, \beta)' = Beta(\alpha', \beta')$, with $\alpha' = \alpha + s$ and $\beta' = \beta + f$. The input
that is thus required for evaluating and updating the trust towards any node $j$ is the number of positive and negative, direct and indirect, observations for $j$, which are used for the update of the Beta function. The trust value is calculated by taking the probability expectation of the Beta distribution as $E(Beta(\alpha, \beta)) = \alpha / (\alpha + \beta)$. The authors also include different options for combining the direct and indirect observations and weighting recommendations.

Ganeriwal and Srivastava \cite{GS04,GBS08} also follow a Bayesian approach for trust evaluation. The Reputation-based Framework for Sensor Networks (RFSN) includes a watchdog mechanism for monitoring the behaviour of neighbouring nodes in terms of data forwarding, raw sensing data consistency, and reported data correctness. Each sensor node $i$ maintains reputation for other nodes $j$ in the form of a probabilistic distribution $R_{i,j}$ representing the reputation between the two nodes, and trust is obtained by taking its statistical expectation as $T_{i,j} = E[R_{i,j}]$. Reputation $R_{i,j}$ is built based on the results of the watchdog mechanism (direct reputation), maintained as $(R_{i,j})_D$, in combination with second hand information for deriving the indirect reputation. $R_{i,j}$ is derived as the addition of two probabilistic distributions $(R_{i,j})_D + (R_{i,j})_{ID}$. The following equation is defined for deriving the indirect reputation $(R_{i,j})_{ID}$ by weighting the second-hand information from the neighbouring nodes of $i$, denoted as $N_i$:

\[(R_{i,j})_{ID} = (R_{i,j})_{ID} + \{g(R_{i,k}) * R_{k,j}\} \forall k \in N_i \quad (3.7)\]

RFSN also incorporates exponential averaging when combining reputation information, in order to place more weight on recently obtained information. In order to discourage adversaries from changing identities or creating virtual nodes, the initial reputation of each node is an empty value and has to be gradually built. Moreover, the authors propose the propagation of good reputation information only to protect against defaming attacks, so that trust is revoked only on the basis of direct observations. This solution against defaming attacks had also been applied earlier by the CORE system \cite{MM02}, which required the information collected through indirect reputation to take only positive values.

The notion of confidence in trust relationships is used in the Bayesian theory-based Hermes framework \cite{ZMHT06}, where trust and confidence values are mapped in a trustworthiness composite metric. In order to improve the resilience of trust evaluation against defaming attacks, Hermes extends the collection of direct trust evidence among neighbouring nodes to non-neighbouring nodes through an acknowledgement protocol. It also provides the means for calculating the recommender trustworthiness, and includes this metric in the indirect trust evaluation.

An incentive compatible reputation mechanism was proposed by Liu and Issarny \cite{LI07} to increase robustness against defaming attacks and to provide incentives for sharing honest recommendations. Similarly to the previous Bayesian approaches, the mechanism separates service reputation from recommendation reputation, which are both expressed as Beta distributions. The recommendation reputation is used to classify the other nodes as active/inactive truthtellers, active/inactive liars, and newcomers. The recommendations requests are handled according to the class of the requestor, and essentially according to its honesty in providing recommendations. This strategy is shown to stimulate reputation information sharing and honest recommendation elicitation.
3.1.4.3 Belief Theory Approaches

Belief theory is related to probability theory, with the difference that the sum of probabilities over all possible outcomes does not necessarily add up to 1, and the remaining probability is interpreted as uncertainty [JIB07]. Subjective Logic [Jøs01] is a belief calculus that uses a belief metric for uncertain probabilities called opinion and a set of logical operators for reasoning with uncertain propositions. It was applied for trust network analysis by Jøsang et al. [JGK06] to explicitly represent uncertainty and to allow trust measures to be expressed as opinions. An opinion expressing A’s belief in the truth of proposition \( x \) is denoted by \( \omega^A \) and expressed as a 4-tuple \((b, d, u, a)\). Elements \( b, d \) and \( u \) represent belief, disbelief and uncertainty respectively, where \( b, d, u \in [0, 1] \) and \( b + d + u = 1 \). \( a \) is called base rate, and is used to determine how uncertainty contributes to the opinion’s probability expectation value as \( E(\omega^A) = b + au \). For an unknown party, where uncertainty \( u = 1 \), the base rate determines the a priori trust that would be assigned to it. For the calculation of the opinion elements \( b_x, d_x \) and \( u_x \) after the observation of \( r \) positive and \( s \) negative occurrences of \( x \), the authors derive a mapping with Beta probability density functions parameters, and define \( b_x = r/(r + s + 2), d_x = s/(r + s + 2), \) and \( u_x = 2/(r + s + 2) \).

The model applies subjective logic operators for the construction of trust paths and the combination of opinion from third parties. Specifically, it applies the discounting operator to derived trust from transitive paths, and the consensus operator to aggregate trust from parallel paths. These operators are essentially used as the link operator and the aggregation operator discussed in Section 3.1.1. The discounting operator \( \otimes \) is used to compute trust transitivity when an agent \( A \) has referral trust in \( B \), denoted by \( \omega^B \), and \( B \) has functional trust in the truth of proposition \( x \), denoted by \( \omega^x_A \). Trust transitivity causes uncertainty to increase. It is denoted as \( \omega^{A:B}_x = \omega^B_A \otimes \omega^x_A \) and defined as:

\[
\begin{align*}
    b_x^{A:B} &= b_x^B b_x^A \\
    d_x^{A:B} &= b_x^B d_x^A \\
    u_x^{A:B} &= d_x^B + u_x^B + b_x^B u_x^A \\
    a_x^{A:B} &= a_x^B \\
\end{align*}
\]

The consensus operator \( \oplus \) is used to aggregate trust in the presence of more than one opinions or trust paths. If \( \omega^A \) and \( \omega^B \) are the opinions of \( A \) and \( B \) about proposition \( x \), the base rate between these reflects both opinions in an equal way. The aggregation of trust through this operator amplifies belief and disbelief and reduces uncertainty. The operator is equivalent to Bayesian updating of Beta probability density functions, denoted as \( \omega^{A\oplus B}_x = \omega^A_x \oplus \omega^B_x \) and defined as:

\[
\begin{align*}
    b_x^{A\oplus B} &= (b_x^A u_x^B + b_x^B u_x^A)/(u_x^A + u_x^B - u_x^A u_x^B) \\
    d_x^{A\oplus B} &= (d_x^A u_x^B + d_x^B u_x^A)/(u_x^A + u_x^B - u_x^A u_x^B) \\
    u_x^{A\oplus B} &= (u_x^A u_x^B)/(u_x^A + u_x^B - u_x^A u_x^B) \\
    a_x^{A\oplus B} &= a_x^A \\
\end{align*}
\]

An alteration to the subjective logic-based model of Jøsang et al. was proposed
3.1. TRUST MANAGEMENT

by Kane and Browne [KB06]. In their work, they devise a mechanism for formulating and updating opinions starting from an entirely uncertain opinion \( \omega_A^B = (0, 0, 1, a) \) for an unknown node \( B \). They use the information on the positive and the negative interactions with \( B \), and a parameter \( \delta \in [0, 1] \) which determines how much a rating changes after an individual interaction between nodes. They define that when a positive interaction occurs, if uncertainty \( u_B \geq \delta \), then belief \( b_B = b + \delta \) and \( u = u - \delta \). Otherwise, belief \( b = b + \delta \), disbelief \( d = d - (\delta - u) \), and \( u = 0 \), which denote the decrease of disbelief with positive interactions as uncertainty is exhausted. Similar formulas are defined for negative and uncertain interactions, all causing belief and disbelief to increase monotonically when information becomes available and as long as uncertainty exists.

3.1.4.4 Other Approaches for Trust Evaluation

The theory of semirings is used for trust evaluation by the solution proposed by Theodorakopoulos and Baras [TB06], who focus on the evaluation of indirect trust as the combination of opinions from neighbouring nodes, assuming that some mechanism exists for these nodes to assign their opinions based on local observations. The process of indirect trust evaluation is formulated as a shortest path problem on a weighted directed graph, where graph nodes represent network nodes and edges represent trust relations. The edges are weighted with the trust value the issuer node has on the target node and the confidence value it assigns on its opinion, depending on the number of the previous interactions and positive direct evaluations. The theory of semirings is used for formalising two versions of the trust inference problem: finding the trust-confidence value that node \( i \) should assign to node \( j \), based on the trust-confidence values of the intermediate nodes, and finding a sequence of nodes that has the highest aggregate trust value among all trust paths from \( i \) to \( j \). The authors define path and distance semirings for computing the trust distance along trust paths from the issuer to the target, and a computation algorithm that is an extension to Dijkstra’s algorithm.

Markov chains are used in the work of Jiang and Baras [JB06] on trust evaluation on autonomous networks in order to investigate their characteristics when the system is at the steady state. They propose a statistical trust evaluation rule based on local voting, which is further specified as an iterated stochastic rule. The Markov chain interpretation and convergence of the stochastic rule is then used to study the properties of the resulting trust values.

In the information theoretic framework proposed for trust evaluation by Sun et al. [SYHL06] trust is viewed as a measure of uncertainty with its value represented by entropy. Assume that the trust issuer node observed that the target agent performed an action \( k \) times upon the request of performing the action \( N \) times. Denoting the trust value of a relationship as \( T \{ \text{subject : agent, action} \} \) and the probability that the target agent will perform the action successfully from the subject’s point of view as \( p = P \{ \text{subject : agent, action} \} \), they define the entropy-based trust value as:

\[
T \{ \text{subject : agent, action} \} = \begin{cases} 
1 - H(p), & 0.5 \leq p \leq 1 \\
H(p) - 1, & 0 \leq p < 0.5 
\end{cases}
\]

\[
H(p) = -p \log_2(p) - (1 - p) \log_2(1 - p)
\]
\[ p = Pr(V(N + 1) = 1|n(N) = k) = \frac{k+1}{N+2} \]

\( V(i) \) is a random variable equal to 1 if the target agent performs the action successfully at the \( i \)th trial and 0 otherwise. \( n(N) \) denotes the number of actions performed by the target agent out of \( N \) trials and is defined as \( \sum_{i=1}^{N} V(i) \). For trust propagation, the entropy-based trust model distinguishes functional trust \( T\{A:B, action\} \) from referral trust \( R = T\{A:B, makingRecommendation\} \), and uses the product operator as the link operator, so that \( T_{AC} = R_{AB}T_{BC} \). For multipath trust aggregation, it uses maximal ratio combining as:

\[ T\{A:B, action\} = w_1(R_{AB}T_{BC}) + w_2(R_{AD}T_{DC}) \quad (3.8) \]

where \( w_1 = R_{AB}/(R_{AB} + R_{AD}) \) and \( w_2 = R_{AD}/(R_{AB} + R_{AD}) \). The authors also develop four axioms that address the rules for trust propagation, and prove that the proposed model satisfies them. These are that (1) uncertainty is a measure of trust, that (2) concatenation propagation of trust does not increase trust, that (3) multipath propagation of trust does not reduce trust, and that (4) trust based on multiple recommendations from a single source should not be higher than that from independent sources. Section 3.1.1 includes a further discussion on the axioms.

The trust establishment model proposed for ad hoc networks by Raya et al. [RPGH08] uses the Dempster-Shafer theory for evaluating the validity of node data reports with corresponding trust levels. The model extends the traditional notion of trust to data-centric trust, by attributing trustworthiness to node-reported data per se. The proposed mechanism first computes trust in each individual piece of reported data, then combines the data and uses the Dempster-Shafer Theory to infer their validity by a decision component.

### 3.1.5 Hybrid Trust Management Models

The hybrid trust management models accept, combine, or utilise different types of trust evidence, and have properties of both certificate-based and behaviour-based approaches. The differences between these approaches were studied by Bonatti et al. [BDOS05]. They recognise that two different perspectives exist on trust management: in the strong and crisp policy-based approach, trust relies on logical rules, strong security mechanisms and verifiable properties encoded in digital credentials, while in the soft and social reputation-based approach, decisions are based on reputation measures gathered using local experiences and shared by a distributed community. While the two approaches address the same problem, i.e. the establishment of trust among interacting parties in distributed and decentralized systems, they assume different settings, different sources for trust, and target different requirements. The policy based approach was developed within the context of structured organizational environments, using CAs as its main source of trust, to address the requirement for decentralised access control. The reputation-based models were proposed to address the unstructured user community, use the community opinion as their source of trust, to fulfil the requirement for trust evaluation in environments such as peer-to-peer networks and the semantic web, where a large pool of individual user ratings is usually available.

In their work, they argue that an integrated approach would significantly improve trust management systems. They provide real world scenarios to support this argument.
An example setting is where users are interested both in knowing whether a service provider has a certificate from a CA, and in experiences other users had in the past while performing transactions with it. They propose using a trust management language that enables the integration of reputation-based and policy-based approaches, through combining policy-based decisions with numerical-based ones. An example policy for controlling access to credit card information is formed like:

\[
\text{allow}(\text{visaCard}) \leftarrow \text{credential}(\text{member}(\text{Requester}), \text{myCA}), \\
\text{trust}(\text{self, Requester, buying, } X), \\
X \geq 0.8.
\]

This policy would specify that access is granted only to entities that are certified by \text{myCA} and have a good reputation. The proposed solution does not, however, specify how reputation is evaluated and spread within a community. Moreover, it is still a policy-based approach, albeit enriched with reputation predicates.

Lin et al. [LV07] studied the properties of the individual approaches on trust establishment and provided evidence on the benefits that can be achieved if they are integrated. They classified trust relationships to hard and soft according to their scope and the type of trust evidence used to establish them. Hard trust relationships are defined as those that can be derived from underlying cryptography based security mechanisms, such as digital certificates, signatures and cryptographic checksums, while soft trust is based on trust relationships derived from behavioural evidence obtained through localized and external observations, through social control mechanisms such as direct experiences, recommendations or combination of both.

The properties of the trust models of each of the individual approaches on trust establishment were studied in order to identify their advantages and their drawbacks. They found that hard trust models present drawbacks such as lack of learning capability, static and rigid model structure, which makes them difficult to scale in open decentralised network environments, and lack of feedback control due to their inability to process behavioural evidence. Soft trust models were found to suffer from lack of traceability, from the trust saturation problem occurring after a long history of positive experiences, and from the low trust evaluation problem occurring when trust evaluation stays low even with a relatively high number of experiences in the evidence space. An interesting result from this analysis is that the drawbacks of each individual approach can be compensated by the advantages of the opposing approach. For example, traceability may be lacking from soft trust models, but it is inherent in the hard trust models.

Having found that a hybrid approach can solve the problems introduced by the individual approaches, they propose a hybrid trust model for enhancing security in distributed applications. They specify the combinations that can be made from hard and soft trust of different scopes, resulting to single class and multi class trust, and define hybrid trust as a composite trust relationship formed by combining hard trust and soft trust. However, they do not specify any metrics and processes for obtaining and combining the contributions from hard and soft trust.

In the field of open multi-agent systems, the FIRE trust model proposed by Huynh
et al. [HJS06] integrates multiple types of trust to produce an assessment of an agent’s likely performance. Their motivation for enabling multiple types of trust to be used are the inherent uncertainties in open multi-agent systems and the fact that, in various circumstances, not all individual types of trust evidence will be readily available. The four types of trust that are integrated in the model are (1) the interaction trust, which results from direct past experiences with the target agent and is equivalent to the direct trust of the behaviour-based models, (2) the witness reputation, which results from the reports of witnesses about the target agent’s behaviour and is equivalent to the indirect trust of the behaviour-based models, (3) the role-based trust, which results from the local evaluation of context-based rules, of different levels of influence, representing pre-known, role-based, quantified trust relationships between agents, and (4) the certified reputation, which is evaluated using the references of other agents about the target’s behaviour that the target holds and provides as evidence.

The last type of trust is introduced to address the lack of trust evidence of the other types when that target agent is unknown, when witnesses about its behaviour can not be found or reached, and when no role-base relationships have been defined. In such cases, the target agent can provide certified evidence about its past performance in the form of references. The model assumes that some form of security mechanism, such as a public-key infrastructure, is employed to ensure that the references are not tampered with by the target. The weighted mean method is used to combine the different types of trust in a composite trust value, where the weights are set by the end users to reflect the importance of each component in a particular application.

3.1.6 Analysis and Discussion

The separate discussion of the approaches that exist on trust establishment highlighted the different perspectives of trust that they adopt, and the diversity in their scope, purpose, admissible types of evidence, and methods for modelling and evaluating trust. Each approach does, however, has its own limitations and drawbacks in terms of the characteristics and properties of trust that it supports, its resource requirements, and its applicability on WSNs:

Certificate-based solutions The inherent limitations of the solutions that rely on strong security mechanisms is their lack of learning capability and the lack of feedback control, due to their inability to process behavioural evidence at the runtime, which makes them vulnerable to malicious behaviour [LV07]. The web-of-trust distributed models are considered to offer more flexibility than the hierarchical models, but may not be suitable for applications where high degrees of accountability and security are required [Dav04]. The main reasons are that they are less structured and more prone to attacks by malicious agents, since they do not have any central management points to enforce strict policies on trust assessment. The distributed certification authority models are considered more robust, but are the ones that impose the greater deployment complexity and have the higher communication requirements per evaluation request. Moreover, the models following both approaches require the cooperation of ad hoc network nodes that may behave selfishly to preserve resources [HBC01, Dav04]. Finally, we consider that both approaches are too resource consuming to be practically applied in WSNs. The energy consumption for applying asymmetric cryptography
on sensor nodes has been measured \cite{WGE+05} and it is considered to be expensive for sensor nodes \cite{SP04, AEAQ05}, especially due to the communication overheads imposed. Extensive asymmetric cryptography operations are however required both in the web-of-trust approaches for the evaluation of certificate chains, and in the distributed CA approaches due to the use of threshold cryptography.

**Behaviour-based solutions** The resilience of the behaviour-based solutions on the observed behaviour and the reputation of the nodes for trust evaluation limits their capability of utilising the pre-deployment knowledge that may exist on the trust associations between the nodes. The solutions of this category do not support role-based trust, which the hard trust models support through certificates, rules and policies. The lack of traceability is also an inherent drawback of soft trust models, since they do not base trust evaluation on strong security mechanisms \cite{LV07}. Another drawback is their resource requirements. For the calculation of the direct trust values, these models utilize techniques similar to the ones of intrusion detection systems, which are considered expensive for the sensor nodes in terms of memory, energy and communications requirements \cite{PSW04}. As discussed in section 2.1 dynamic power management is crucial for prolonging the lifetime of the sensor nodes, which enter sleep state when not needed and are woken up when necessary. The behaviour-based solutions, however, require the radio on each node to be continuously enabled in order to monitor its neighbouring nodes’ behaviour, and to store and continuously update the information required for trust evaluation as interactions occur. Because of this, although behaviour-based trust evaluation models appear to be less complex computationally than the certificate-based ones, they are in practice more energy consuming. Since for the sensor nodes monitoring the network traffic is resource consuming in terms of computation, memory and energy, and we consider that, practically, the applicability of these models on WSNs is limited.

**Hybrid solutions** The hybrid models that were discussed in the previous section are all proposed for distributed systems or multiagent systems, while no hybrid model targeted to ad hoc or sensor networks was found in the related bibliography. Due to their emphasis on these environments, these models do not take into account the special requirements and limitations of sensor networks, like their resource limitations and the varying connectivity, that make it challenging to obtain the different types of trust evidence that they require. For example, FIRE \cite{HJS06} uses four different types of available evidence to be accumulated for trust to be established, which require all local observations, communications with the target node for obtaining its references, and communications with other nodes for gathering their recommendations. At the same time, it requires the availability of some security mechanism, such as a public-key infrastructure, for ensuring the integrity of the references. While these prerequisites could be established in a traditional multi-agent environment and could possibly lead to a highly flexible and robust trust management system, they are too resource consuming for WSN scenarios both in communication and in computation requirements. At the same time, other hybrid solutions \cite{BDOS05, LV07} do not address the issue of how the required evidence will be obtained in environments as dynamic and challenging as ad hoc and sensor networks, and do not specify any metrics and processes for combining
Table 3.3: Comparison of the supported trust characteristics of the trust establishment models for ad hoc and sensor networks. The values are: (C)-Controlled, (U)-Uncontrolled, (N)-Not supported or not defined, (G)-Gradual, (I)-Immediate.

<table>
<thead>
<tr>
<th>Trust Model</th>
<th>Certificate-Based Trust Models</th>
<th>Behaviour-Based Trust Models</th>
<th>Hybrid Trust Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>[VOT01]</td>
<td>U N C I</td>
<td>[VJCU05]</td>
<td>U C U G</td>
</tr>
<tr>
<td>[HBC01]</td>
<td>U U N</td>
<td>[BLB03]</td>
<td>U C C G</td>
</tr>
<tr>
<td>[EGB02]</td>
<td>C C C I</td>
<td>[GS04]</td>
<td>U C C G</td>
</tr>
<tr>
<td>[ZH99]</td>
<td>U N N</td>
<td>[ZMHT06]</td>
<td>C C C G</td>
</tr>
<tr>
<td>[YK03]</td>
<td>U N C I</td>
<td>[JGK06]</td>
<td>C C C G</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[KB06]</td>
<td>C C C G</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[TB06]</td>
<td>C C U G</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[SYHL06]</td>
<td>C C C G</td>
</tr>
</tbody>
</table>

Apart from the supported trust characteristics, the limitations and the drawbacks that are specific to each category, there exist some issues that are common for behaviour-based, certificate-based, and hybrid models. The common trust characteristics include support for uncertain evidence, transitivity of trust and trust revocation. Table 3.3 presents a comparison of the supported common trust characteristics of the models that have been discussed. The use of uncertain evidence is characterised as controlled for models that support assignment of confidence values to the evidence supplied for trust evaluation, including recommendations from other nodes. Transitivity of trust, if supported, is considered controlled if trust values from third parties are weighted according to the trust relationship the requester has with the third party, before being used for trust evaluation. For models that support trust revocation, it is considered controlled either if trust is revoked only by trusted third parties, or if some mechanism exists to protect from defaming attacks. Moreover, trust revocation is characterised gradual if trust is not revoked explicitly, but as the result of bad reputation spread gradually due to node misbehaviour.

Concerning the representation of trust, none of the behaviour-based and hybrid models discussed uses discrete values, since it is considered too restrictive. Instead, they represent trust in a continuous range and compare its value with a trust threshold.
to decide on node cooperation. Certificate-based models base the decision on node cooperation on the provision of a trusted certificate, i.e. a certificate that either is valid since it is signed by a (distributed or centralised) trusted third party, or a trusted certificate chain that includes it can be formulated.

The issue of tackling node selfishness, that is especially important for models that entail node cooperation, either for reputation spreading or for providing CA functionality, is not sufficiently addressed in the models studied. The only solutions that addressed this issue are the one proposed by Liu and Issarny [LI07] and the model proposed by Weimerskirch and Thonet [WT02], where incentives and punishment mechanisms are specified for recommending nodes.

None of the behaviour-based models supports pre-established and stable trust relationships, since they do not include any bias with respect to the identity of the node under evaluation. From the certificate-based models, pre-established trust could be supported by [EGB02] through introducing identity related bias in the trust metrics and policies of the nodes. For the model introduced by Hubaux et al. [HBC01], this requirement could be satisfied through the appropriate selection of the locally stored subsets of the trust graph, if the certificate repositories of nodes were configured to include the certificates of trusted nodes that each issuer should maintain direct and stable trust relationships with.

It is our belief, that the behaviour-based and the certificate-based models that have been proposed are better targeted at ad hoc rather than at sensor networks. The behaviour-based models do not exploit the pre-deployment knowledge that will usually be available in WSN deployments, and they do not support pre-established trust relationships. Moreover, the computational complexity of the certificate-based and the energy requirements of the behaviour-based trust evaluation models raise concerns related to their applicability on resource constrained sensor nodes. The hybrid solutions, being targeted to other types of environments, do not take into account the special requirements and limitations of sensor networks, and do not specify concrete metrics and processes for combining the contributions from hard and soft trust. Overall, given the high computational costs of certificate validations, the involvement of third parties for the provision of recommendations, and the energy costs of behaviour monitoring, we believe that a hybrid approach, which is targeted to WSNs, can leverage the drawbacks associated with the individual approaches.

### 3.2 Privacy Protection

Privacy issues of WSNs can be addressed at multiple levels of the network stack, at different points of the information flow, and with different assumptions on the network entities that are considered trusted. The requirements for privacy preserving WSNs include confidentiality of the sensed and the aggregated data, protection of the communications context, privacy sensitive information disclosure, and privacy sensitive information gathering. The first of these requirements applies to any privacy protection solution, and all schemes that have been proposed for WSNs assume the use of data encryption mechanisms. However, since data encryption is more a standard security issue than a privacy protection issue, it is separately discussed in Section 3.3.

A number of privacy protection solutions have been proposed for traditional wired
networks and the internet, and there exist various approaches, especially for the protection of the communications context. However, the resource constraints of the sensor nodes make it challenging to apply these approaches to WSNs, which has inspired research on the design of lightweight and flexible privacy protection solutions. After exploring the traditional approaches for privacy protection, this section provides a review of the solutions that have been proposed for sensor and ad hoc networks, categorised according to the privacy requirements in the design dimension of Section 2.3.2, since, once these are addressed, the application and service level privacy requirements for support for anonymous data and for location privacy are fulfilled as a result.

3.2.1 Privacy Protection Approaches

There exist significant differences between the privacy requirements identified for ubiquitous computing and sensor networking, and the privacy requirements for the case of traditional wired networks, the internet, mail and telephony technologies. The most important privacy risks in online transactions are these related to the context of the communications, and specifically to the anonymity of the clients and to the un-linkability of the requests. The mechanisms that have been proposed for enhancing web privacy [Gri04] focus on protecting the contextual aspects of communications, and on enabling informed and controlled online interactions through the enforcement of privacy policies. Although most approaches proposed for traditional networks are considered inapplicable in the case of WSNs for the reasons that were explained in Section 2.3.2, many privacy schemes and mechanisms for WSNs have borrowed aspects or are designed as lightweight versions of these.

The World-Wide-Web Consortium’s Platform for Privacy Preferences Project (P3P) provides a framework for enabling users to apply preferences over privacy policies of web sites and online services, that may describe why and what data is being collected, by and for which entity. P3P allows the encoding of privacy policies into machine-readable XML, and enables automated processing and decision making based on personal privacy preferences expressed using a machine-readable preference language such as APPEL. It can be considered as a protocol for exchanging structured data: it includes specification of both syntax and semantics for describing privacy policies and preferences, and of the mechanisms for users agents to get access to the announced privacy policies. It is however more of a privacy enabler than a privacy enforcer, since it does not enforce privacy or anonymity through technology, but relies on the trustworthiness of organisations to comply with the privacy policies they announce. It thus acts as complementary to other web anonymity tools and protocols.

Approaches that address privacy and anonymity issues for the contextual aspects of communications either interpose an intermediary between the sender and the receiver to provide sender anonymity, or use a network of intermediate nodes, which can protect both the server and the receiver identities from eavesdroppers. An example of the first category is the Anonymizer web anonymity tool, where a web proxy or a group of proxies act as intermediaries, receive the web requests from users and filter the client identification information from the request headers before forwarding to web servers. The identities of the clients are not revealed to the web servers, but only to the Anonymizer service, which is the only component that needs to be trusted. Although Anonymizer is effective in hiding the user identity from the web server, this
3.2. PRIVACY PROTECTION

does not apply for eavesdroppers that may monitor the traffic between the user machine and the Anonymizer server.

The first approach in the second category introduced the concept of MIXes \cite{Cha81} to address the anonymity and untraceability issues for email communications in the presence of eavesdroppers performing traffic analysis. The basic idea of this approach is that each message is sent over a series of independent servers, called mixes. Each mix collects a number of messages from multiple sources, performs cryptographic transformations on them (including encryption and padding) to alter their appearance, and changes their flow by delaying and reordering them before forwarding them to the next destination, so that eavesdroppers cannot link incoming messages to outgoing messages. A number of mixes are usually combined, in order to increase the strength of the anonymity offered by the mix system. ISDN-MIXes \cite{PPW91} were proposed as an extension to the basic mix approach, where the use of mixes is combined with dummy traffic and with broadcasting of the encrypted incoming messages.

According to how the flow of messages is altered by the mix system, the approaches can be classified into two categories: the pool mixes and the continuous mixes \cite{DP04}. Chaum’s mixes belong to the first category, since they place the messages in a pool and flush them in random order when a flushing condition is fulfilled. Depending on the flushing condition, the pool mixes may be time-based, if they send messages every fixed internal time, or threshold-based, if flushing occurs when a certain amount of messages has been collected. In the continuous mixes, the first of which was Kesdogan’s Stop-and-Go-MIXes, the delay is chosen by the users from an exponential distribution and added to the headers of the messages, so that the mix knows when to forward them \cite{KEB98}. The mixes send messages continuously, every time the delay period of a message expires. As a result, the messages are reordered by the randomness of the delay distribution. The delay does not depend on the incoming traffic, so the strength of the anonymity depends on the number of the active users.

The onion routing scheme \cite{RSG98} is also a mix-based approach, where during an initialisation phase the sender determines the message route path to the destination through a series of onion routers. Encrypted routing information is repeatedly added to the payload, so that the message finally transmitted is an onion consisting of several layers of encryption that are stripped off by the onion routers as it traverses the path. Because of the onion structure, each router in the path can only determine the previous and next hop, so the identities of the communicating parties are protected. Onion routers also act as mixes, to make the path untraceable not only by intermediate routers but also by eavesdroppers.

An alternative approach to protecting anonymity is the Crowds protocol \cite{RR98}, which is based on the concept that any member of a crowd can remain anonymous if his actions are indistinguishable from the actions of other members of the crowd. Like the other approaches, Crowds uses a set of intermediaries, called jondos, logically positioned between the sender and the receiver. The routing path is not determined by the sender, as in Onion Routing, but randomly chosen at each hop that the message traverses. Moreover, the message remains the same along all hops of the path, so that no jondo can distinguish whether the previous hop in the path was the sender or a forwarding node. Once a path is established, it is used for all communications between the source and destination, with each jondo communicating with its predecessor or successor on the path. The Hordes protocol \cite{SL00} similarly uses multiple jondos to
anonymously route a message toward the receiver. Instead of using the reverse path of the request, Hordes routes the replies back to the sender through multicasting. Except from ensuring sender anonymity, this allows the use of the shortest reverse path, thus reducing the communication latency.

An evaluation of these anonymisation approaches with respect to their architecture, operational principles and vulnerabilities is presented in reference [Gri04]. Overall, it is considered that the better the level of anonymity protection, the greater the overhead latencies. Crowds is considered less expensive than Onion Routing as it does not require complex cryptographic operations, such as multi-layer encryption. Hordes incurs lower overheads despite the fact that it uses the same mechanisms as Crowds, since Hordes’ members do not perform any complex tasks during the backward routing procedure. The anonymity properties of these approaches in the presence of compromised nodes have been studied in reference [GFBZ04]. It was found that the probability that the true identity of a sender is discovered might not always decrease as the length of communication path increases, and that the complexity of the path topology does not significantly affect this probability.

Although most of these privacy protection approaches are considered unsuitable for WSNs because of the characteristics and constraints discussed in Section 2.3.2 they are not entirely rejected. Similar architectures and mechanisms have been proposed for WSNs that introduce modifications and adjustments to the basic schemes in order to be more lightweight and less dependent on fixed infrastructures. For example, network identity privacy approaches borrow aspects from schemes like Crowds, while being designed to be more targeted for ad hoc and sensor networks. Privacy policy enforcement schemes use modified versions of P3P policies, while adjusting the privacy negotiation protocols to fit the decentralised nature of WSNs. As discussed in the next sections, many basic aspects for protecting privacy are borrowed from traditional networks and adjusted to fit the case of WSNs.

### 3.2.2 Protection of the Communication Context

The messages communicated in a WSN, independently of their content or of the encryption scheme that is being used, should not allow induction of information related to their context by adversaries that can overhear them. Techniques like message timestamping, padding, using serial numbers, or frequent key redistribution can be used so that the communicated cipher texts do not reveal information through their similarity or size. Examples of other contextual information surrounding sensor nodes that could be derived are the location or the time of measurements. In many scenarios, this information is considered sensitive and needs to be protected. In remote monitoring applications where sensing devices are used to track assets of significant value, it would be necessary to avoid the disclosure of their locations to malicious entities. In military applications where wearable sensors are being used in order to securely monitor the status of soldiers, it would be crucial to avoid the disclosure of their movements to the enemy that might monitor the wireless communications.

However, protecting the traffic patterns within the network, which include the network identities, the time and the relative locations of the sender of data packets is not trivial, as it requires interference with the routing protocols. The solutions that are
3.2. PRIVACY PROTECTION

Presented in this section aim to obscure the communication traffic patterns from communication traffic observers through routing strategies that protect the source location, the time the transmitted data was captured, and the network identities.

3.2.2.1 Source Location Privacy

In the general case of large scale WSNs, messages will follow multihop routes from the source node to the destination. The shared wireless medium makes it feasible for traffic observers to identify the origins of radio transmissions within their range through the use of localisation techniques. If the network traffic for a given time period can be correlated to distinguishable sources, a mobile observer can perform hop by hop traceback to the source node of each communication. The worst case scenario for preserving source location privacy is when there is traffic only from a single source in a network, like in the Panda-Hunter problem that is presented in the following paragraphs.

The Panda-Hunter problem

The Panda-Hunter problem [OZT04] pertains to how to enable monitoring of pandas through detection sensor nodes that have been spread over a sensing field, without exposing the location of the pandas to hunters. For simplicity, it is assumed that there is a single sink node for collecting the data and a single panda in the sensing field, that is detected by a single stationary source node each time. When the presence of a panda is detected, the corresponding source node $P$ will start reporting data periodically to the sink node $S$ through multihop routes. It is assumed that the messages are encrypted.

The adversary in this scenario is the hunter $H$, who tries to capture the panda by locating the node that reports the panda’s presence by sending messages to the sink. The hunter starts at the location of the sink and is constantly in a receiving mode. Devices like antenna and spectrum analyzers allow the hunter to observe the messages and identify their immediate senders. By moving to the immediate sender node each time he receives a periodic message and waiting until the next message is received, the hunter makes consequent movements towards the originator node, which he will eventually traceback and capture the panda.

For each experiment in the Panda-Hunter problem, the initial location of the panda is random and the panda is not mobile. If the hunter reaches a specified hop distance from the panda within a threshold amount of time, the panda is considered captured. The primary goal for any message routing strategy that achieves source location privacy is therefore delivering the messages to the sink, while obscuring the location of the source node for a safety period, calculated as the number of periodic messages sent by the source before the hunter localises it. The likelihood of the panda being captured within the safety period depends highly on the traceability of the information leaked by the message routing strategy of the WSN.

The effects of the routing strategy

In order to examine the effects of the message routing strategy on the source location privacy protection, two popular classes of routing protocols for WSNs are considered [OZT04, KZTO05]: the class of flooding protocols, and the class of single path routing protocols.
In the simplest case of flooding, each message is broadcasted from the source node to its neighbours, who in turn rebroadcast until it eventually reaches the sink node having followed a set of different paths. Provided that the hunter’s initial position is the sink node, the first message that will reach him has most likely followed the shortest path from the source to the sink. The hunter will be able to traceback the shortest path by moving each time to the last forwarding node and waiting for the next first message that arrives. Simple flooding thus offers the least possible privacy protection, since the safety period is equal to the shortest path length.

The privacy protection performance of the single path routing protocols is similar. Irrespective of how the path from the source to the sink is selected by each protocol, since only the nodes that are on the selected path participate in message forwarding, the hunter can traceback the single message that is observed when located in each node in the path. The safety period is thus equal to the selected path length.

A strategy that can offer increased privacy protection performance is flooding while using fake message sources. In order for these to be indistinguishable from the real source, the fake messages that are produced should also be encrypted and of the same length as the real messages. For this technique to be effective in misleading the hunter, it was found [KZTO05] that the fake sources should be persistent throughout the experiment, they should generate messages as frequently as the real source, and they should have a distance from the sink similar to that of the real source. Even by using fake sources to produce fake message paths, however, a persistent hunter will eventually select to follow the message path to the real source. At the same time, this strategy is too costly in terms of communication overheads and of energy consumption.

The phantom routing strategy

The poor privacy protection performance of the routing protocols discussed in the previous section is attributed to the fact that they allow the use of stable message paths that lead the hunter from the sink to the source node. The phantom routing strategy [OZT04] that was introduced for WSNs aims to provide source location privacy through directing the periodic messages from the source node towards different paths in the network. This prohibits the hunter from receiving a stable stream of messages that would enable backtracing the source. Instead of that, by the received messages the hunter is led towards phantom sources.

The phantom routing strategy, depicted in Figure 3.4, consists of two subsequent phases for every transmitted message: a pure or directed random walk for a given number of hops that directs the message to a phantom source $N_i$ away from the original source $P$, and a message flooding phase that delivers the message to the sink $S$. An alternative that has been proposed in reference [KZTO05] for the second phase is the use of single path routing instead of flooding. As long as the random walk of the first phase leads to a different phantom source for every message, both approaches are equally effective in increasing the safety period. If the hunter $H$ detects a message forwarded by node $N_i$ and moves to that node to get closer to the source, the next message is unlikely to follow the same random path, thus making the hunter’s previous move worthless. A further advantage that the random walk offers is that the safety period improves as the network size increases, as the paths followed by subsequent messages, and consequently the phantom sources, become more diverse.

The diversity of the paths is not, however, the only issue that the random walk
3.2. PRIVACY PROTECTION

implementation needs to ensure. The main purpose of this phase is to send each message to a phantom source that is far from the original source. In a pure random walk, if the network is uniformly deployed, the message will likely loop around the source node for the required number of hops. A solution to this problem is to use directed walk for the first phase by dividing the neighbouring nodes of the source into two sets, for example north and south as in Figure 3.4, and having the source node randomly pick one partition for forwarding. All nodes in the path, starting from the source node, will choose to forward the message to a random neighbour from the partition initially selected. The partitioning of the network nodes can either be sector-based [OZT04], depending on their relative locations, or hop-based [KZTO05], depending on their distance from the sink in hop counts.

The self-adjusting directed random walk approach [Zha06] was proposed in order to overcome a serious problem that was observed in the two previous approaches: Their performance would drop if the source was located in certain areas of the network where there were not enough nodes of the initially selected partition to perform the given number of hops in a truly random way. In Figure 3.4, where the nodes were divided according to their vertical location in relation to the source, node $N_j$ would be used as a phantom source for half of the messages, i.e., every time the north partition was selected. In the self-adjusting directed random walk approach, the nodes are divided into four partitions, two for each dimension, so that each node is included in one set for each dimension. The initially selected partition can then be changed by the intermediate nodes if it is found that the message can not be further forwarded to the given set for each dimension.

The greedy random walk approach [XSS06] is a two way random walk, performed both from the source and the sink. It was inspired by the observation that if the hunter had gained a good coverage of the network by distributing a number of observation points around the sink, the source location could be approximated because the flooding phase would reveal too much information. In order to avoid this, instead of using flooding to deliver the message to the sink, the sink sets up a random walk which serves as the receptor of the messages. Each message is randomly forwarded from a source until it reaches the receptor, and is then forwarded to the sink through the pre-established path.
Wang et al. formulated the routing strategies for source location privacy as an optimization problem, in terms of the average traceback time and minimal traceback time for the adversary to reach the message source starting from the sink \cite{WSL09}. It was found that the traceback time is related to the number of sensor nodes involved in routing, irrespectively of the routing strategy used. The authors propose the weighted random stride routing strategy to maximise traceback time. The strategy utilises two parameters for the construction of the routing paths from each source node: the forwarding angle, i.e. the angle between the projected forwarding route and the line connecting the source node with the sink, and the stride, i.e. the number of hops associated with the forwarding angle. For each message transmission, the forwarding angle and the stride are randomly picked by the source node, and all nodes selected as the next hops need to follow this angle for forwarding until the end of the stride. When the stride finishes, the last intermediate node selects another angle towards the sink node. The nodes are arranged to pick larger forwarding angles with a higher probability, so as to increase the number of messages distributed to longer paths and to maximise the traceback time.

Phantom routing strategies, independently of the random walk selection techniques that are used, offer increased safety period compared to traditional WSN routing protocols. Source location privacy increases the largest the network is, the more hops the random walk contains and the more mobile the source can be. Moreover, compared to basic flooding, the energy consumption, which mainly depends on the number of the transmitted messages, is not increased. The message latency, however, could be significantly increased depending on the length of the random walk.

**Dummy traffic and message buffering** None of the random walk approaches presented so far can protect the source location against a global eavesdropper, i.e., an adversary who can collect and analyse all network traffic. For this to be achieved, random walk does not suffice: dummy traffic is required to make the original message, and its source node, indistinguishable. Since, however, the use of dummy messages would significantly increase the network traffic and the energy consumption of all sensor nodes, it can not be employed naively.

In the *proxy-based filter scheme* \cite{YSZ08}, dummy traffic is not generated continuously but in specific time intervals. Source nodes postpone the transmission of real event messages until the next interval. To reduce the network traffic incurred from dummy messages, some sensors are selected as proxies using a proxy placement algorithm before network deployment. In each time interval, the proxies filter the dummy traffic messages they receive. In the next time interval, they forward to the base station either the real message or an encrypted dummy message that they generate. To further reduce the network traffic if the size of the network and the number of proxies is large, the *tree-based filtering scheme* is proposed to organise the proxies in a hierarchical structure and to enable dummy traffic to be filtered multiple times on its way to the base station.

This technique, however, results in increased message latency, since the real messages are buffered during each time interval in each of the proxies they traverse. The *fitted probabilistic rate* scheme was proposed to minimise real event notification latency while achieving statistically strong event unobservability \cite{SYZC08}. The time intervals
of dummy traffic generation, instead of being of constant duration, follow an exponential distribution which is controlled; when a real message needs to be transmitted, the duration of the current interval is reduced for an amount of time that does not make the distance from the selected exponential distribution statistically significant.

A different approach for the optimisation of the time intervals and the selection of the nodes that generate dummy traffic is proposed in reference [OLL+08]. Four schemes are proposed, namely the naive, the global, the greedy, and the probabilistic. In the global and greedy approaches the duration of the time intervals is not fixed, but determined by a pseudo-random number generator. Each node needs to know the topology placement and the transmission schedules of its neighbours, so that it can discover the routing path that leads to the shortest delivery latency. The most sophisticated of the approaches, the probabilistic one, is based on the observation that it is not required for all sensor nodes to send dummy messages; if the eavesdropper always hears at least one message at any location and any given time period, it will be difficult for him to identify the delivery path of any event message being sent in the network, since the message observed at any given sensor node may originate from any of its neighbours. In the probabilistic method, if a node is not in the delivery path of any real event message, it will send a dummy message at each time interval with a probability $P$. A process is specified for finding the optimal $P$ such that the network generates the minimum dummy messages required for protecting the source location of real event messages.

Message buffering can be used not only to provide source location privacy, but also temporal privacy, i.e. protection against inference of the creation time of the sensor readings. In the rate-controlled adaptive delaying scheme [KXTZ07], each message is buffered for a random time interval in each intermediate node along its route to the base station. Due to the limited buffer space of the sensor nodes, the buffering strategy is designed to be adaptive: the message delay distribution is adjusted as a function of the incoming traffic rate and the available buffer space. This strategy requires each node to pre-empt buffered messages to accommodate newly arriving ones when the buffer is full, with the pre-empted messages being selected according to their remaining delay time. Similarly to the dummy traffic-based schemes, in the buffering-based scheme the tradeoff is between privacy, resource consumption, and delay, making it unsuitable for time-critical applications.

3.2.2.2 Network Identity Privacy

In the panda hunter problem, the panda can remain hidden within the network only as long as the source node remains out of the network area that the hunter monitors. In the case of an adversary with multiple observation points that has gained enough coverage of the network, hop by hop trace back would not be required, since all messages, including the one from the source node, would be overheard. Even in scenarios where messages are not transmitted by one single node, like in the panda hunter problem, but multiple nodes are transmitting messages at the same time, adversaries would be able to distinguish the ones that originate from a given source. Message sources are distinguishable, since the routing information that is embedded in the packet headers includes permanent identifiers, like network addresses, of the communicating nodes.

Serious privacy breaches can be caused because of unprotected network identities,
especially in the case of wearable sensor nodes, where the network identities correspond to user-carried devices. By observing the network traffic, adversaries can trace the motion patterns of the mobile nodes over the periods of time during which they overhear communications. In a military application where wearable sensors are being used in order to monitor the status of soldiers, their number, relative location and movement could be disclosed to an enemy that has deployed a surveillance network and analyses the headers of the overheard packets.

The aim of network identity privacy approaches is to enable message transmission without disclosing the exact permanent identities of the communicating nodes. While onion routing and mix-based techniques can be used to meet this requirement in traditional networks, they are inapplicable for WSNs for the reasons described in Section 2.3.2. The approaches that can be applied to WSNs meet this requirement by using either node or route pseudonyms instead of permanent identities for routing.

**Node pseudonymity approach** The permanent identities of the communicating nodes are protected by the node pseudonymity approach through the use of mutually verifiable temporary pseudonyms. The pseudonyms are used as common identifiers that replace the identification information in the header of the exchanged packets, so that they appear as unlinkable random identities for anyone except the sender and the receiver. In order to achieve unlinkability between communications, pseudonyms should be frequently updated. For the pseudonym update process to be resilient against pseudonym correlation attacks, mechanisms that utilise the concept of silent periods have been recently proposed [HMYS05]. Silent periods are defined as transition periods between the use of new and old pseudonyms that introduce ambiguity into the spatial and temporal relation between the node’s disappearing and emerging locations and times. These mechanisms, however, require the existence of pseudonymisation authorities to handle the synchronisation.

A distributed approach to node pseudonymity was proposed in reference [WS05]. The toolbox for the use, generation and update of unlinkable pseudonyms $P_i$ for node $i$ consists of symmetric keys $K_{ij}$, random nonces $R_n$ and hash functions $H$. A three-way handshake designed as modification of the ID packet from the original Bluetooth specification can be followed for the pseudonymised communication between nodes $A$ and $B$ that share a symmetric cryptographic key:

\[
\begin{align*}
A \to B & : (R_1|H(P_B|R_1|K_{AB})) \\
B \leftarrow A & : (R_2|H(P_A|R_1|R_2|K_{AB})) \\
A \to B & : (R_3|H(P_B|P_A|R_1|R_2|R_3|K_{AB}))
\end{align*}
\]

Node $A$ that initiates the communication chooses a random nonce $R_1$, computes the hash that is used to protect $B$’s pseudonym and sends the message to node $B$. The next two steps are needed in order for both nodes to verify that they both know their pseudonyms and their shared key. The three random nonces are used in order to ensure the freshness of the messages. The protection that the handshake provides to the pseudonyms is based on the randomness of the nonces, the shared key and the use of the hash function.
Another approach that was proposed for wireless personal area networks [SP06] uses the symmetric key to produce a chain of pseudonyms for each A and B. The initial mutually verifiable pseudonym is computed by using as input to a pseudo-random function the shared key $K_{AB}$ and a random publicly known value. After each communication, the pseudonym is updated by using the old pseudonym as input to the pseudo-random function. Because of these updates, the header of each message between the two nodes will contain a different identifier, that can be linked to the previous ones only by the nodes that know the secret key.

Although the node pseudonymity approaches succeed in protecting the network identities when two nodes communicate, an issue that is insufficiently addressed is how these solutions could scale to multihop transmissions. For single path routing protocols, for example, either the intermediate nodes would need to possess some information on the pseudonym of the destination, or the source node would need to know the pseudonyms of all intermediate hops. For large scale WSNs, synchronisation issues for the update of pseudonyms would also need to be addressed.

**Route pseudonymity approach** The alternative approach for the protection of the permanent identities of communicating nodes against adversaries performing traffic analysis is the route pseudonymity approach. This approach does not provide any identification for the sender and receiver in the packet headers. Instead, it pseudonymises the routes that the messages follow during their multihop transmissions. Data forwarding is performed through unlinkable pseudonyms that are assigned to each hop of the message route.

The problem of developing untraceable routes through the use of route pseudonyms is addressed by the anonymous on-demand routing protocol [KH03] that was proposed for mobile ad hoc networks. The protocol uses an on-demand route discovery process to establish route pseudonyms through randomly naming each transmission hop and recording the mapping between subsequent hops in the forwarding table of each node. At the end of the process, each hop in the route is associated with a random route pseudonym $P_1 \ldots P_n$. The protocol is based on the concept of broadcast with embedded trapdoor information $tr$, that is known only to the receivers of data packets that can open the trapdoor and provide proof $pr$ for it, so that data is anonymously delivered only to them. The route discovery process by a communication source $N_s$ for a receiver $N_r$ is initiated by assembling a request packet and broadcasting it. The request packet contains a unique sequence number $secNum$, the trapdoor for the receiver and a cryptographic onion. The onion is formed by encrypting a tag that denotes the source using a random symmetric key $K_s$. Each intermediate forwarding node $N_1 \ldots N_n$ that receives the request packet, embeds a random nonce $R_1 \ldots R_n$ to the cryptographic onion and encrypts it with its own random symmetric key $K_1 \ldots K_n$:

\[
N_s \rightarrow N_1 : \langle \text{Request, secNum, tr, } K_A(src) \rangle \\
\vdots \\
N_n \rightarrow N_r : \langle \text{Request, secNum, tr, } K_n(R_n, K_{n-1}(R_{n-1}, \ldots, K_A(src))) \rangle
\]

When the request packet is received by $N_r$, the embedded cryptographic onion is a
valid structure to establish an anonymous route towards the source. The receiver opens
the trapdoor to get the proof $pr$ that it embeds in the response packet along with a
randomly selected route pseudonym $P_r$ and the received onion. The response packet
is bounced back to the source. Every intermediate node $N_i$ that can peel off one layer
of the onion using its symmetric key $K_i$, i.e. has participated in the route path of the
request packet, selects a random route pseudonym $P_i$, stores the association $P_i \Rightarrow P_{i+1}$
in its forwarding table, replaces $P_{i+1}$ with $P_i$ in the response and rebroadcasts it:

$$N_r \rightarrow N_n : \langle \text{Response}, P_r, pr, K_n(R_n, K_{n-1}(R_{n-1}, \ldots, K_A(src)) \rangle$$
$$\vdots$$
$$N_1 \rightarrow N_s : \langle \text{Response}, P_1, pr, K_A(src) \rangle$$

Once the source node receives the response and verifies $pr$, it stores the outgoing
route pseudonym $P_1$ in its forwarding table, which can be later used in the headers of
the packets for anonymous data forwarding. When a message is transmitted by the
source, the node in the set of the local receiving nodes that has the stored association
for $P_1$ replaces it with the matched outgoing pseudonym and broadcasts the changed
packet. The process is repeated until the packet reaches the destination.

The Destination-Controlled Anonymous Routing Protocol for Sensornets similarly
utilises cryptographic onions for establishing route pseudonyms [NMIM08]. Following a
topology discovery phase, the protocol makes the base station responsible for assigning
pseudonyms to each hop of the communication links from the sensor nodes towards
it. It informs the sensor nodes about their corresponding pseudonyms for incoming
and outgoing packets by locally broadcasting a cryptographic onion per each main
communications branch, which contains the incoming pseudonym and the outgoing
pseudonym for each sensor that can successfully decrypt each layer using its pre-shared
symmetric key. Each sensor broadcasts the rest of the onion after peeling off the outer
layer, until each hop of the communications with the base station is assigned with route
pseudonyms.

When a sensor node has a packet for the base station, the construction of the
cryptographic onion is reverse. The sensor node encrypts it with its symmetric key and
uses its outgoing route pseudonym to send it to its next hop neighbour. The neighbour
whose incoming route pseudonym matches the packet route pseudonym accepts the
packet, switches its pseudonym to its own outgoing route pseudonym, and encrypts
the already encrypted payload with its own symmetric key before forwarding it. Every
sensor node in the path repeats this process, and the onion is created en route to the
base station. As a result, for an eavesdropper running traffic analysis, the packet looks
randomly different on consecutive links. Upon receiving the packet, the sink performs
recursive decryptions to recover the original data.

Since node identities are not used for message routing, and the unlinkable route
pseudonyms have the scope of a single hop for each sender and receiver, adversar-
ies overhearing communications around any node participating in data forwarding can
neither identify the communicating parties nor traceback the messages. Furthermore,
the described protocols are intrusion tolerant. In the case of a compromised node, only
its own route pseudonym correspondences would be revealed, thus the effects would be
localised. The route pseudonymity approach is better fitted for multihop communications than the node pseudonymity approach, because of the pseudonym synchronisation issues of the latter. Regarding the processing and communication overheads of the route pseudonymity protocols, they do require the use of expensive message flooding techniques, cryptographic operations and trapdoor functions but, most of them, only during route discovery and pseudonym setup.

3.2.3 Privacy Sensitive Information Disclosure

As opposed to the context protection mechanisms discussed in the previous section, the privacy sensitive information disclosure schemes aim to protect the content of the messages from disclosure to illegitimate entities. Guaranteeing strong privacy through complete or minimal disclosure of information is not desirable in all situations. Some services that users may perceive as useful shall require the disclosure of sensitive information. The solutions that are discussed in this section aim to control the privacy tradeoffs according to the service information requirements and the restrictions applied by user privacy preferences. The mechanisms that are discussed aim to enable users to control if any of their personal data should be disclosed, and at what level of detail, according to the context surrounding data requests. The main issues that are addressed are related to data access control, data granularity control, and protection from inference through information correlation.

3.2.3.1 Privacy Policies and Preferences for Access Control

The control of sensitive information flow from deployments that collect the data to applications that request the data in order to provide services is typically addressed through the use of privacy policies and preferences. The more complex the data collection and distribution environment is, the more challenging the specification and enforcement of privacy preferences becomes. The case of location based services is a typical example: location information can be collected by multiple deployments with varying levels of trust that use different technologies, like networks that identify and locate user-carried devices or environmental sensor networks that monitor the presence of individuals, and distributed by some middleware service to multiple, possibly untrusted, user-centred or location-centred applications.

In one of the earliest approaches to privacy preferences enforcement in environments with different administrative entities, trusted user agent components that reside on user controlled devices were proposed to act as intermediaries, collecting and controlling access to personal information through predetermined access policies [ST93]. A distributed location query service responds to location-centred requests of external third parties. Location brokers residing in the middleware layer are used to interact with the user agents in their regions. However, although this approach enables access control in a decentralised manner, it requires the user agents to collect all information required for access control decisions.

The middleware service that allows applications to query the locations of users independently of the underlying technologies being used is provided by a centralised location server in reference [MFD03]. This service is designed to control the distribution of location information by using machine readable privacy policies, defined as an
extension of P3P policies, and without the need for repeated user intervention, in order to minimize intrusiveness. It uses user-defined privacy preferences to determine the acceptance or rejection of information release requests, while requiring user approvals only if the request can not be handled by the established preferences. The preferences that govern access permissions can include factors like the organisation requesting the data, its information handling policies, the type of service that is offered, time, location and context limitations, and constraints on the types of requests that can be accepted, i.e., specific user location requests, enumeration requests for specific locations, or asynchronous requests for information on specific events, like when users enter or leave specific areas. An issue that was studied in this approach is how user privacy preferences will be specified. It is proposed that default preferences are provided by service providers and other trusted organizations, and simple tools like wizards are used to help users create appropriate configurations.

When the users subscribe to the location server to make their location information available for external applications, they register their privacy preferences, which take the form of system components, called validators. When an application requests location information, it includes with the query a statement of its privacy policy, in order for the validators to evaluate the acceptability of the privacy policy, to decide if the requested information will be released, and whether any special constraints, like for the acceptable data accuracy, should be imposed. Multiple validators may exist for a single user, at least one corresponding to each of his identifiers, and the data release decision may be taken cooperatively. Potential validator components include user confirmation components and external validation services, for example services providing information on the requestor’s reputation. Anonymity support is provided through enabling the validators to determine for each request which user identifier will be returned - the long-term identifier, a short-term identifier associated with the user, or a new randomly generated identifier.
The centralised nature of the access control decisions by the middleware location service of this solution, however, raises scalability concerns. In the approach proposed in reference [HS05], it is considered that access control through location privacy policies enforcement should be performed in a distributed way. It should be flexible enough to support multiple sources that provide location information, that may be within different administrative domains or belong to different organisations, with different levels of trust. Privacy preferences in this approach similarly determine who can access location information, what level of accuracy is allowed to be disclosed, and for which locations and time intervals. One significant difference from the previous approach is that privacy policies can be specified not only by individuals, but also by central authorities, like companies that do not want the location of their employees to be disclosed to outsiders. Moreover, the scheme allows access rights to be forwardable.

The sources that provide location services are hierarchically organised as in the example of Figure 3.5 and each of them either uses a particular technology for collecting location information, like the positioning service, or processes the information received from other location services, like the device locator. In order for the hierarchically organised location services to cooperate in sharing information and propagating it to the upper layers for the requestor to receive it, the notion of service trust is introduced. Within the privacy preferences of each entity, the trusted location services are defined. Each trusted service implements access control by checking if the privacy preferences allow disclosing the requested information to the entity that issued or forwarded the request. This way, access control is performed in all steps of data collection or forwarding by the distributed location services. Privacy policies are encoded as digital certificates and the SPKI/SDSI scheme, originally proposed for decentralized trust management that supports specification and evaluation of security policies, is used for the implementation of the certificates.

Policy based access control is based on an alternative data model in the Confab framework [HL04], proposed in order to provide software architecture support for privacy-sensitive decentralized ubiquitous computing applications. According to its data model, all contextual entities, like users, services or locations, are assigned infospaces, which are network-addressable logical storage units that keep static and dynamic, intristic and extristic sensed data about these entities. Infospaces representing contextual information about users are kept in user-owned devices. Each piece of information in infospaces can be associated with privacy tags to describe how it should be handled when external entities request it. Infospaces contain operators for enforcing the privacy preferences of the user and the restrictions set by the privacy tags, and operators for user notification and interaction and for the evaluation of service descriptions that the applications publish when requesting data in order to describe the type and granularity of personal information required for each provided service level.

In the context-aware privacy protection framework proposed by Mitseva et al. [MWP08], privacy policies and privacy level flags are used to indicate how particular pieces of sensitive data of varying granularity should be handled. A privacy agent component is proposed for determining if, and in what form, data should be disclosed, and for invoking privacy safeguard mechanisms that filter personal data before any disclosure, either by allowing or forbidding it, or by pseudonymising it. Context attributes are evaluated using profiles and rules as input to a context assessment block, so that data disclosure decisions are made by evaluating context data like the requesting party,
the user role, their locations, and other related attributes against privacy policies.

3.2.3.2 Information Granularity Control

Privacy policies can be used not only to control if personal data will be disclosed within a specified context, but also to control the acceptable level of detail and accuracy of the disclosed data. In this sense, information granularity control approaches do not focus on controlling data release, but instead enable controlling the granularity of the data so that its disclosure is in accordance to the user’s privacy preferences for the given context. Granularity control applies especially to location related requests, enabling the adjustment of location data accuracy. These approaches assume that accurate and detailed data is being collected by a trusted network and is released upon request to service providers. Because the service providers are not necessarily trusted by the users, the aim is to prevent the release of detailed data to them, by restricting data accuracy to the minimum level that is necessary for the services to be offered.

Except from location information, in the scheme proposed in reference [Sne01], the observations of data subjects that can be protected through granularity control can include their identities, their speed during the observation, and the time that the observation was made. Information granularity control based on location privacy preferences is performed by the location providers, i.e., the trusted entities that collect the user’s accurate location information and have some interest (e.g. for compliance with regulations or service agreements) in letting subscribers control the release of their location data. The service providers in this setting are the untrusted entities that require location data of some level of detail from the location providers to provide location based services. The granularity of private data is reduced prior to its release to the service providers according to the privacy preferences, the role of the requestors and the purpose of information usage. The responsibility for privacy policies enforcement in this scheme is thus split between location providers that are responsible for reducing the accuracy of the data, and the service providers that are responsible for restricting the usage of the data to the stated purpose. The accuracy of data related to the location, time, speed and identity of the user is modelled using lattice structures. Partial identification of users can be done by defining sets corresponding to sex, occupation, nationality, employer etc. The scheme enables adjustment of spatial, temporal and identity accuracy according to policies and data use purposes.

Although reducing information accuracy according to predefined policies can protect the exact location of users, this approach does not suffice in protecting their anonymity. In the approach proposed in reference [GG03], spatial and temporal information granularity is reduced according to the user’s anonymity set size, in order for the disclosed information to be sufficiently altered to prevent reidentification. Similarly to the previous scheme, external service providers receive information through trusted location brokers. The location brokers act as a mixrouters by randomly reordering incoming messages to make them unlinkable with the outgoing messages, remove any identifiers such as network addresses, and reduce information granularity before releasing the information in order to make the subject $k$-anonymous, i.e., not identifiable within a set of $k$ subjects comprising its anonymity set. With respect to location information, $k$-anonymity is achieved if the location information released is indistinguishable from the location information of at least $k - 1$ other subjects.
3.2. PRIVACY PROTECTION

The quadtree-based algorithm that has been proposed for reducing information granularity meets an anonymity constraint in any location, regardless of population density, by decreasing the resolution of the revealed spatial and temporal data. It takes as inputs the accurate position of the subject, the coordinates of the area covered by the location broker, the positions of all other subjects reporting their location through it, and the minimum acceptable anonymity set size $k_{min}$. For reducing spatial accuracy, it works by subdividing the area around the given position until the number of subjects within the area falls below $k_{min}$, and returns the last acceptable quadrant, which represents the smallest area where $k_{min}$ anonymity is preserved. For spatial resolution to be improved to a smaller area, anonymity can alternatively be preserved by adjustments in the resolution of the temporal dimension, by delaying information release until $k_{min}$ subjects have reported from within the given area. Although this would allow for more precise anonymous information, it is unsuitable for time critical services, because of the delay.

In the database field, the work presented in reference [Swe02] has a similar goal: to disclose person-specific information by adhering to $k$-anonymity while ensuring minimal distortion, so that the released information remains practically useful but the identity of the individuals who are the subjects of the data cannot be determined. $K$-anonymity is achieved through generalisation and suppression mechanisms. The difference of information granularity control that is discussed in this section, however, is that the data to be protected is dynamic and collected in real time.

3.2.3.3 Protection Against Location and Identity Inference

The solutions that have been discussed in the previous sections to provide user identity privacy and location privacy in location sensing environments involved the enforcement of privacy policies and granularity control by trusted middleware services. However, for services that require precise location information, the use of privacy policies alone can not protect from identity and location inference by adversaries or illegitimate service providers that may be given access to location records. In order to protect identity and location privacy while allowing users to take advantage of location based services, mechanisms that protect from identity or location inference are proposed as complementary to the privacy policies enforcement schemes, for scenarios where granularity control can not be applied.

The problem of precise location information in location tracking environments that enables user identity inference can be tackled though pseudonymisation and the use of mix zones [BS03]. This approach assumes that the user trusts both the deployment that collects location information and the middleware service, but distrusts the location service providers. The middleware service acts as the access policy enforcement point and as an anonymisation proxy, responsible for the generation and frequent update of pseudonyms, so that users can not be identified by their presence in the reported locations. The purpose of the pseudonyms in this approach is the temporary identification of users so that they are provided with a return address for the services offered. However, even if the pseudonyms are frequently updated, their unlinkability depends on the size of the user’s anonymity set: if it shrinks, his pseudonyms can be correlated. The concept of mix zones is used for pseudonyms to be updated in an unlinkable way. A mix zone for a group of users is the largest connected spatial region where no user
has registered any location service callback. An example mix zone setting is presented in Figure 3.6. Mix zones can either be defined by the middleware service a priori or be calculated as the spatial areas that are not application zones, i.e., where no location service requests location information. Pseudonyms are updated when users are within mix zones, and therefore location services do not receive location information, so that the pseudonyms of the users coming into the mix zone can not be linked with the updated ones.

The anonymity set of any user in a mix zone is equal to the number of incoming people during his stay in it. The size of the anonymity set can therefore be used as a measure for the unlinkability of the pseudonyms and the resulting location privacy. It is, however, an optimistic measurement, since a powerful adversary could make intuitive observations and assign different probabilities to pseudonym correspondences. An example observation is the consistency that would probably exist in the direction people are moving towards when entering and leaving the mix zone. In Figure 3.6, for example, it is more likely that pseudonym $P_x$ is linked to $P_{x+1}$ than to $P_{y+1}$, since this would mean that the user changed his direction while being in the mix zone. The entropy metric is therefore considered as more accurate in measuring location privacy when using the concept of mix zones.

A different problem is the one of location inference of users that define in their privacy preferences some sensitive areas where their presence should not be revealed. This is an issue that can not be addressed by privacy policies enforcement alone. A solution that would protect against location inference in environments where periodic location reports are accumulated and sent to service providers should not only suppress location updates when the user is located within sensitive areas, but also suppress prior movement path data that would indicate or enable inference of the current location. This approach also assumes that the user trusts both the location positioning network and the middleware service, and that the adversary is any entity that seeks to infer which sensitive areas the user visits through accessing the location records that are transmitted to external service providers. The middleware service uses location sensitivity maps that identify sensitive locations according to the user’s settings, acts as a policy enforcement point and executes the disclosure control mechanisms to determine whether location updates can be sent to the requesting service providers.

An adversary should not be able to probabilistically infer user presence in sensitive
areas from prior or future location updates. The mechanism that controls disclosure of location data should maximise position uncertainty when the user is in a sensitive area, while minimising location information distortion when the user is outside sensitive areas. The \( k \)-area algorithm is used to suppress location updates in a region around a sensitive area, so that the area cannot be distinguished from at least \( k-1 \) other sensitive areas that the user might have visited. The location sensitivity map is partitioned in zones, each including \( k \) sensitive areas. All location updates from each zone the user is moving within are stored and released to external applications only when the user crosses a zone boundary and has not visited any sensitive areas in the zone. This way, location accuracy is preserved when the user is moving outside sensitive areas, while making the sensitive areas the user visited indistinguishable.

As in the approach against identity inference, however, the size \( k \) of the set of indistinguishable sensitive areas is an optimistic measure of the location privacy protection that this approach offers. An adversary might have prior knowledge on the user’s relationship to sensitive locations, which enables the assignment of different probabilities for the sensitive areas in each zone that the user is known to enter. Moreover, the algorithm discloses other information, such as the frequency and duration of visits to zones. At the same time, the delay that this approach introduces for the disclosure of information makes it inapplicable for time critical location based services.

3.2.4 Privacy Sensitive Information Gathering

The approaches that were presented in the previous section for safeguarding user privacy were based either on the use of privacy policies or on the reduction of data granularity before disclosing it to service providers. The first case assumes that the service provider is trusted to adhere to privacy policies, while in the second case, trust is assumed for the intermediary that is responsible for adjusting data accuracy before disclosing. However, even the deployments of legitimate intermediaries and service providers, may be passively or actively attacked, violated or misused. The privacy sensitive information gathering approaches, which include controlled data gathering and user’s notice and choice, are more proactive, in the sense that they aim to protect user privacy at the point of information capture, before any data release decisions need to be made, to prevent privacy-sensitive data from being accumulated.

3.2.4.1 Restricted Data Gathering

The privacy sensitive data gathering approach aims to protect the privacy of users through restricting the sensor network’s data collection capabilities to the minimum level that is required for the services to be provided. The privacy mechanisms of this approach are applied during data collection to prevent privacy-sensitive data from being accumulated in the WSN, before intermediaries or service providers gain access to it. The only part of the deployment that needs to be trusted is the WSN since, if sensitive data is not collected at all, trust does not need to be assumed for how it is subsequently handled.

A type of applications for which sensitive data gathering can be applied are those that require aggregated information for the population density in certain areas without needing to track the movements of specific individuals, and thus without needing to
Identify them. Applications for road traffic monitoring or transportation schedule monitoring are examples of location-centric deployments. The approach presented in reference \cite{GSJ+03} used the example of a location sensor network for an in-building occupant movement tracking system like the one in Figure 3.7. The network is composed of a number of environmental sensor nodes capable of determining the number of individuals in the area monitored, some base stations and a location server that collects the data and makes it available to applications. In order to defend against traffic analysis attacks and to avoid the panda-hunter problem of Section 3.2.2.1, at the cost of increasing computation and communication overhead, messages are encrypted and data traffic is regularised by requiring all nodes to send at least one packet per data gathering interval even if they have no activity to report. Both the sensor nodes and the monitored areas are hierarchically organised, so that population statistics can be extracted for different levels of spatial resolution. Within this sensing environment, although no information about the identities of individuals is gathered, an adversary that has prior knowledge about the individuals and the spaces that they frequently use could link identities to the information reported by the location server and could track their movements within the area.

To counteract this threat, the data gathering capabilities of the sensor nodes are leveraged through applying distributed, in-network anonymity mechanisms that dynamically change data accuracy in order to preserve $k$-anonymity of the subjects within the WSN before the data reaches the location server. In order to preserve $k$-anonymity while retaining the usefulness of the data, these mechanisms force the minimum necessary reduction in data accuracy in various steps of a hierarchical data aggregation process. The hierarchy reflects the spatial hierarchy of the sensed area, and multiple nodes in all levels of the hierarchy aggregate data while reducing its accuracy, so that no single node has a complete view of the data. Each node can be identified either uniquely or by the identifiers of the nodes that are above it in the hierarchy. The data accuracy is reduced by a rounding function in two ways: the spatial accuracy which is reflected by the identification level that the node which provides the information uses, and the accuracy of the number of subjects reported in the node’s area. Given an anonymity level $k_{\text{min}}$ for each node $i$, if the number of subjects $k$ is above $k_{\text{min}}$, the unique node identifier $S_i$ is used and the accuracy of $k$ is reduced. Otherwise, the node identifier is blurred by using only the identifier of a higher level, as in the cases
3.2. PRIVACY PROTECTION

in Figure [3.7] Eventually, the information that reaches the location server is moderately accurate and only for the levels of the area hierarchy where the total number of subjects in their sub-regions exceeds $k_{\text{min}}$. TinyCasper [CMH08] is a prototype system that implements a similar technique for preserving $k$-anonymity. It includes an aggregate query processing module which supports aggregate and alarm queries through a spatio-temporal histogram of the monitored area that it maintains and updates with every anonymised report.

The hitchhiking approach [TKFH06] is targeted to the case of privacy sensitive user presence identification through carried devices instead of the environmental sensors alone. Like the previous approach, it was inspired by the observation that protecting user anonymity through reducing the accuracy of location data can make location information useless for applications that are location-centric. It requires each user to approve reporting from each location he visits. The identifiers that are used for the messages to the location server are location identifiers, computed by the client devices based on the physical properties, e.g. GPS coordinates, of the location. Since the location server knows these physical properties, it can infer what location is being reported on without being able to infer the exact identity of the device that sent a report. The data gathering capabilities of the network are restrained only in the user identification dimension, by enforcing the location reports that are sent to the location servers to include only the total number of the users detected around the reporting device, without identifying them. Since location data is anonymised, the degree of location privacy in this scheme depends on the density of the population in each location, as this comprises each user’s anonymity set.

The negative surveys technique, introduced in reference [Esp06] and applied for anonymous data collection in WSNs in reference [HGFE07], can be applied not only for location information but for any type of data that can take discrete values from a finite set or range of values. Upon a data request, the nodes reply with a random false value from the set, instead of replying with their actual data. The data requestor can use the false values collected from a number of nodes to retrieve the distribution of the actual data. In reference [HGFE07] it is derived that, for each discrete value $x$, the estimate of the number of nodes that would truthfully report it is $N_x = n - R_x(t - 1)$, where $n$ is the total number of reporting nodes, $t$ is the size of the values set, and $R_x$ is the number of nodes that falsely reported $x$. The accuracy of the estimates was found to increase the more nodes participated in data reporting and the smaller the size of the values set.

3.2.4.2 User Notice and Choice through User Agents

The provision of user notice and choice functionality requires the addition of extra components in the general case of WSNs architecture: the privacy assistants, being the user gateways to the surrounding WSN applications, acting as intermediaries, and applying user-defined policies on information requests. User agent components are introduced especially in the case of WSNs composed of user-related nodes, like body or vehicular sensor networks, that during their lifecycle may share information with services that are not fully trusted. As proposed in one of the earliest approaches in user controlled information disclosure in ubiquitous computing environments [ST93], the user agent components that reside on user controlled devices can collect and control
access to personal information. Any request from external services for such information must be routed through the user agents that enforce predetermined privacy policies. By providing context awareness capability to the user agents, they can act as policy coordinators, enforcing context-sensitive and customisable access control.

In order for user agents to enforce privacy policies according to the context, their role and the communication partner, mechanisms should be included to provide awareness about whether data collection is being performed and what service privacy policies are being announced. These user notice mechanisms can either be provided by service providers or by third parties. A privacy awareness system that allows for user control in ubiquitous computing environments, assuming that the service provider is cooperating and is trustworthy, is paw$S$ \cite{Lan02}. It includes mechanisms for the network service to announce its privacy policies and data handling practices, and for the users to apply their privacy preferences on accepting, declining or customising a service.

The paw$S$ scheme includes a user agent component and a privacy beacon component, responsible for announcing the data collection requirements and privacy policies of the services offered. The scheme differentiates between two types of data collection that require different mechanisms for communicating the privacy policies to user agents: implicit announcement, when the user initiates the service discovery process and actively requests the service privacy policy, and active policy announcement for services working continuously in the background, in which case the user agent receives the policy from a beacon upon entering the data collection area. The announced privacy policy can include various levels of service customisations, together with the type of data required for each case, so that the user agent can determine the accepted service level according to the predefined privacy preferences. The scheme also provides mechanisms for access and recourse of personal information through privacy proxies and privacy-aware databases. It empowers the user agent to keep track of data collections around the user, and enable or disable optional services based on the privacy preferences. It is, however, set as a prerequisite that the services are optional and configured to suit the users’ decisions related to their privacy, and that the service providers are willing to cooperate. The scheme should thus be viewed more as a privacy enabler than as a privacy protector.

An alternative privacy awareness scheme that has been proposed for pervasive sensor networks \cite{SS05} enables the user to conclude whether he is inside some sensing areas, without disclosing his exact position within the area. It is based on the protocol of secure two-party point-inclusion problem to test the privacy state of the user. If the user agent device is at a point $p$ and a data collection area covers a polygon $P$, the secure two-party point-inclusion protocol is used to determine if $p$ is inside $P$ without revealing to each other any information about their exact position. This scheme also achieves to protect information about the boundaries of the data collection area, which may be necessary in some commercial or military networks. For the execution of the protocol, two parties are required: a user agent device that can compute its current position through some external location service, and a sensor network node that either belongs to the service provider or to a trusted third party. The sensor network node can also serve as the privacy policy announcement point if it is concluded that the user is within the sensing area.

AnonySense was proposed as a privacy-aware architecture for pervasive applications that use sensed information from personal mobile devices \cite{CKK+08}. The architecture
enables the applications to collect verifiable but unlinkable sensed data reports from anonymous nodes. For this to be achieved, AnonySense positions additional system entities between the applications and the nodes: the registration authority provides certification services, the task service distributes data requests to the nodes, and the report service receives the sensed data reports, aggregates them and responds to the applications that requested them. A mix network [Cha81] is positioned between the nodes and the report service to achieve unlinkability between the nodes and their reports.

In order to provide software architecture support for privacy-sensitive and context-aware ubiquitous computing applications, Confab [HL04] was proposed as a toolkit that facilitates the development of client-centred architectures where personal information is sensed, stored, and processed on user-owned devices. It defines both the mechanisms for end users to control sensitive information disclosure, and abstractions and customizable privacy mechanisms for developers of privacy-aware applications. In order to give to the end users flexibility over the privacy tradeoffs they are willing to make, Confab enables applications to publish service descriptions that include various service level options. It facilitates the use of three basic interaction patterns, namely optimistic, pessimistic and mixed-initiative, where data disclosure decisions are made interactively by the users, and offers mechanisms for user control over the access, flow, and retention of personal information.

### 3.2.5 Analysis and Discussion

From the schemes that have been proposed, it becomes apparent that there exist solutions to meet most of the privacy requirements for protection of the communications context, privacy sensitive information disclosure, and privacy sensitive information gathering. The separate discussion of the approaches highlighted the diverse privacy aspects that they are focused on, and revealed how the approaches can be viewed as complementary to fulfil the complete spectrum of sensor networks’ privacy needs.

Privacy issues are addressed at multiple levels of the network stack and at different points of the information flow. Some schemes interfere with the routing protocol, by requiring modifications of the message headers or introducing routing path selection strategies, others interfere with the information flow through introducing intermediaries, while others interfere with the data through enforcing adjustments in its granularity. Some schemes aim to protect against adversaries overhearing the communications, while others aim to protect against illegitimate access of information from service providers. Different assumptions are made about the entities that are considered trusted, with some schemes trusting only the WSN deployment and other schemes assuming that there exist trusted intermediaries to enforce privacy mechanisms. Some schemes address user privacy concerns related to the information captured and transmitted by user-carried devices, while others address concerns with respect to environmental sensor networks that capture information about people in their proximity.

However, some issues can not be disregarded; firstly, most of the schemes presented, especially for protecting the context of the communications, interfere with the system design at the routing protocol and information flow levels, which complicates their actual integration to the deployments. The deployment complexity of these solutions may hinder their adoption for practical, real-world WSN scenarios, where privacy protection
is not the core functionality of the network that is deployed. Application-level solu-
tions, like some of these proposed for controlled information gathering and disclosure,  
would be the easiest to integrate to the deployments, but can not protect the context 
of the communications.

In addition, many of the solutions have increased resource requirements and im-
pose negative effects in the operation of the network: The phantom routing strategies  
increase the length of the multihop routes that the messages traverse to reach the base  
station, resulting to increased message latency and energy requirements. The dummy  
traffic and message buffering approaches have similar drawbacks and, in addition, high  
communication and memory requirements. In terms of resource requirements, the less  
expensive solutions are the ones for restricted data gathering which are, however, de-
signed for very specific types of WSNs.

A number of the solutions that have been proposed have limited applicability. The  
node and route pseudonymity approaches are suitable only for relatively small and static  
networks, due to pseudonym setup and synchronisation issues. Most solutions of this  
category require resource consuming cryptographic operations during pseudonym setup,  
while the route pseudonyms depend on the network topology, so they would exhaust  
the energy supplies of a network where frequent topology changes occur. Most privacy  
sensitive information disclosure approaches can be applied only when there exists a  
trusted intermediary or a middleware service to apply data access or granularity control. 
Very few solutions exist for privacy sensitive information disclosure by the sensor nodes,  
which can be attributed to the fact that these solutions require information for the  
anonymity set size, which is not available to the sensors. Privacy sensitive information  
gathering solutions, that are applied by the sensor nodes, are either too context-specific,  
or not suitable when detailed information is required for a service to be offered.

Moreover, issues related to controlling the granularity of data, anonymising or  
pseudonymising it depend on the application domain, the in-network data processing  
schemes and the privacy sensitivity of each user. Thus, it may be infeasible to design 
a generic and high level privacy architecture, that could both be independent of the  
underlying networking protocols and guarantee some level of privacy independently of 
the context of the deployment.

Another issue is related to the level of trust users need to have to the deployments  
in order to take full advantage of the services that can be offered. None of the schemes  
discussed can protect against malicious service providers, that do not adhere to their  
announced privacy policies. The definition by the users of strict privacy policies that  
would guarantee that personal information is not disclosed, would also not allow them  
to use legitimate services that require that information. It would thus be necessary  
to build some level of trust to legitimate deployments, which can not be accomplished  
using solely technical means.

### 3.3 Security Management

WSNs require the provision and management of core security services to ensure the  
authenticity, the confidentiality and the integrity of the data generated and commu-
nicated over the network. These services are needed by the sensor applications that  
involve the cooperation of nodes and the communication of data, and by the trust
management and privacy protection solutions which assume the use of core security services for protecting the confidentiality of the exchanged messages. However, the characteristics of WSNs make most traditional security mechanisms inapplicable. At the same time, as explained in Section 2.3.3, the integration of WSNs with other networks and platforms and the heterogeneous capabilities and security requirements of the nodes and the deployments require the provision of flexibility and self-organisation properties.

The solutions that have been proposed to address specific security issues in WSNs, including the mechanisms for cryptographic key distribution, are discussed in Section 3.3.1. The solutions that have been proposed for the integration and management of security services are then presented in Section 3.3.2.

Throughout this section, the discussion focuses on the issues that are related to our work on security management. We do not discuss security issues at lower levels, like secure routing strategies, nor specific attacks and countermeasures, and refer the reader to references [WAR06, WLSC07, BFTZ09, CMYP09] for WSN-related attacks and secure routing solutions.

3.3.1 Security Primitives and Mechanisms for WSNs

The next sections outline the basic key exchange and distribution schemes that have been designed to serve as the basis for other security services such as authentication and encryption of sensor node communications, the secure data aggregation and verification mechanisms that have been proposed to deal with the presence compromised nodes within the network, and the findings of the studies that have been made on the applicability of security primitives and protocols on the sensor nodes.

3.3.1.1 Cryptographic Key Distribution and Data Encryption

The main requirement to be addressed for secure WSNs is to provide the means for authentication and data encryption. An issue that has received significant attention is how encryption should be applied. It has been argued that end to end cryptography is not the most suitable solution, because it would require that each sensor node stores encryption keys for all possible end points, while being incompatible with in-network processing and aggregation, passive participation and local broadcast [PSW04]. Alternatively, by using hop by hop encryption, each sensor node would only need to store encryption keys shared with its immediate neighbours. Link layer security was thus proposed as an attractive alternative to secure the authenticity, integrity and confidentiality of messages communicated between neighbouring nodes, while permitting in-network processing and providing the greatest ease of deployment [KSW04].

The basic prerequisite for providing authentication and encryption services is the sharing of cryptographic keys between the communicating parties. The cryptographic key distribution and management solutions for WSNs have to respect the resource constraints of the nodes, the infrastructureless and dynamic nature of the networks, and the possibility of node compromise. As a result, they can not be based on asymmetric cryptographic operations, nor require the nodes to store unique symmetric keys for all the nodes they communicate with. Moreover, they need to ensure that the effects of node compromise are localised and do not violate the forward and backward secrecy
requirements. A wealth of lightweight cooperative key management solutions, which are surveyed in references [CGPM05, CY05, XRS+07, Rom08], have been proposed to address these requirements and limitations.

The solution that prevails towards meeting the core security requirements in WSNs is symmetric key encryption, due to its low computational cost and memory requirements. The random key pre-distribution technique was introduced by Eschenauer and Gligor in reference [EG02]. It assigns each sensor node a random subset of keys from a key pool before deployment, so that each pair of sensor nodes has a certain probability to share at least one symmetric key. Various improvements and extensions to the basic probabilistic scheme have been proposed. These include the random pairwise keys scheme and the $q$-composite scheme [CPS03], that requires the existence of at least $q$ common keys to set up a link between any two nodes. Moreover, a number of solutions utilise pre-deployment knowledge to improve the performance of pairwise key pre-distribution [DDH+04, LN05, LND08]. The group-based deployment model [LND08], assuming that sensor nodes are deployed in groups and that the nodes in the same group are close to each other after deployment, pre-distributes the symmetric keys according to the group-based deployment knowledge. The heterogeneity of the nodes in WSN deployments was utilised by the asymmetric pre-distribution scheme [DXGC07] to assign larger key pools to the more powerful network nodes and to make them responsible for setting up the keys of the sensor nodes.

One of the most well known security architectures that utilises symmetric cryptography is SPINS [PST+02], where asymmetry is introduced into symmetric key cryptography through delayed key disclosure and one-way function key chains. It includes the protocols SNEP for data confidentiality, two-party data authentication and evidence of data freshness, and $\mu$TESLA for authenticated broadcast. The LEAP protocol suite supports multiple symmetric keying scopes, including individual keys, pairwise shared keys, cluster keys, and group keys, each providing different security levels for the exchanged messages [ZSJ03]. The LiSP security protocol uses group temporal shared keys, and is based on a rekeying mechanism periodically performed by the group-head nodes [PS04], while PIKE facilitates pairwise key establishment using peer sensor nodes as trusted intermediaries [CP05].

Although traditional public key cryptography was initially considered inapplicable for WSNs, Elliptic Curve Cryptographic (ECC) key generation [Mil86, Kob87] emerged as an attractive alternative that would allow for greater scalability and flexibility [AEAQ05], while being efficient enough to be attained and executed on resource-constrained sensor nodes [WGE+05], mainly due to the fact that it can offer equivalent security with smaller key sizes (it is estimated that in order for security levels to be equivalent to these achieved with symmetric key length of 128-bits, the key size is 283-bits for ECC systems and 3072-bits for RSA [Lau04]). Other asymmetric key algorithms have also been considered for WSNs [GKOS05], such as NtruEncrypt [HPS98], a relatively new cryptosystem that claims to be highly efficient and particularly suitable for embedded applications and low computational power devices. However, provided the infrastructureless nature of WSNs, a challenging concern related to applying public key cryptography is what will comprise the certification authority. The approaches discussed in Section 3.1.3 have been proposed to address this issue through using offline certification authorities, the web-of-trust approach, or threshold cryptography.
3.3. SECURITY MANAGEMENT

3.3.1.2 Secure Data Aggregation

The vulnerability of WSNs to active internal attacks through node compromise increases the risk of producing false data. For example, a stealthy attack, where an attacker’s goal is to make the network accept a false data value using a compromised node, combined with a sybil attack, where a malicious node illegitimately claims multiple identities, would allow a single compromised node to have a great impact on causing a false aggregation result [SP04].

At the same time, WSNs allow for redundancy on the views of the environment, which can be exploited for ensuring data correctness either by using majority voting between the nodes that were around a reported event, or by cross-checking the collected results for consistency. This characteristic is exploited in a scheme proposed in reference [DDHV03] in order to ensure the validity of the results provided by aggregator nodes, where Message Authentication Codes (MACs) produced by witness nodes that conduct the same data fusion operations as the aggregators are used as proofs.

A reactive scheme aiming to ensure the results provided by aggregator nodes are good approximations of the true values, even if the aggregators and a fraction of the sensor nodes are compromised, is proposed in reference [PSP03]. It is based on representing the data used for the aggregation using Merkle hash trees, that the base station can verify using random sampling mechanisms and interactive proofs. The problem of compromised nodes tampering with the transmitted data on the network path was studied in reference [ZSJN04], where an authentication scheme is proposed to guarantee that the base station will detect any injected false data packets when no more than a certain number of nodes are compromised.

The concealed data aggregation scheme [WGA05] was proposed to address a different issue: To enable the in-network aggregation of data that is encrypted with end-to-end symmetric keys. The scheme enables the aggregator nodes to operate on encrypted data, without decrypting it, by applying privacy homomorphism encryption transformations. The Private Data Aggregation Scheme [HLN07, HNL08] was proposed to preserve the secrecy of the data during the aggregation operations. It builds on slicing and assembling techniques and the associative property of the aggregation operators, like the addition operator. It requires each sensor or aggregator node to hide its individual data by slicing it, encrypting the data slices, and sending them to different neighbouring aggregators, that collect and route the aggregated results to the base station.

3.3.1.3 Applicability of Security Primitives

Several experimental evaluations of the energy consumption of cryptographic primitives and security protocols on sensor nodes have been reported. These include experiments on CrossBow and Ember sensor nodes [CMN07], on MICA2 nodes [GSSK05], and on Mica2dot nodes [WGE05]. These experiments showed that symmetric cryptographic operations are considerably less costly than asymmetric operations, and lightweight symmetric cryptographic algorithms are considered acceptable for resource-constrained sensor nodes. The energy costs of symmetric cryptographic operations on Mica2 sensors, featuring 7.3 MHz ATmega128L microcontroller, with 128 KB program
Table 3.4: Energy costs of cryptographic operations on sensor nodes

(a) Impact of 29-bytes payload cipher (CBC) on CPU consumption on MICA2 [GSSK05]

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time (ms)</th>
<th>Energy (µJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkipJack</td>
<td>2.16</td>
<td>51.84</td>
</tr>
<tr>
<td>RC5</td>
<td>1.50</td>
<td>36.00</td>
</tr>
<tr>
<td>RC6</td>
<td>10.78</td>
<td>258.72</td>
</tr>
<tr>
<td>TEA</td>
<td>2.56</td>
<td>61.44</td>
</tr>
</tbody>
</table>

(b) Energy cost of asymmetric computations on Mica2dot [WGE+05]

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Signature (mJ)</th>
<th>Verification (mJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSA-1024</td>
<td>304</td>
<td>11.9</td>
</tr>
<tr>
<td>ECDSA-160</td>
<td>22.82</td>
<td>45.09</td>
</tr>
<tr>
<td>RSA-2048</td>
<td>2302.7</td>
<td>53.7</td>
</tr>
<tr>
<td>ECDSA-224</td>
<td>61.54</td>
<td>121.98</td>
</tr>
</tbody>
</table>

memory and 4KB data memory, were measured in reference [GSSK05] and are summarized in Table 3.4 (a).

Similar evaluations for asymmetric operations show that ECC is considerably less costly than traditional public key cryptography [AEAQ05, WRZ06, WGE+05]. The energy costs of asymmetric cryptographic operations on Mica2dot sensors (Table 3.4 (b)), with 4MHz ATmega128L 8-bit microcontroller, were measured in reference [WGE+05]. Apart from ECC, Gaubatz et al. analysed and compared Rabin’s Scheme [Rab79] and NtruEncrypt [HPS98], and demonstrated that special purpose ultra-low power hardware implementations of the public key algorithms can be used on sensor nodes [GKOS05].

The applicability of various cryptographic primitives, including hash function, symmetric encryption, and public key cryptography primitives on sensor nodes has been studied in references [RAL07, Rom08]. Different platforms and hardware components of research-based and commercial-based sensor node devices, and varying existing hardware and software implementations of the cryptographic primitives were examined to facilitate the analysis of how the hardware influences the provision of the primitives. Regarding hash functions it was found that, since most are optimized to work with 32 bit microprocessors, they can not be efficiently implemented by the 8 and 16 bit microcontrollers used in sensor nodes. Regarding symmetric encryption algorithms, it was found that the stream cipher RC4 is the most optimal for sensor nodes, with SkipJack also being a good candidate, while regarding public key algorithms ECDSA was found suitable for Micaz and TelosB nodes.

3.3.2 Security Services Integration and Management

WSNs require the provision of complete security solutions which, as explained in Section 2.3.3, can support the diversity in the capabilities and the security requirements of the nodes and the deployments, facilitate the secure integration of WSNs with other
networks, enforce pre-defined settings and policies in each node, and provide flexibility and self-organisation properties. However, while a significant amount of research has been carried out on specific security, privacy and trust services for WSNs, few published works are found on the integration and management of these services.

The adaptive security solution proposed by Prasad and Ruggieri [PR04] for low data rate networks includes a security manager component, which is positioned above the link layer and controls the supported security mechanisms. The motivation for the provision of the security management service is the heterogeneity between the network devices and the deployments and that, analysing different security protocols and mechanisms, the authors found dependencies between the security levels that the protocols offer and their impact on the nodes in terms of the required computational power, battery, and overhead. The solution defines three security levels, along with their corresponding protocols and mechanisms, to address the security requirements of the different nodes and scenarios. The security manager is used to integrate the security elements, decide on the security level that is adequate for each service, and enable the network to adapt to the different needs of devices and services.

The security architecture proposed for WSNs in reference [WGS06] utilises the toolbox concept for the integration of various security services. The authors identify five key features that need to be provided by WSN solutions: scalability, security, reliability, self-healing and robustness, with the required strength of each depending on the application. The modular toolbox approach enables the integration of mechanisms that address different security issues, including authentication, concealed data aggregation, key distribution and secure distributed data storage, and the configuration of the security architecture by manufacturers and service developers according to the application security requirements.

The heterogeneity that exists between the capabilities of the nodes in WSNs is utilised in reference [MWY08] for the design of a lightweight security framework. Using the framework, the computationally complex authentication and key distribution operations are performed by the gateway nodes; each network node has a public/private key pair, but only the gateway nodes use public key cryptography to compute digital signatures and vouch for the sensor nodes. Trust is used as the basis for security related decisions in the PLUS (Parameterised and Localised trUst management Security) framework for WSNs [YKD08]. The framework includes a localised trust model for each sensor node to rate the trustworthiness of its neighbours and apply appropriate cryptographic methods accordingly. A secure routing protocol is devised as an application of the framework. However, neither of the two frameworks aim to provide complete security solutions, since each addresses only part of the WSNs security, trust and privacy requirements.

In the field of next generation networks, the Daidalos security framework [FDM+05] has been devised for mobile devices to allow both operators to securely provide their communication infrastructures, and third party providers to securely provide their services, while preserving the privacy of the end users. The security functionality provided by the framework includes privacy and identity management, privacy policy negotiation, multiple authentication mechanisms, distributed authorisation mechanisms, and distributed key management, all integrated to support multiple sessions of users using different devices. For intra and inter-domain key management, Daidalos includes a key distribution centre component in charge of managing credentials, certificates and
digital identities. It provides support for the integration of WSNs in the infrastructure, but does not specify the security mechanisms to be used within the WSNs.

3.3.3 Discussion

A number of different approaches exist for key distribution and management and for the authentication and encryption of sensor node communications, based both on symmetric and on asymmetric cryptography. Moreover, different solutions have been proposed for secure data aggregation and verification in the presence of compromised nodes.

The several experimental evaluations and analyses that have been made on the resource consumption and the applicability of cryptographic primitives on sensor nodes produced results that are particularly useful for the design of security solutions for WSNs. These results include the symmetric and the asymmetric cryptography primitives that were found to be the most appropriate for different sensor platforms, and estimations on the resource overheads that they incur.

While there exist works on certain issues related to security, privacy and trust in WSNs, an issue that we consider that has not been sufficiently addressed is the integration of the security services in complete and flexible security management frameworks. There exist solutions offering flexibility [PR04], adopting a configurable toolbox approach [WGS06], exploiting the heterogeneity of the nodes [MWY08], using trust as the facilitator of security-related decisions [YKD08], or enabling the integration of WSNs with other platforms [FDM+05]. We did not, however, find any complete solution for WSNs combining all these characteristics and offering flexibility, adaptability and context-awareness for all security, privacy and trust services.

3.4 Summary

This chapter presented the solutions that have been proposed for WSNs to address the requirements for trust management, privacy protection, and security management. The trust management solutions were categorised into certificate-based, behaviour-based and hybrid models, according to their scope, purpose and admissible types of evidence. The certificate-based models were separately discussed according to the entities that provide certification authority functionality, while the behaviour-based according to their approach for modelling and evaluating trust. What the analysis has also shown, however, is that the computational complexity of the certificate-based and the energy requirements of the behaviour-based trust evaluation models limit their applicability on resource constrained sensor nodes. Moreover, the differences in scope and purpose between the two first categories of models highlighted that they should not be viewed as alternative approaches, but as supplementary.

The hybrid models combine different approaches on trust establishment to benefit from the properties of trust that each individual approach offers. However, the hybrid models that were found in the related bibliography are all proposed for distributed systems or multiagent systems, and do not take into account the special requirements and limitations of WSNs, like their resource limitations and the varying connectivity, that make it challenging to obtain the different types of trust evidence that the models require.
Regarding the privacy protection solutions, it was found that there exist solutions to be applied at different levels of the network stack and at different points of the information flow to meet most of the privacy requirements. The solutions were categorised into the ones for protecting the communication context, the ones for privacy sensitive information disclosure, and the ones for privacy sensitive information gathering. The analysis of the privacy protection solutions highlighted some issues related to their applicability on WSNs. The integration of some of the mechanisms to the deployments may be complex, either because they influence the system design at the networking protocol level, or because they have increased resource requirements and would impose negative effects in the operation of the network. A number of the solutions that have been proposed were found to have limited applicability, being either too application-specific or requiring the existence of trusted intermediaries or middleware services.

The solutions that have been proposed to provide core security functionality in WSNs include a wealth of key distribution and secure data aggregation mechanisms. Several experimental evaluations and analyses have been performed on the resource consumption of various cryptographic primitives on sensor nodes, and it was found that a number of symmetric and asymmetric operations are applicable. However, few works were found on the integration and management of security services, none of which was targeted to WSNs and offering flexibility, adaptability and context-awareness properties and including all security, privacy and trust services.
Chapter 4

A Hybrid Trust Management Model for WSNs

WSNs are characterised by the distributed nature of their operation and the resource constraints on the nodes. The conventional view of security does not suffice given the unique characteristics of WSNs, that are susceptible to a variety of node misbehaviours. From compromised nodes acting as internal attackers to legitimate nodes that act selfishly, internal misbehaving nodes pose a vulnerability that cannot be tackled using strong security mechanisms alone. This vulnerability, along with the cooperative nature of WSNs, impose the need for a trust management solution to assess and maintain the trust relationships among the network nodes. Any trust management solution that is targeted at these environments needs to be lightweight in terms of computational and communication requirements, yet powerful in terms of flexibility in managing trust between nodes of heterogeneous deployments.

From the analysis of the related work in the area, we found that both the behaviour-based and the certificate-based models that have been proposed are better targeted at ad hoc rather than at sensor networks. The behaviour-based models view the network nodes as autonomous members, which is not the case for the sensors that, in practice, have an ownership, a role, and a purpose that justifies their placement or their membership in the network. Due to this assumption, these models do not support role-based trust and do not utilise the pre-deployment knowledge that may be available in WSN deployments. The certificate-based models have limitations that include their lack of feedback potential on the behaviour of the nodes. Moreover, the computational complexity of the certificate-based and the energy requirements of the behaviour-based models hinder their application on resource constrained sensors. At the same time, the hybrid solutions that have been proposed for other types of environments do not take into account the special requirements and limitations of WSNs, and do not specify concrete metrics and processes for combining the different types of evidence that they support.

Having studied the characteristics and the security and trust requirements of WSNs, and having analysed the related work on the area, we summarise the problem that we are trying to tackle to:

Addressing the need for a generic and secure solution for dynamically managing the trust relationships within and between heterogeneous WSN
deployments according to the pre-deployment knowledge, the network purpose, and the available feedback on malicious behaviours.

The term *generic* refers to the scope and the semantics assigned to trust, i.e. to the function that functional trust represents. The term *secure* refers to the robustness against attacks to the trust management model itself, like defaming attacks and conflicting behaviour attacks. The solution should be designed for *WSN deployments*, which implies that it should take into account the special characteristics and limitations of these environment both in node level and in network level. At the same time, the solution should support the *network purpose*, and the trust management solution should enable some control over the trust relationships according to this purpose. The *pre-deployment knowledge* on the role of each sensor node, its capabilities, and its trust evaluation requirements should be utilised. Finally, the solution should be able to revoke trust from nodes that exhibit *malicious behaviour*.

In this chapter, we present a generic trust management model that can uniformly support the needs of nodes with highly diverse network roles and capabilities, by exploiting the pre-deployment knowledge on the network topology and the information flows. The model is designed to support, through proper configuration, from simple nodes that have very restricted role, computational capabilities and should only trust the nodes they are pre-configured to trust, to highly adaptive nodes and gateways to other networks. The model is hybrid, combining aspects from certificate-based and behaviour-based approaches on trust establishment on common evaluation processes and metrics, in order to leverage the drawbacks associated with the individual approaches and to allow for flexibility in the trust establishment process. It enables controlled trust evolution based on the network pre-configuration, and controlled trust revocation through the propagation of behaviour evaluation results made available by supervision networks. The proposed model has been validated both through security property analysis and through simulation, and the results and analysis demonstrate its effectiveness in managing the trust relationships between nodes and clusters, while distributing the computational cost of trust evaluation operations.

In Section 4.1 we specify the scope and objectives of our work, including the requirements for the trust management model, our approach and main design decisions, the limitations of our work and the assumptions that we have made. Section 4.2 presents a working scenario that we later use for explaining the model and demonstrating the rationale for a number of our design decisions. The trust management model consists of components structured as discussed in Section 4.4, that implement the trust management processes that are presented in Section 4.5 in order to accumulate the trust evidence of Section 4.3.2 and establish the trust associations discussed Section 4.3.1. The trust management processes accumulate trust evidence according to the configuration parameters that are discussed in Section 4.3.3 and evaluate the trust associations using the metrics presented in Section 4.6. Section 4.7 includes the analysis of the model, of its security and trust properties and of its resource overheads, while Section 4.8 includes the experimental evaluation analysis, results and discussion.
4.1 Scope and Objectives

4.1.1 Trust Management Requirements and Approach

In the following paragraphs we analyse the requirements for the trust establishment model as they result from the problem statement in the previous section, and we present and document our approach for addressing them.

4.1.1.1 Application to WSN Deployments

The solution should take into account the special characteristics and limitations of WSNs. At the node level, this implies that it should be lightweight and make conservative use of the communication, computation, memory and energy capabilities of the sensor nodes. At the network level, this requirement implies that the solution should be decentralised, not based on centralised monitoring and management points or online trusted parties. It should support dynamic and mobile nodes, and it should be flexible in handling topology and membership changes. Moreover, due to the heterogeneity that may exist in real-world WSN deployments, the solution should be able to adapt to networks composed of nodes with heterogeneous capabilities, such as sensor nodes, cluster heads, aggregator nodes, or gateways to other networks.

Approach

Taking into account the resource limitations of the sensor nodes, the trust establishment model is not purely behaviour-based since, as explained in Section 3.1.6, we consider that behaviour monitoring is too energy consuming for the sensors. Moreover, it does not utilise mechanisms that require extensive cryptographic operations (like evaluation of certificate chains or threshold cryptography) to compensate for the absence of online trust managing authorities. Instead, it supports the use of certificates issued by offline authorities, whose public keys are held by the trust evaluator nodes for signature verification. To address the decentralised operation and the dynamic topology and membership, trust evaluation is distributed and cooperative: each node computes and maintains its trust relationships with other nodes individually, and trust evidence is made available in the form of recommendations. The verification of certificates and the exchange of recommendations do, however, entail significant resource consumption. In order to minimise the resource implications without limiting the supported trust evaluation mechanisms and primitives, the model exploits the heterogeneity of the nodes for distributing the cost of trust evaluation operations. The trust management model is adaptable to the role of each network node and its computational capabilities, it can be configured to be lightweight in terms of the supported primitives and the communication requirements, and it enables the delegation of the trust evaluation operations from sensor nodes to pre-determined, more powerful, nodes.

4.1.1.2 Support for the Heterogeneity in the Trust Evaluation Needs

WSNs may be deployed for different purposes, belong to application domains, and have diverse trust evaluation requirements. Even in a single deployment, diversity may exist in the trust evaluation requirements of the nodes, with some having very restricted role and being required to cooperate only with a limited, known set of other parties (like their cluster head or gateway), and some being more dynamic and regularly having to
evaluate and cooperate with unknown nodes (like the cluster heads of mobile clusters). Moreover, depending on the application domain of the deployments, diversity may exist in the level of distrust that the nodes should exhibit during the network lifetime towards unknown parties. The trust management solution should be able to support the heterogeneity between the deployments and the nodes in terms of these trust evaluation requirements, and provide the means for controlling trust evolution according to the level of distrust that the nodes should exhibit.

**Approach**  
The heterogeneity between the nodes and the deployments is supported by the model through a rich set of configuration parameters that make it adaptable to different scenarios. These parameters enable the trust evaluation needs of each node and deployment to be expressed and to affect both the trust evaluation mechanisms that will be used by the node and the way its trust relationships shall evolve during its lifetime. At the same time, they enable expressing the distrust that the nodes should exhibit towards unknown parties. Through the configuration parameters, the model can support from restricted nodes that should only trust the nodes that they are pre-configured to trust, to highly adaptive or mobile nodes.

### 4.1.1.3 Utilisation of the Pre-Deployment Knowledge

Unlike the general case of ad-hoc networks, real-world WSN deployments are purpose-specific, and the network purpose is known before deployment. Although they can be distributed, dynamic and mobile networks, in practice they are not composed of random autonomous nodes. The sensor nodes do not randomly become members in any deployment; each node has an ownership and purpose, which can be identified before deployment. Some nodes may be organised in, possibly hierarchical, clusters, while others may be randomly deployed in flat, dynamic, unstructured topologies. Even in the latter case, each node has some purpose and role that justifies its membership in the network. Some sensor nodes, for example the ones of a body sensor network, may even be clustered by deployment so that the trust relationships within the cluster can be preconfigured and be considered as long-term and stable. The requirement that is set for the trust management model is thus to exploit the pre-deployment knowledge that may exist on the roles of the network nodes and their trust associations, enable the pre-configuration of trust relationships, and allow for some control over their evolution according to the network purpose.

**Approach**  
The first step towards enabling the utilisation of the pre-deployment knowledge on the role and the trust associations of the nodes is enabling them to be expressed. The trust management model enables expressing this knowledge as pre-established, role-based trust associations. This representation was selected for the flexibility that it offers: It enables the definition of not only the trust associations, but also of their strength and their transitivity properties. A role-based trust association in the model can represent all functional trust, referral trust, and trust for assuming specific roles like the trust managing authority role or the supervision node role. However, some role-based trust associations may not be known before deployment. Consider the case of a vehicular sensor network (VSN) deployed by the vehicle manufacturer to provide information when the vehicle enters within the range of the environmental
4.1. SCOPE AND OBJECTIVES

sensor networks (ESN) deployed in authorised service points. It would not be possible to pre-configure the VSN with the trust associations for all possible ESNs. The model enables the use of certificate-based trust evidence to benefit from the flexibility that certificate-based trust establishment offers. The ESN nodes of the example could provide a certificate signed by the vehicle manufacturer’s offline trust managing authority, with which the VSN nodes have a pre-established role-based trust association. Through enabling these two types of trust associations, the model not only utilises the pre-deployment knowledge that may be available, but also enables the definition of predefined clusters, provides the means for restricting the set of external trusted parties of these clusters, and allows for control on trust evolution.

4.1.1.4 Trust Revocation due to Malicious Behaviour

WSNs are susceptible to node misbehaviours due to the potential for node compromise and to the resource limitations that motivate selfish behaviour. The trust management solution should be able to take into account the behaviour of the nodes and to revoke trust from nodes that exhibit malicious behaviour. It should therefore provide the means for information about node misbehaviours to be efficiently spread in the network, and for the trust values to be re-evaluated.

**Approach** The identification of malicious or selfish nodes and the evaluation of trust according to behavioural evidence requires behaviour monitoring and evaluation mechanisms which, as explained in requirement 4.1.1.1, was excluded from our model. In order to receive the benefits of behaviour-based trust evaluation with minimal cost in resources, the model enables the utilisation of behaviour-based trust evidence that is provided in the form of recommendations from a subset of designated nodes or from supervision networks. These perform monitoring and evaluation operations independently, and propagate their trust evaluation results both on request and proactively, when a node is found to be misbehaving. On the basis of negative behavioural evidence, a trust re-evaluation process is included in the model to revoke trust from accused nodes.

4.1.1.5 Robustness Against Attacks on the Trust Management Model

The trust management model needs to be robust against attacks aiming to tamper with the trust evaluation process and either revoke trust from legitimate nodes or assign high trust values to malicious nodes. In the defaming attack of the first category, the attacker spreads false negative evidence about legitimate nodes, or pretends to be victim of a defaming attack to make legitimate nodes appear malicious [SP04]. The second category includes the on-off attack, where malicious nodes behave correctly and badly alternatively, in order to remain undetected, the conflicting behaviour attack, where malicious entities behave well only to a subset of the network nodes in order to gain some positive recommendations, and the sybil attack, where malicious nodes create several fake identities to share the blame for their bad behaviour [SHL08]. These types of attacks can exploit the trust management model, and result in the isolation of legitimate nodes and the disruption of the network operations. It is therefore crucial that the model provides the means for protection against them.
Approach The trust management model combines strong security mechanisms for certificate-based trust evidence with the exchange of recommendations for role-based trust evidence and for the propagation of behaviour evaluation results. The combination of different types of evidence, the first of which can not be tampered with through the aforementioned attacks, offers it increased robustness in comparison to the purely behaviour-based models. This characteristic, however, does not suffice in protecting the trust values from false positive or negative trust evidence. For this reason, we opted for controlled trust establishment and revocation: Trust evaluation and revocation operations are executed only when a predetermined amount of evidence has been accumulated, and the recommendations that are utilised as evidence need to originate from third parties that are trusted as recommenders. However, increased robustness against attacks inevitably increases the cost for accumulating trust evidence. To address the issue of increased resource consumption, the model provides the means to configure the amount of required evidence before deployment for each node, according to its criticality, capabilities and trust evaluation requirements.

4.1.1.6 Generic Scope and Utilisable Results

The evaluation and maintenance of the trust associations between the network nodes are not meaningful unless utilised by other security services, like data access control, secure routing, or trusted key exchange services. The model should be integrable as a component in a security solution, provide utilisable results, hide the trust management details from the other security subsystems or services, and be applied uniformly throughout the network. Moreover, it needs to be independent from trust semantics, for its results to be semantically appropriate for use by different security services.

Approach The trust management solution is designed to operate as an independent security service that can be integrated in a security solution and provide interfaces to other subsystems for requesting the functional trust metric for any network node. All trust establishment, maintenance and revocation operations that need to be executed to provide this result are performed internally. For the solution to remain generic, the semantics of trust are not specified, and the function that functional trust represents is not instantiated. Instead, the trust management processes are designed to handle trust evidence that is semantically consistent and dependant on the context for which the trust model is used. Semantic consistency can be achieved for role-based and certificate-based trust evidence, if the role-based trust associations are established and the certificates are issued according to the context for which the trust model is used, e.g. to represent trust for data access operations. This is does not, however, apply to the behaviour-based trust evidence, which is not semantically independent. Behaviour-based solutions implicitly assign strict semantics to trust: functional trust is related to routing decisions, since all the trust evidence utilised to evaluate it are related to routing behaviour. If, however, the trust management model is utilised for data access operations, a sensor node should not be lead to disclose its data to an unknown node only because it has demonstrated good routing behaviour. For this reason, the model utilises the results of behaviour monitoring only as the means to identify malicious nodes and, once a trust association has been established using a combination of semantically consistent trust evidence, it does not allow trust to increase as the result of good
4.1. SCOPE AND OBJECTIVES

behaviour.

4.1.2 Relation to Other Approaches

The proposed trust management model is not purely certificate-based nor behaviour-based, but hybrid. Unlike the hybrid solutions that have been proposed for distributed systems or multiagent systems, the model is targeted to WSNs, and the requirements that it addresses are substantially different. It uses the concept of role-based trust like the FIRE model for multi-agent environments [HJS06], but (1) role-based trust is represented by trust associations and is spread as recommendations instead of utilising context-based rules, (2) is more conservative on the required infrastructure and evidence for trust evaluation, (3) enables the adaptation of the trust evaluation operations according to the network pre-configuration to support the heterogeneity in the capabilities and the trust requirements of the nodes. Moreover, unlike other hybrid solutions ([BDOS05, LV07]), the model defines the processes and metrics for accumulating the different types of evidence and for combining them in composite trust values that can be utilised by other services.

Similarly to some of the hierarchical trust solutions discussed in Section 3.1.3.1, the trust management model utilises offline certification authorities to address the problems of varying connectivity and of the lack of online management points. Davis [Dav04] also utilises certificates signed by offline certification authorities, whose public keys are maintained locally by the trustors to verify the signatures. Unlike the solution proposed in reference [VOT01], the model uses offline certification authorities but, in order to remain generic, it does not define the semantics of the certification process, and does not utilise the concept of trust negotiation, since the incremental exchange of certificates would be too resource consuming for the sensor nodes.

The model is related to behaviour-based solutions, since it utilises their results once they are provided as an independent network service by trusted, dedicated supervision deployments. Unlike the behaviour-based models that were discussed in Section 3.1.4, the model does not include any watchdog mechanism for behaviour monitoring by the sensor nodes. Because of this, the trust evaluation model can not utilise as evidence the behaviour records of other nodes (like the number of positive or negative experiences), and it does not model trust using the statistical approaches that the purely behaviour-based solutions utilise. Instead, it utilises the final results produced by the behaviour-based solutions, and combines them with different types of evidence to benefit from the pre-deployment knowledge that may be available.

4.1.3 Assumptions and Limitations

The driving factor for most of the requirements that are set for the model and for the main design decisions is that real-world WSNs are purpose-specific. Unlike the behaviour-based models that have been proposed for WSNs, this work does not assume a theoretical scenario where the sensor network is purely ad hoc, unstructured, and the nodes are completely autonomous. Instead, it assumes a WSN deployment that has a purpose and an ownership, irrespectively of how complex, dynamic and large the network is. Because of this, a basic assumption that we have made is that there is some pre-deployment knowledge, at least for the roles or network membership of some
of the nodes, and this knowledge can be utilised. Although this assumption is made, the model can be configured to operate without pre-deployment knowledge and use only the input from behaviour-based trust evaluation as evidence; however, this is not the purpose or scenario it was designed for.

Moreover, considering that the resource limitations and the purpose-specific nature of sensor nodes hinder the use of node behaviour monitoring and evaluation mechanisms, the trust establishment model does not include such functionality for the sensor nodes. It enables the use of behaviour evaluation results provided as an independent network service, but it is out of the scope of our work to specify the mechanisms used by this service for monitoring node behaviour and detecting malicious nodes and defaming attacks. This work does not therefore include any mechanisms for behaviour monitoring and for identifying malicious behaviour, or any behaviour-based trust evaluation model. Instead, we specify which existing behaviour-based solutions the model can be integrated with and receive their feedback.

It should also be noted that the proposed model implicitly assumes the existence of authentication and integrity protection mechanisms. Tasks in the trust establishment processes like the exchange of trust evidence require explicit message protection mechanisms, in order to defend against attacks aiming to tamper with the values of the metrics. Fulfilling this assumption relates to the positioning of the trust management components within a complete security solution. In Chapter 6, trust management is integrated as an independent security service in a security solution, which includes cryptographic components for the provision of authentication and integrity protection services.

4.2 Working Scenario

A generic example scenario that we assume and later use for explaining the parameters, processes and metrics of the trust management model consists of a WSN composed of one hundred nodes. As shown in Figure 4.1, which depicts the logical grouping of the nodes, the scenario also includes an offline certification authority (CA) and a supervision network (SvN) of one hundred nodes. The SvN nodes are assumed to perform behaviour monitoring and evaluation using some behaviour-based trust management model, and provide their results both on request and proactively when they identify node misbehaviours. A separate SvN deployment is assumed instead of designated WSN nodes for providing this service, to enable distinguishing and examining the interactions between the sensor nodes and the supervision nodes.

This setting was selected because it can combine all types of trust evidence that the
model utilises, including certificates, locally stored information on pre-established trust associations, and recommendations from third parties and supervision nodes. However, in this basic scenario we do not assume any pre-deployment knowledge on the roles and the trust evaluation requirements of the nodes, and we do not specify their configuration or any role-based trust associations. This scenario shall be gradually enriched in the following sections where we introduce the various aspects of the model that affect it.

In order to validate the proposed model and the algorithms and metrics that it includes we implemented a simulation environment in Java. It receives as inputs a representation of the network and its initial configuration, along with a representation of how the simulation will proceed and of what cooperation request messages will be sent towards the WSN nodes. The trust establishment processes were implemented, so that the simulator can accumulate and output information on how the trust relationships have evolved and on what operations were executed by each node.

The simulation executes in simulation rounds. For the working scenario, in each round cooperation request messages are sent towards random WSN nodes. The number of the requests in each round is 100, equal to the size of the WSN. The sender of each request message is also randomly selected from the WSN or the $SvN$, to simulate the effect of the dynamically changing physical position of the nodes. With the receipt of each request message, the recipient resolves its trust relationship with the sender and, if the sender is unknown, initiates the trust establishment process.

The simulation involves three phases. In the first phase, for 200 rounds, we assume that the supervision nodes have not gained enough experience from the behaviour of the WSN nodes, so they can only provide recommendations for the rest of the $SvN$ nodes. The second phase proceeds like the first for an equal number of rounds, but the $SvN$ nodes are assumed capable of providing recommendations for the WSN nodes. In both phases, the recommendations are provided only on demand, as responses to recommendation requests generated during the trust establishment process. To examine separately the effects of the trust establishment and of the trust re-evaluation operations, recommendations are sent proactively only during the third phase, for the next 400 rounds, when the simulation focuses on trust re-evaluation due to the propagation of negative behaviour evaluation evidence from the $SvN$ nodes. We assume that 10% of the $SvN$ nodes are malicious and they collude on a defaming attack.

### 4.3 Hybrid Trust Management Overview

The discussion of the trust management requirements and our approach for addressing them highlighted that the model needs to be adaptable to the varying capabilities and trust requirements of the nodes, yet flexible in the types of trust evidence that it utilises. For these reasons, the model enables the exploitation of pre-deployment knowledge in order to adjust the supported trust characteristics for each node, and combines aspects from certificate-based and behaviour-based approaches on common evaluation processes and metrics, to benefit both from the representation of pre-established trust relationships as certificates and from the continuous behaviour-based evaluation of trust.

---

1 The source code of the simulator, along with the network configuration files and simulation execution configuration files that were used for the experimental evaluation of Section 4.8 are available at http://www.icsd.aegean.gr/infosec_base/download/HybridTMSimulator.zip
Table 4.1: Types of trust evidence combined in the proposed model

<table>
<thead>
<tr>
<th>Trust type</th>
<th>Motivation</th>
<th>Trust representation</th>
<th>Requirement addressed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role-based</td>
<td>Representation of pre-deployment knowledge, traceability, control on trust evolution, resistance to trust saturation and low trust valuation problems [LV07].</td>
<td>Pre-established trust association.</td>
<td>4.1.1.3: Expressing the pre-deployment knowledge on the trust associations of the nodes, 4.1.1.5: Increasing robustness against attacks through the combination of different types of evidence, 4.1.1.6: Sustaining the generic scope of the model through a semantically independent type of trust.</td>
</tr>
<tr>
<td>Certificate-based</td>
<td>Resilience on strong security mechanisms, flexibility, availability for new nodes when no other type of trust can be utilised.</td>
<td>Certificate issued by trusted offline CA.</td>
<td>4.1.1.3: Allowing for flexibility in the representation of pre-deployment knowledge, 4.1.1.5, 4.1.1.6, 4.1.1.1: Dealing with the absence of online trust managing authorities.</td>
</tr>
<tr>
<td>Behaviour-based</td>
<td>Feedback on malicious behaviours, distributed, gradual and cooperative trust revocation.</td>
<td>Recommendation from trusted supervision node.</td>
<td>4.1.1.4: Enabling trust revocation due to malicious behaviour.</td>
</tr>
</tbody>
</table>

The model combines three types of trust evidence: role-based, certificate-based, and behaviour-based. Table 4.1 summarises the motivation for including these types of trust in the model and the requirements that are addressed. Moreover, it highlights the fact that the different types are not viewed as alternative, but as complementary, and that the hybrid approach is undertaken to leverage the drawbacks associated with the individual approaches. In the following sections we provide an overview of the model and introduce its most basic aspects: the hybrid trust associations, the admissible types of trust evidence, and the configuration parameters.

### 4.3.1 Trust Associations and Metrics

The trust associations and their elements need to respect the fundamental properties of trust as discussed in Section 3.1.1. Regarding trust transitivity and the construction of valid trust paths, certain constraints were set: functional trust, representing trust for the particular function that the model is utilised for, needs to be explicitly distinguished from referral trust, representing trust for referring other entities. We thus define separate metrics and control mechanisms for functional trust and referral trust, to distinguish between trust in providing a service, from trust in providing a recommendation for a node providing that service.

The functional trust that a trustor node $i$ assigns to a trustee node $j$ is represented by $T_{ij} \in [0, 1]$, while referral trust by $R_{ij} \in [0, 1]$. Functional trust is context-specific; for each deployment of the model, the scope or semantic content of functional trust needs to be the same for all nodes and all types of trust evidence in the deployment. Moreover, from all elements that are included in a trust association and listed in Table 4.2, functional trust is the only metric that is utilisable by and published to the client security services of the trust manager.

Apart from the functional and the referral trust metrics, each trust association
4.3. HYBRID TRUST MANAGEMENT OVERVIEW

Table 4.2: Elements of the trust associations of node \( i \) with any node \( j \)

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_{ij} \in [0,1] )</td>
<td>Functional trust value of ( i ) for ( j )</td>
</tr>
<tr>
<td>( R_{ij} \in [0,1] )</td>
<td>Referral trust value of ( i ) for ( j )</td>
</tr>
<tr>
<td>( R_{ij}^{\text{total}} &gt; 0 )</td>
<td>Total referral trust for ( j )’s recommenders</td>
</tr>
<tr>
<td>{BEvidence}_j</td>
<td>Set of temporarily stored recommendations by supervision nodes for ( j )</td>
</tr>
<tr>
<td>( CA_j )</td>
<td>Flag denoting if ( j ) is considered a Certification Authority by ( i )</td>
</tr>
<tr>
<td>( Supervision_j )</td>
<td>Flag denoting if ( j ) is considered a supervision node by ( i )</td>
</tr>
</tbody>
</table>

includes \( R_{ij}^{\text{total}} \), which is calculated during trust establishment and represents the total referral trust of the nodes that implicitly or explicitly provided the trust evidence for \( j \). Its value essentially indicates the strength of the evidence that was utilised for the calculation of \( T_{ij} \) and \( R_{ij} \). Whenever a new piece of evidence becomes available, this metric determines how much it will affect the current trust values. For the role-based trust associations that are configured before deployment, the higher the \( R_{ij}^{\text{total}} \) is set, the more difficult trust revocation shall be for \( j \) in terms of the required negative evidence.

The model utilises the results of behaviour-based trust evaluation performed by supervision nodes in order to receive feedback on malicious behaviours. The pieces of negative behavioural evidence that become available for node \( j \) are denoted by \( BEvidence_j \). This evidence is in the form of negative recommendations received from supervision nodes (their contents are further explained in Section 4.4). When \( BEvidence_j \) is received by \( i \) from a trusted supervision node about a node \( j \) that already exists in the trust records of \( i \), then it is temporarily stored in the set \( \{BEvidence_j\} \), so that trust can be revoked when enough negative evidence exists in the set. The set is emptied after each trust revocation process. The decision on storing the negative evidence instead of proceeding to trust revocation after each accusation is driven by two factors: First, it can increase the robustness of the model to defaming attacks that may be launched by supervision nodes against legitimate trusted nodes, since it enables filtering out of the set the accusations originating from the same nodes or from nodes whose trust has been revoked until the re-evaluation process. Second, it enables parameterising the amount of required negative evidence, and thus balancing the level of protection against defaming attacks with the delay incurred for revoking trust from malicious nodes.

The elements \( CA_j \) and \( Supervision_j \) take boolean values to denote if \( j \) is considered to be an offline certification authority or a supervision node. If \( CA_j = \text{true} \), then \( i \) should also locally maintain the public key of \( j \). The associations with certification authorities can only be role-based trust associations that have been manually configured; nodes that are unknown and their trust evaluated through a trust establishment process can not assume the role of certification authorities. Through this parameter, the model allows controlling the certificate validation operations that the nodes will perform: if no relationships are specified with certification authorities, then the node shall never request or accept certificates as trust evidence.

The requirement that was set for the model about the utilisation of the pre-deployment knowledge on the trust associations of the nodes can be realised through adding role-based trust associations in the trust records before deployment. The initial configuration of each node includes assigning values \( T_{ij} \) and \( R_{ij} \) for the nodes \( j \) that node \( i \)
CHAPTER 4. A HYBRID TRUST MANAGEMENT MODEL

Figure 4.2: Working WSN scenario: Adding pre-deployment knowledge on trust associations

should trust and receive recommendations from. Through the corresponding flags of Table 4.2, trust relationships can also be defined for the offline trust managing authorities that i will accept certificates from, and for the supervision nodes that it will accept behaviour-based evaluations from.

The model can be configured to operate as purely certificate-based if the only initial role-based trust associations are with certification authorities, and the trust degradation parameter is assigned a low value (explained in Section 4.3.3), so that the nodes will accept only certificates as evidence during their lifetime. However, for nodes that have strictly defined roles in the network or have limited computational capabilities, no relationships with certification authorities should be defined, in order to avoid the resource consuming certificate validation operations. Moreover, the model can operate using the results of behaviour evaluation alone for scenarios where the nodes should depend entirely on a trusted supervision deployment for providing them trust evidence. In this case, trust relationships should be defined before deployment only with some of the supervision nodes. This would however mean that the sensor nodes may have to wait for the supervision nodes to gain sufficient experience on the behaviour of their neighbours, in order to be able to provide their recommendations.

In the working scenario, an example configuration of the initial associations of the WSN nodes is shown in Figure 4.2 (a). This setting would imply that each WSN node fully trusts a specific offline certification authority and partially trusts a subset (10) of the nodes in the supervision network. Practically, the certification authority may be one that is owned or managed by the deployment owner, the supervision network may be an independent network offering its services to the WSN, and the 10 nodes can be selected as these that are within the range of each sensor node during the pre-deployment phase.

Through the pre-configuration of role-based trust associations and associations with certification authorities, the model not only utilises the pre-deployment knowledge that may be available, but also enables the configuration of predefined clusters, provides the means for restricting the set of external trusted parties of these clusters, and permits restricting the trust operations of the nodes. If the initial configuration of the example scenario was as in Figure 4.2 (b), and included only associations between the WSN nodes and no associations with other parties, then during the WSN lifecycle its nodes would not be able to expand their associations and cooperate with any node outside the cluster, and would not perform any certificate validations. The model thus allows for control on the trust operations and evolution for settings where strict limitations
4.3. HYBRID TRUST MANAGEMENT OVERVIEW

should apply.

4.3.2 Trust Evidence

The types of evidence that are utilised for trust establishment are these included in Table 4.1. The model uses certificates signed by offline trust management authorities by a subset of the network nodes. Certificate validation is performed locally, provided that each node stores the public keys of the trust management authorities that it is pre-configured to trust. Consequently, trust associations can be established between nodes that are associated with common trust management authorities.

The trust establishment process also supports the involvement of third parties for the provision of recommendations. Third parties are other trusted network nodes, that may be sensor nodes, cluster heads, gateways, or supervision nodes. Trust transitivity is controlled; the third parties have to be trusted as recommenders. The model also includes support for the explicit degradation of referral trust according to the distance from pre-established trust relationships.

Behaviour-based trust evaluation can be performed only by a subset of designated supervision nodes or by independent supervision networks, that propagate their behaviour evaluation results to the WSN. The supervision nodes are designated to monitor the network traffic, evaluate the nodes within their range according to their behaviour in network and data related operations, and make their evaluations available as recommendations. The propagation of these behaviour evaluation results is performed both on demand, as responses to recommendation requests, and proactively, when negative evaluation results are computed for a node that was found to be misbehaving. To defend against defaming attacks from malicious recommenders, trust revocation is a controlled process: the number of accusations that must be accumulated by supervision nodes for trust to be revoked depends on the referral trust of the individual recommenders.

The trust management model enables the use of behaviour evaluation results provided as an independent network service, but does not specify the mechanisms used by this service for monitoring node behaviour and detecting malicious nodes and defaming attacks. Instead, it can be integrated with most of the behaviour-based schemes that were discussed in Section 3.1. Specifically, it can be integrated with those that can formulate opinions that include both functional and referral composite trust metrics ([Li07, ZMHT06]) instead of trust-confidence pairs ([TB06]) or subjective logic 4-tuples ([KB06]). The specific mechanisms that are used for behaviour monitoring depend on the scheme that is used, and may vary from mechanisms similar to the watchdog [GBS08, MGLB00], to mechanisms estimating the validity of the data reported by the sensor nodes [RPGH08].

The parties that may be involved in the trust establishment process are offline trusted third parties whose public key is locally stored for signature verification, other network nodes, and supervision nodes that perform behaviour-based trust evaluation. The trust associations between any trust issuer $i$ and any trust target $j$ that the model supports are based on combinations of the following types of trust evidence:

1. The locally stored information of $i$ on the role-based trust associations that were established prior to deployment.
Table 4.3: Trust establishment evidence and validation for trust relationship between any $i, j$

<table>
<thead>
<tr>
<th>Evidence type</th>
<th>Evidence elements</th>
<th>Evidence validation &amp; acceptance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role-based / Hybrid</td>
<td>$Rec_{xj}$ received from neighbouring or supervision node $x$</td>
<td>Stored $R_{ix} \geq R_{threshold}$, $T_{xj}$, $R_{xj}$ used in the evaluation</td>
</tr>
<tr>
<td>Certificate-based</td>
<td>$Cert_{xj}$ received from $j$ issued by offline CA $x$</td>
<td>Stored $R_{ix} \geq R_{threshold}$, stored public key of $x$, valid $Cert_{xj}$, $T_{xj} = 1$, $R_{xj} = 1$ used as a recommendation in the evaluation</td>
</tr>
<tr>
<td>Behaviour-based</td>
<td>$Rec_{xj}$ received from supervision node $x$ for known $j$</td>
<td>Stored $R_{ix} \geq R_{threshold}$, $Supervision_{x} = true$, $T_{xj} &lt; T_{ij}$ or $R_{xj} &lt; R_{ij}$, $T_{xj}$ or $R_{xj}$ used in the re-evaluation</td>
</tr>
</tbody>
</table>

2. The valid certificates that $j$ can provide to $i$ and $i$ can verify using the stored public keys of offline trust management authorities that it has a trust association with.

3. The recommendations received for $j$ upon request by third parties that $i$ has a trust association with.

4. The results of behaviour-based trust evaluation made available by supervision nodes that $i$ has a trust association with.

In order to formulate common processes and metrics for these different types of trust evidence, we view all types as recommendations: The first two are implicit recommendations from the network owner and the trust managing authorities, while the remaining are explicit ones. Table 4.3 lists the types of supported trust evidence. Once the trust relationship of node $i$ with node $j$ needs to be determined, one or more of the options on Table 4.3 are used for the required evidence to be accumulated according to the processes described in Section 4.5.

For the trust evidence to be semantically coherent, both the implicit and the explicit recommendations must refer to the same function. Depending on the context in which the model is used, the trust managing authorities certify not only the identity of the nodes, but their authorisation to be members of a particular network for a particular function, for example to route messages, to receive or send sensor readings, etc. For example, if the model is used for message routing decisions, then these semantics will be assigned to the functional trust recommended either through a certificate or from a third party, while the behaviour evaluation results will concern the message routing behaviour of the monitored nodes.

Once a trust relationship has been established, it can be revoked only on the basis of negative evidence received from trusted supervision nodes. The results of behaviour-based trust evaluation can only decrease the trust value after its initial computation. As explained in Requirement 4.1.1.6, this decision was made because trust evolution should respect the network pre-configuration, and good behaviour should not compensate for
4.3. HYBRID TRUST MANAGEMENT OVERVIEW

Table 4.4: Notations for the configuration parameters of sensor node $i$

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{\text{thresh}} \in [0,1]$</td>
<td>Threshold value for the functional trust metric</td>
</tr>
<tr>
<td>$R_{\text{thresh}} \in [0,1]$</td>
<td>Threshold value for the referral trust metric</td>
</tr>
<tr>
<td>$d_i \in [0,1]$</td>
<td>Trust degradation parameter</td>
</tr>
<tr>
<td>$R_{\text{req}}$</td>
<td>Required total referral trust of evidence providers</td>
</tr>
<tr>
<td>${\text{Propagation}}_i$</td>
<td>Set of nodes that $i$ will resolve requests from if asked for recommendations about unknown nodes</td>
</tr>
<tr>
<td>$\text{Supervision?}$</td>
<td>Flag denoting if $i$ is a supervision node</td>
</tr>
<tr>
<td>${\text{Cert}_{x,i}}$</td>
<td>Certificates issued by $x$ for $i$</td>
</tr>
</tbody>
</table>

the distrust that it was decided that a node should be dealt with during the pre-configuration phase. Moreover, behaviour-based trust evaluation models suffer from the trust saturation problem because of their resilience on the observed behavioural evidence \cite{LV07}. For the supervision nodes, any node will be considered fully trusted after a long history of positive behaviour, irrespectively of the deployment needs or the restrictions that should be applied. For this reason, the model enforces the use of the available evidence that the network owner can control (one of the first three types in the list) for building trust. This way, we avoid trust being built solely on good behaviour evidence for nodes for which no other positive evidence exists.

In the working scenario, through the configuration of the role-based trust associations as in Figure 4.2 (a), different types of trust evidence shall be communicated: Since the WSN nodes trust and maintain the public key of $CA$, any certified node $j$ can provide its certificate $\text{Cert}_{CA,j}$ as trust evidence. Moreover, since role-based trust associations have been defined with $10$ $SvN$ nodes, these can provide recommendations $\text{Rec}_{x,j}$ for any node $j$ whose behaviour they have evaluated.

4.3.3 Configuration Parameters

The proposed model can uniformly support, through proper configuration, nodes with highly diverse capabilities and needs. The initial configuration of each node is made not only to control trust evolution during the network lifecycle, but also to distribute the computational cost of the trust evaluation operations, to control the supported trust characteristics, and thus to control the resource overheads that the trust establishment operations will impose. This is achieved though the pre-configuration of the trust associations, which was discussed in Section 4.3.1 and the set of configuration parameters of Table 4.4. The following paragraphs introduce the parameters, explain their rationale, and provide the guidelines for their use.

Figure 4.3 illustrates the working scenario enriched with an example configuration of the WSN nodes. All 100 of the nodes have the same configuration, apart from the certificates issued by the $CA$ for the 10 $WSN_c$ nodes.

4.3.3.1 Trust Thresholds

The functional trust threshold $T_{\text{thresh}}$ and the referral trust threshold $R_{\text{thresh}}$ are used to classify the trust associations between any $i$ and $j$ according to the functional trust
CHAPTER 4. A HYBRID TRUST MANAGEMENT MODEL

value $T_{ij}$ and the referral trust value $R_{ij}$ into four classes, as illustrated in Figure 4.4. The classification according to the functional trust value is not utilised within the trust management system, since its semantics are application-specific; $T_{\text{thresh}}$ is only used in order to respond with the class of the trust target to application requests. $R_{\text{thresh}}$, on the other hand, is a determinant factor for trust evolution: It represents the minimum referral trust value that should be assigned to $j$ for its recommendations about other nodes to be requested and accepted.

The referral trust threshold parameter should be set according to the level of distrust that each node should exhibit after deployment towards unknown parties. The value of $R_{\text{thresh}}$, along with the configuration of the initial role-based trust associations, determines which nodes will be used as recommenders, and from which authorities the issued certificates shall be accepted from. In the working scenario, if $R_{\text{thresh}} = 0.5$, then the nodes in the setting of Figure 4.3 will accept both recommendations from all known nodes and the certificates issued from the CA.

The $R_{\text{thresh}}$ parameter is used to address the requirements 4.1.1.5 for robustness against attacks on the trust management model through enabling the rejection of the recommendations from nodes that are not considered to be honest recommenders, and 4.1.1.2 for supporting the heterogeneity in the trust evaluation needs through configuring the level of distrust. Moreover, the use of a separate threshold for referral trust implicitly addresses requirement 4.1.1.1 for controlling the computational cost of trust.
4.3. HYBRID TRUST MANAGEMENT OVERVIEW

Figure 4.5: Working WSN scenario: Varying the degradation parameter

(a) Node WSN₉, Round 400, \( d_i = 1 \)

(b) Node WSN₉, Round 400, \( d_i = 0.5 \)

evaluation operations: It allows the configuration of role-based trust associations with nodes that are trusted, but should not be used as recommenders. An example scenario where this could apply is a body sensor network that includes a computationally more powerful cluster head. The sensor nodes could be configured to have functional trust above \( T_{\text{thresh}} \), in order to cooperate for data or routing related operations. At the same time, the referral trust value of the sensors could be above \( R_{\text{thresh}} \) only for the cluster head, so that it would be the only node responsible for handling all recommendation requests and expanding the trust associations of the cluster.

4.3.3.2 Trust Degradation Parameter

It is within the properties of trust that it naturally degrades through transitivity. As explained in Section 3.1.1, the link operator for constructing transitive trust paths must satisfy this property. However, although an appropriately selected link operator can guarantee the degradation of trust, it is not adaptable to deployments with diverse trust evaluation needs. The rationale for utilising an explicit trust degradation parameter in the model is to enable configuring how much trust should degrade through transitivity.

The trust degradation parameter \( d_i \) determines how much the referral trust will degrade for unknown nodes. It is used to address the requirement 4.1.1.2 for supporting the heterogeneity in the trust evaluation needs of the nodes and the deployments, and specifically for expressing the distrust that the nodes should exhibit towards unknown parties. When \( d_i = 1 \), no explicit degradation is performed to referral trust apart from the natural degradation enforced through the link operator. When \( d_i = 0 \), then the referral trust \( R_{ij} \) assigned to any unknown node \( j \) during trust establishment will be set to 0, essentially indicating that, during the lifetime of \( i \), it shall accept recommendations only from the nodes it has pre-configured role-based trust associations with. \( d_i \) should therefore be set according to the node’s level of distrust and, together with the \( R_{\text{thresh}} \), it eventually determines the maximum allowed distance from pre-established trust relationships.

In the working WSN scenario, configured as in Figure 4.3 with \( d_i = 1 \), the trust relationships established by one of the WSN nodes after 400 simulation rounds are
depicted in Figure 4.5 (a). Through the validation of certificates from the trusted CA and the request for recommendations from the 10 trusted SvN nodes, most of the nodes (the ones not accused of misbehaviours) are classified as trusted honest nodes. Executing another simulation experiment for the same node with \( d_i = 0.5 \), the referral trust assigned to all nodes, except the ones that were known before deployment, is below the threshold, as shown in Figure 4.5 (b). This node will therefore keep relying on its role-based trust associations for the provision of recommendations, although it trusts the rest of the nodes. This allows for a greater effect of the pre-deployment knowledge on trust evolution, and would be appropriate for nodes or deployments that are relatively static or security-critical, and should rely more on their role-based pre-established trust associations.

4.3.3.3 Total Referral Trust

The required total referral trust parameter \( R_{req} \) represents the amount of trust evidence that needs to be accumulated for a trust relationship to be established or revoked. It is the parameter that is compared whenever a new piece of evidence becomes available with the total referral trust \( R_{ij}^{total} \) of the nodes that have implicitly (in the case of offline certification authorities) or explicitly provided the trust evidence for the trustee \( j \). For unknown nodes to be able to resolve their trust association after deployment, they need to have or to be able to establish relationships with common trusted recommenders with a total referral trust of more than \( R_{req} \). Therefore, \( R_{req} \) essentially indicates the required strength of the trust evaluation evidence, based on the referral trust of the direct or indirect recommenders.

\( R_{req} \) is used as an alternative to the required number of recommendations for a trust relationship to be established. The reason for opting for \( R_{req} \) is that not all recommenders are equally trusted (in terms of their referral trust), and thus should not contribute the same value towards deciding when sufficient evidence has been accumulated.

The value of \( R_{req} \) should be set according to the trust or distrust that a node should exhibit. Security-critical and distrustful nodes should require greater amounts of evidence before they make any trust decisions. However, at the same time, the resource limitations of the nodes should be taken into account. For nodes that have limited resources and computational capabilities \( R_{req} \) should be relatively low, since it affects the resource consumption that trust establishment imposes by determining the required number of messages to be exchanged and the number of certificate validations to be performed.

In the working scenario configured as in Figure 4.3, the value of \( R_{req} \) is set to 1, equal to the referral trust that the WSN nodes assign to the CA and less than the referral trust that they assign to the supervision deployment nodes. As a result, a valid certificate from the CA will suffice as evidence for trust evaluation, while more than one recommendations would be required by the SvN nodes. If the value of \( R_{req} \) was greater, then the WSN nodes would have to combine the evidence originating from the CA and the SvN nodes in order to accumulate enough \( R_{ij}^{total} \) for any \( j \). To examine this, we have run a simulation using the configuration of Figure 4.3 and executing the three phases described in Section 4.2, and then run the simulation having set \( R_{req} = 2 \). Figure 4.6 shows the number of trust associations of each type collectively for all WSN nodes. During the first phase of the simulation, when the SvN nodes do not yet provide
4.3. HYBRID TRUST MANAGEMENT OVERVIEW

Figure 4.6: Working WSN scenario: The effect of the total referral trust parameter on the trust associations

recommendations for the WSN nodes, a greater number of relationships is resolved when $R_{req} = 1$, because the certificates suffice as evidence. At round 400, however, the system appears significantly more vulnerable to the defaming attack, since only 2 malicious accusations suffice for trust revocation. This demonstrates that a higher value of $R_{req}$ causes greater delays for evidence accumulation but, at the same time, it increases the robustness against defaming attacks.

The $R_{req}$ parameter is used to address the requirements [4.1.1.1] for application to WSN deployments, since it enables restricting the required evidence and allows the lightweight configuration of the model, and [4.1.1.2] for supporting the heterogeneity in the trust evaluation needs, since it expresses the distrust that the nodes should exhibit. More importantly, it relates to requirement [4.1.1.5] for robustness against attacks on the trust management model, since it enables configuring the strength of the required evidence for trust revocation and, as a result, affects the robustness against defaming attacks.

4.3.3.4 Propagation Set

Our approach for addressing Requirement [4.1.1.1] for application to WSN deployments included exploiting the heterogeneity of the nodes for distributing the cost of trust evaluation operations and enabling the delegation of resource consuming operations from sensor nodes to other pre-determined, more powerful, nodes. In settings where such nodes exist and pre-deployment knowledge on the network structure is available, the $\{\text{Propagation}\}$ set is used to express these delegation operations.

The $\{\text{Propagation}\}$ set of any node $i$ includes the identifiers of the nodes that $i$ is obliged to resolve requests from if asked for recommendations about other unknown nodes. Upon the receipt of a recommendation request from node $x$ about node $j$, if $i$ cannot provide its recommendation $Rec_{ij}$ because it does not have an already established trust relationship with $j$, the default case is to reply that its relationship with $j$ is unresolved. If, however, its $\{\text{Propagation}\}$ set includes the requestor node $x$, then $i$ is obliged to initiate the trust establishment process with $j$, try to resolve the
relationship and reply to the recommendation request.

Depending on the scenario and the pre-deployment knowledge on the network topology, this parameter can be used to enable the definition of clusters of sensor nodes that rely on cluster heads for expanding their list of trust associations. If the \{\textit{Propagation}\} sets include the child nodes of cluster heads, then these would be the nodes responsible for executing the resource consuming tasks of exchanging intra-cluster recommendations and validating certificates, in order to reply to the requests of their child nodes. This parameter thus enables controlling the distribution of the computational cost of trust evaluation operations between the network nodes, when the scenario enables the definition of clusters. Moreover, it enables limiting the computationally expensive operations at the network level: When a request is delegated to a cluster head, and it resolves its association with another node through certificate validations and intra-cluster recommendations, then the knowledge on its trust association can be spread to all its child nodes through inter-cluster recommendations only.

To illustrate the effect of the \{\textit{Propagation}\} set on the required operations for trust evaluation, we have run simulation experiments using the configuration of Figure 4.3 and then run the simulations having populated the \{\textit{Propagation}\} sets of the 10 WSN\textsubscript{c} nodes to include the rest of the WSN nodes. The results on the number of operations that were executed collectively by the 100 WSN nodes, averaged from 15 simulation runs for each parameterisation, are shown in Figure 4.7. When the propagation set is being used, the number of intra-cluster recommendations exchanged (recommendations requested and received from the \textit{SvN} nodes) is considerably lower, since the WSN\textsubscript{c} nodes were forced to resolve their trust associations following recommendation requests, and then spread this knowledge towards the rest of the WSN nodes whenever they were asked for inter-cluster recommendations.

The effect of the propagation set on the trust evaluation operations would be greater if further optimisations were made in the initial configuration of the nodes. An example
scenario where the pre-deployment knowledge could be exploited is a body sensor network where the cluster head $c$ is responsible for expanding the trust relationships of the cluster. For the sensor nodes only $R_{ic}$ should be set above the threshold, $R_{req}$ should less than $R_{ic}$, for $c$’s recommendation to suffice for the establishment of new relationships, and $d_i$ should be set to zero. The $\{\text{Propagation}\}$ set of the cluster head could then include the sensor nodes, and its initial trust associations could allow for more flexibility, according to its role and the computational capabilities. With this parameterisation, the distribution of the resource overheads is controlled, since the sensor nodes would never need to perform any operations except from requesting from their cluster head the recommendations for unknown nodes.

### 4.4 Structure, Components and Interfaces

The components of the trust management system are structured as in Figure 4.8. It is designed to operate as an independent security service. It utilises the network interface for receiving and responding to evidence requests, and provides an applications interface for making the trust ratings available to requesting applications and services.

The established trust associations are maintained as trust records, whose elements are the ones included in Table 4.2. The trust management operations include:

1. Publishing the results of the established trust associations upon request through the TRUST-QUERY interface. The primitives that are included in this interface are
the **TRUST-QUERY.req** used by the applications to request the trust information for specific nodes, and the **TRUST-QUERY.res** primitive, that delivers the requested information:

**TRUST-QUERY.req**(  
  ServiceID, /*The service requesting the trust information*/  
  TargetID /*The identifier of the trust target node*/  
)

**TRUST-QUERY.res**(  
  TargetID, /*The identifier of the trust target node*/  
  TrustClass, /*Discrete value in {trusted, distrusted, unresolved}*/  
  TrustValue /*The value of the functional trust metric*/  
)

The **ServiceID** is included in the request primitive to enable the trust management system to be used by various services that assign the same semantics to trust. Upon the receipt of a **TRUST-QUERY.req** request, the Trust Associations Resolution operation is invoked to provide the value of $T_{ij}$ and the class of the trustee. The classification is made with respect to the functional trust metric as in Figure 4.4. The functional trust value is also provided in the response, since some applications may require more detailed information or have their own classification mechanism —an example is the controlled data disclosure scheme presented in Chapter [5](#).

2. Resolving the trust associations with other nodes. This operation is invoked using the **TRUST-RECORD.req** primitive because of an application request, because of a trust evidence request which has been received from another node, or because of a request from the Trust Maintenance and Re-Evaluation operation. The operation executes the trust associations resolution process that is described in Section [4.5.1](#), and responds using the **TRUST-RECORD.res** primitive:

**TRUST-RECORD.req**(  
  RequestorID, /*The identifier of the node requesting the trust evidence*/  
  TargetID /*The identifier of the trust target node*/  
)

**TRUST-RECORD.res**(  
  TargetID, /*The identifier of the trustee node*/  
  TrustClass, /*Discrete value in {trusted, distrusted, unresolved}*/  
  TrustValue /*The value of the functional trust metric*/  
  ReferralValue, /*The value of the referral trust metric*/  
  Supervision, /*Boolean value indicating recommendation for a supervision node*/  
)
3. Establishing trust with unknown nodes. This operation is invoked using the
\texttt{TRUST-EST.req(TargetID)} primitive, executes the trust establishment process
that is described in Section 4.5.2, updates the trust records, and responds to the
requesting component using the \texttt{TRUST-EST.res} primitive, which contains all the
information from the newly inserted trust record.

4. Maintaining the trust records and re-evaluating and revoking trust from malicious
nodes. The operation is invoked after an accusation is received, and executes the
process described in Section 4.5.3 to update the corresponding \( \{BEvidence_j\} \)
field of the accused node in the trust records and, when required by the process,
to re-evaluate the trust values.

Apart from the core trust management operations, supportive operations are in-
cluded to handle the trust evidence requests and replies. These include operations
to:

1. Receive and respond to trust evidence requests. This operation is invoked in any
node \( i \) when a node \( j \) requests trust evidence either for another node \( x \), in which
case the Trust Associations Resolution operation is invoked to handle the request,
or for \( i \) itself, in which case \( i \) responds with the certificates that it may hold.

2. Request and receive trust evidence from the trust target node or from other nodes
during a Trust Establishment operation, and accusations sent proactively from
supervision nodes. In the case of trust evidence, the operation forwards all incom-
ing evidence to the Trust Establishment operation, while the accusations (when
the \texttt{EvidenceType} field of the \texttt{TRUST-EVIDENCE.res} primitive is \texttt{accusation})
are forwarded to the Trust Maintenance and Re-Evaluation operation.

All information that is communicated by these operations, including certificates,
recommendations, and accusations, is represented by the \texttt{Evidence} construct. The
\texttt{Evidence} represents either a recommendation / accusation, or a list of certificates:

\begin{verbatim}
(Recommendation)Evidence <
  NodeID, /*The identifier of the node providing the recommendation*/
  TargetID, /*The identifier of the trust target node*/
  TrustValue, /*The recommended functional trust value*/
  ReferralValue, /*The recommended referral trust value*/
  Supervision, /*Boolean value indicating recommendation for a supervision node*/
>

(Certificates)Evidence <
  TargetID /*The identifier of the certificate holder*/
  List <
    CAIdentifier, /*The identifier of the issuing authority*/
    Certificate
> >
\end{verbatim}
The Evidence is requested and communicated using the TRUST-EVIDENCE interface. This includes the TRUST-EVIDENCE.req primitive, which is used by the nodes to request trust evidence, and the TRUST-EVIDENCE.res primitive, which is used to communicate trust evidence or accusations:

TRUST-EVIDENCE.req(
    SenderID, /*The identifier of the node requesting the trust evidence*/
    RecipientID, /*The identifier of the recipient*/
    TargetID /*The identifier of the trust target node*/
)

TRUST-EVIDENCE.res(
    SenderID, /*The identifier of the node providing the trust evidence*/
    RecipientID, /*The identifier of the recipient*/
    EvidenceType /*Discrete value in {certificate, recommendation, accusation, none}*/
    Evidence /*The Evidence construct*/
)

The Evidence construct contains either a recommendation or a list of certificates, when the RecipientID is the same as the TargetID. It may be empty when the requested evidence is unavailable, either because the value of the TrustClass is “unresolved” in the TRUST-RECORD.res returned from the Trust Associations Resolution component, or because the node does not hold any certificates. It should be noted that the Trust Evidence Management operations do not affect the trust records, and they rely on the Trust Management operations for providing them with the requested evidence for other nodes.

The communication of the certificates that the nodes hold is performed upon request without applying any constraints regarding the requestor node. The certificates are communicated even to unknown nodes, since the rejection of certificate requests would lead to a deadlock in the case of two-way trust establishment, where the nodes involved are unknown until the processes are finalised. A solution towards avoiding the deadlock while applying constraints to the exchange of certificates is to apply negotiation techniques [discussed in Section 3.1.2], which would regulate the incremental exchange of trust evidence in order to protect its secrecy. This solution was, however, rejected because of the additional communication costs that the negotiations would incur on the sensor nodes.

4.5 Trust Management Processes

Trust management processes are included in the model to handle the trust associations of the nodes and to accumulate and integrate the different types of evidence that the model utilises for trust establishment and revocation according to the configuration of the nodes. Three processes are defined for the corresponding trust management operations of Figure 4.8: The trust associations resolution process, the trust establishment process, and the trust re-evaluation and revocation process. In the following sections,
4.5. TRUST MANAGEMENT PROCESSES

Algorithm 4.1: Resolving the trust association of node \( i \) with node \( j \) following a TRUST-RECORD request by node \( x \)

```plaintext
//Initialisation of the Trust Records constructs
TR_x = null, TR_j = null
Class_j = “unresolved”

if \( i \neq x \) //Resolution due to recommendation request
    TR_x = RECORDS-QUERY (x)
    if \( TR_x = null \) or \( (TR_x.R_{ix} < R_{thresh} \text{ and } x \notin \{\text{Propagation}\}) \)
        return \((j, “unresolved”, null, null, null)\)
    end if
end if
TR_j = RECORDS-QUERY (j)
if \( TR_j = null \)
    //Trust establishment initiated only in the case of internal requests or
    //recommendation requests from nodes in the \{Propagation\} set
    if \( i = x \) or \( x \in \{\text{Propagation}\} \)
        Issue a TRUST-EST.req (j)
        TR_j = TRUST-EST.resp
    end if
end if
Class_j = Classify(TR_j.T_{ij})
return \((j, Class_j, TR_j.T_{ij}, TR_j.R_{ij}, TR_j.SvN_j)\)
```

we provide algorithmic representations and analyse each process, and we discuss the main design decisions and the alternatives for their optimisation.

4.5.1 Trust Associations Resolution

The trust associations resolution process is invoked in node \( i \) by a TRUST-RECORD request following either an internal request or a trust evidence request which has been received from another node. For the description of the process, we assume that \( x \) is the node requesting the trust information (the RequestorID in the TRUST-RECORD.req primitive), which is the same node as \( i \) in the case of an internal request, and \( j \) is the trust target node (expressed as the TargetID in the primitive). The outcome of the process is the trust class of \( j \), along with the values \( T_{ij}, R_{ij} \) and \( SvN_j \) from the trust records, structured as in the TRUST-RECORD.res primitive.

The resolution of the trust association with \( j \) is performed as in Algorithm 4.1. The algorithm first checks whether the entity requesting the information is a client application or another node. If the requestor is another node, the trust records are queried and, if the requestor is unknown or belongs to the “liar” class of Figure 4.4 and is not included in the \{Propagation\} set of the node, then the algorithm returns that the association with \( j \) is “unresolved”. Trust information is thus not disclosed to unknown or distrusted nodes, but only to these that are known and are either classified as “honest”, so they may in turn requested for recommendations, or are included in the \{Propagation\} set, so they are client nodes. The reason for this is twofold: Firstly, to
preserve the resources of the sensor nodes by not allowing further resources to be spent on resolving requests from nodes for which no pre-deployment knowledge exists and no two-way trust association has been established. Secondly, to preserve the secrecy of the trust associations, that could be used by adversaries in order to infer information on the network structure and operations.

In the case of legitimate requests, the algorithm queries the trust records for the association with \( j \). If no record is found, then the trust establishment process is initiated through a TRUST-EST.req only in the case of internal requests or recommendations requests from nodes that exist in the \{Propagation\} set. Using the outcome of the trust establishment process, \( \text{Class}_j \) is resolved using the functional trust value \( T_{ij} \):

\[
\begin{align*}
\text{"unresolved"}, & T_{ij} = \text{null} \\
\text{"trusted"}, & T_{ij} \geq T_{\text{threshold}} \\
\text{"distrusted"}, & T_{ij} < T_{\text{threshold}}
\end{align*}
\]

The information that is then returned using the TRUST-RECORD.res primitive includes the information that is required by the components of Figure 4.8 in order to populate the TRUST-QUERY.res or the (Recommendation)Evidence construct.

### 4.5.2 Trust Establishment

The trust establishment process is initiated during a trust association resolution using the TRUST-EST.req(TargetID) primitive when the trust relationship that needs to be resolved is not already established either before deployment or as a result of a previous trust establishment process. The process attempts to evaluate the trust association with the target \( j \) and, if it is successful, it updates the trust records and responds to the requesting component with the newly generated trust record.

The process accumulates trust evidence of the types that were discussed in Section 4.3.2 so that node \( i \) can determine its trust relationship with any node \( j \) using direct or indirect recommendations from commonly trusted third parties. As shown in Table 4.3, we take a view of a signed certificate from an offline trust management authority as an indirect recommendation with the highest trust value. The process uses referral trust as the metric that determines which recommendations are requested, accepted, and given priority in the process.

Following Algorithm 4.2, node \( i \) accumulates trust \( \{\text{Evidence}_j\} \) for node \( j \) until sufficient evidence is available for trust evaluation. The \( \{\text{Evidence}_j\} \) that is being accumulated by \( i \) is either certificates \( \text{Cert}_{xj} \) issued by trusted certification authorities \( x \), or recommendations \( \text{Rec}_{xj} \) from the mostly trusted nodes. Parameter \( R_{req} \) is the one that determines when sufficient evidence has been accumulated, by being compared whenever a new piece of evidence is made available with the total referral trust \( R_{\text{total}}^{\text{ij}} \) of the nodes that have implicitly or explicitly recommended \( j \). As explained in Section 4.3.3.3 we chose to use \( R_{req} \) instead of the required number of recommendations, since not all recommenders are equally trusted by \( i \), and thus should not contribute the same value towards deciding when to finalise the process.
Algorithm 4.2: Establishing a trust relationship between node $i$ and unknown node $j$

//Initialisation of evidence accumulation constructs
$R_{ij}^{total} = 0$, $\{\text{Evidence}_j\} = \text{null}$

//Accumulation of certificates
if exists $CA_x$ with $R_{ix} \geq R_{\text{thresh}}$
    Send certificates request to $j$
    for each returned certificate $\text{Cert}_{xj}$ issued from known $x : R_{ix} \geq R_{\text{thresh}}$
        if $\text{Cert}_{xj}$ invalid then return $\{\text{null}\}$
        else
            $\{\text{Evidence}_j\} = \{\text{Evidence}_j\} \cup \text{Cert}_{xj}$
            $R_{ij}^{total} = R_{ij}^{total} + R_{ix}$
            if $R_{ij}^{total} \geq R_{\text{req}}$
                Evaluate $T_{ij}, R_{ij}$ using $\{\text{Evidence}_j\}$
                Set $\{\text{Evidence}_j\} = \text{null}$
                Insert into Trust Records and return $(T_{ij}, R_{ij}, R_{ij}^{total}, \text{false}, \text{false}, \text{null})$
            end if
        end if
    end for
end if

//Accumulation of recommendations from third parties
Retrieve Trust Records with $R_{ix} \geq R_{\text{thresh}}$ sorted by $R_{ix}$ descending
for each node $x$ in returned Trust Records
    Send request for recommendation $\text{Rec}_{xj}$ to $x$
    if reply $\neq$ unresolved
        $\{\text{Evidence}_j\} = \{\text{Evidence}_j\} \cup \text{Rec}_{xj}$
        $R_{ij}^{total} = R_{ij}^{total} + R_{ix}$
        if $R_{ij}^{total} \geq R_{\text{req}}$
            Evaluate $T_{ij}, R_{ij}, \text{Supervision}_j$ using $\{\text{Evidence}_j\}$
            Set $\{\text{Evidence}_j\} = \text{null}$
            Insert into Trust Records and return $(T_{ij}, R_{ij}, R_{ij}^{total}, \text{false}, \text{Supervision}_j, \text{null})$
        end if
    end if
end for

return $\{\text{null}\}$
CHAPTER 4. A HYBRID TRUST MANAGEMENT MODEL

The certificates $Cert_{xj}$ and recommendations $Rec_{xj}$ in the algorithm are instantiating the (Certificates)Evidence and (Recommendation)Evidence primitives. Certificates and recommendations requests are sent using the TRUST-EVIDENCE.req primitive where, in the case of certificate requests, $RecipientID = TargetID = j$. The algorithm uses the RECORDS-QUERY interface for searching if there exists $CA_x$ such as $R_{ix} \geq R_{thresh}$, for retrieving the trust records of the certification authorities that issued the obtained certificates, for obtaining the trust records of the possible recommenders shorted by $R_{ix}$, and for inserting the newly calculated trust record. The fields that are included in the inserted records are the ones depicted in Figure 4.8.

The algorithm gives priority to obtaining certificates, because it involves communications only between the two interested parties and does not consume the resources of other network nodes. Moreover, from the types of evidence in Table 4.3, certificates are the most flexible in representing the trust relationships configured before deployment. To minimise the number of certificate communications and validations, node $i$ sends certificates requests only if it has at east one trust association with a trust managing authority, and verifies the obtained certificate only if it trusts the issuing authority $x$. To conserve the energy of the nodes, the process terminates if a certificate is found to be invalid or expired, since a legitimate sender is able to verify its certificates before providing them as evidence.

Recommendations may be further required, either because no verifiable certificate could be obtained, or because the accumulated $R_{ij}^{total}$ of the certification authorities does not suffice. In that case, the collection of recommendations from third parties is controlled: $i$ selects the nodes with the highest referral trust to send its request for $j$. This selection is made because the higher referral trust is assigned to a node, the more its opinion should be valued, and the more frequently it should be consulted for the establishment of new trust relationships. Moreover, a desirable side-effect of this selection is that it leads to less message exchanges, since less recommendations will be required to reach $R_{req}$.

The content of the recommendation messages $Rec_{xj}$ is the same if the recommenders are supervision nodes. The behaviour-based trust evaluation scheme that the supervision nodes implement should thus enable the results of the evaluation to be formulated in separate functional and referral trust values to be provided as $T_{xj}$ and $R_{xj}$ [LI07, ZMHT06]. No criteria or restrictions are applied for recommending a node, except that the recommendations in this initial trust establishment process can only be sent upon request.

Once sufficient evidence has been accumulated, it can be used for the evaluation of $T_{ij}$ and $R_{ij}$, through the metrics that are provided in Sections 4.6.1 and 4.6.2. The Supervision$_j$ parameter, which is stored in the trust records to indicate if $j$ is a supervision node, can be true only if there exists at least one recommendation in the final $\{Evidence_j\}$ set that recommends $j$ as a supervision node. This decision was made to enforce trust associations with supervision nodes to be or result from initial role-based pre-configured associations. In the model, the identification of a node as a trusted supervision node implies that accusations for other nodes should be accepted from it. Therefore, to increase the robustness of the model against defaming attacks, the nodes are recognised as supervision nodes only when there exists trust evidence that results from role-based trust associations.

After the evaluation, only the $R_{ij}^{total}$ metric is stored to indicate the strength of the
evidence utilised. An alternative would be to store the \(\{\text{Evidence}_j\}\) set, which would enable trust traceability and re-evaluation if some of the nodes that had provided the evidence were found to be malicious. Although this would increase the robustness of the model, it would result in considerably higher memory overhead on the sensor nodes, so the \(\{\text{Evidence}_j\}\) was excluded from the parameters that are stored for each trust association (Table 4.2). The total referral trust of \(j\)'s recommenders, \(R_{ij}^{\text{total}}\), is stored instead, as an indication of the strength of the evidence that was used for evaluating the functional and referral trust.

The trust establishment process can resolve a trust relationship only if enough commonly trusted third parties exist; if \(i\) cannot obtain trust evidence originating from trust managing authorities or third parties with a total referral trust of \(R_{req}\), then the trust relationship remains unresolved, to be established in future executions of the process if sufficient evidence becomes available. This will be accomplished either when new trust associations are established between \(i\) and other nodes \(x\) that can provide recommendations, or when the trust relationships between already trusted nodes \(x\) and \(j\) are resolved. An alternative to the process for the case of unresolved trust relationships is to temporarily store the accumulated \(\{\text{Evidence}_j\}\), so that it can be retrieved and enriched in future executions of the process. However, the evidence is dynamic: the trust values recommended by third parties may change, the certificates may expire, and trust in the third parties may be revoked. Since the freshness of the evidence is crucial for the calculation of the trust metrics, the \(\{\text{Evidence}_j\}\) set is initialised with every execution of the process.

### 4.5.3 Trust Re-evaluation and Revocation

Once a trust relationship between \(i\) and \(j\) has been established, it can only be re-evaluated and revoked on the basis of sufficient negative evidence received by trusted supervision nodes using the TRUST-EVIDENCE.res primitive. Through the trust re-evaluation process that follows Algorithm 4.3, the opinion of behaviour evaluation nodes can only decrease the trust value that was initially computed, for the reasons explained in Section 4.1.1.6. The aim of this process is to maintain the set \(\{\text{BEvidence}_j\}\) of the temporarily stored negative recommendations that have been received by supervision nodes for any trusted node \(j\), and re-evaluate trust and update the trust record when sufficient negative evidence has been accumulated.

Upon receipt of a recommendation \(\text{Rec}_{xj}\) from node \(x\) about node \(j\) (instantiating the (Recommendation)Evidence primitive, where TrustValue is represented as \(T_{xj}\) and ReferralValue by \(R_{xj}\)), the algorithm determines the recommendation’s acceptance. Firstly, \(x\) has to be both a trusted recommender and a supervision node. A TRUST-RECORD request is issued in order to retrieve or establish the trust association with \(x\). Accusations from unknown supervision nodes are therefore not rejected, but the process will attempt to establish a trust association with them in order to determine the acceptance of their accusation. The reason for this is that associations with supervision nodes may not be established after deployment due to client application requests, since dedicated supervision nodes may not participate in any other network function except from behaviour monitoring and evaluation. The receipt of accusations may therefore be the only opportunity for the sensor nodes to expand their list of trust associations with supervision nodes after deployment.
Algorithm 4.3: Re-evaluating $T_{ij}$ upon receipt of an accusation $Rec_{xz}$ from node $x$

```plaintext
// Eligibility check for recommender
Issue a TRUST-RECORD.req request for $x$
if $Supervision_x = \text{true and } R_{ix} \geq R_{thresh}$
   // Eligibility check for recommended values
   $TR_j = \text{RECORDS-QUERY (j)}$
   if $TR_j \neq \text{null and } (Rec_{xz}.T_{xz} < TR_j.T_{ij} \text{ or } Rec_{xz}.R_{xz} < TR_j.R_{ij})$
      if previous $Rec_{xz}$ in $TR_j.\{B\text{Evidence}_j\}$
         Replace $Rec_{xz}$ in $TR_j.\{B\text{Evidence}_j\}$
      end if
   else
      $TR_j.\{B\text{Evidence}_j\} = TR_j.\{B\text{Evidence}_j\} \cup Rec_{xz}$
      Filter $TR_j.\{B\text{Evidence}_j\}$
      $R_{ij}^{BEvidence} = \text{sum}(R_{xz}), \forall x : Rec_{xz} \in TR_j.\{B\text{Evidence}_j\}$
      if $R_{ij}^{BEvidence} \geq R_{req}$
         Evaluate $T_{ij}^{new}, R_{ij}^{new}$ using $TR_j.\{B\text{Evidence}_j\}$
         Update $TR_j.T_{ij}$ if $T_{ij}^{new} < TR_j.T_{ij}$
         $R_{ij}^{total} = TR_j.R_{ij}^{total} + R_{ij}^{BEvidence}$
         Update $TR_j.\{B\text{Evidence}_j\} = \text{null}$
      else
         Update $TR_j.\{B\text{Evidence}_j\}$
      end if
   end if
end if
```

The second criterion for the acceptance of the accusation is that $j$ has to exist in the trust records of $i$, and the recommendation has to be negative compared to the current trust values for $j$. The algorithm uses the RECORDS-QUERY interface for retrieving and updating the trust record of $j$ and, in subsequent steps, for retrieving the referral trust values of the nodes whose accusations are included in $\{B\text{Evidence}_j\}$. If the recommendation is accepted, it is used for enriching or updating the set $\{B\text{Evidence}_j\}$, which can not contain more than one recommendation from each supervision node. If a previous $\text{Rec}_{xj}$ exists in the set, then the newly received will replace it.

Before re-evaluation, the $\{B\text{Evidence}_j\}$ is scanned for filtering out recommendations from nodes whose trust has been revoked since their recommendation was accepted, in order to avoid using defaming recommendations from malicious nodes. If sufficient evidence has been accumulated, with $R_{req}$ being again the deciding parameter, $T_{ij}$ and $R_{ij}$ are re-evaluated using the metrics that are provided in Section 4.6.3, and the set $\{B\text{Evidence}_j\}$ is emptied for future recommendations for $j$ to be accumulated. By using $R_{req}$ for determining when to re-evaluate trust, the more trusted the supervision nodes providing the negative recommendations are, the less recommendations will be required, and the sooner trust will be revoked. If a node is persistently reported by enough trusted supervision nodes to be misbehaving, its trust value will gradually converge to the value that the supervision nodes recommend.

A passive approach is undertaken instead of a proactive one for the trust re-evaluation process; a sensor node receiving a negative recommendation could proactively ask for recommendations from other supervision nodes in order to accumulate sufficient evidence for re-evaluation. It was, however, considered that this approach could lead to high communication overheads, and could be exploited by malicious nodes to drain the resources of the sensors.

The trust establishment and the trust re-evaluation processes are highly affected by the value of parameter $R_{req}$. If it is low, then trust could be established and revoked by only a few recommendations, simplifying the evidence collection process and shortening the period required for accumulating behaviour-based trust evaluation results. At the same time, however, it would make the network vulnerable to malicious nodes spreading false recommendations. A high value of $R_{req}$ makes the trust model more robust against defaming attacks, but increases both the communication overheads for collecting sufficient evidence during trust establishment, and the memory overheads for storing the $\{B\text{Evidence}_j\}$ for trust re-evaluation. It is thus important that the value of $R_{req}$ is properly balanced according to the criticality of the deployment and the capabilities and security requirements of the nodes.

### 4.6 Trust Evaluation Metrics

The final operation of the trust establishment and re-evaluation processes that were discussed in the previous sections is the evaluation of the functional and the referral trust metrics $T_{ij}$ and $R_{ij}$ and their use for the update of the trust records. The evaluation of the metrics is based on the information included in the sets $\{\text{Evidence}_j\}$ and $\{B\text{Evidence}_j\}$; for (Recommendation)Evidence items, the values utilised are the $T_{xj}$ and $R_{xj}$ values that the recommendations include, while the (Certificate)Evidence items provide $T_{xj} = R_{xj} = 1$. Apart from the recommended values, the referral trust
of the recommenders \( R_{ix} \) is retrieved from the trust records for each piece of evidence included in the sets and is also used in the evaluation.

In the following sections we define four trust evaluation metrics: Two metrics for the evaluation of the functional and referral trust using the evidence accumulated during trust establishment, and two metrics for the update of the values based on accumulated negative recommendations. We provide the formulas for the evaluation, the rationale for their use, the alternative options that we have considered, and discuss the properties of trust that the selected formulas have.

### 4.6.1 Functional Trust Metric

The trust metric \( T_{ij} \) represents the functional trust value \( T_{ij} \in [0, 1] \) of node \( i \) for \( j \). It is provided by a function that can uniformly calculate the trust value based on the evidence implicitly or explicitly accumulated from the third parties through Algorithm 4.2. Assuming third party \( x \), the function is common for trust evidence represented both by recommendations and by certificates. Given the high importance of the pre-deployment knowledge that exists in WSN deployments, \( T_{ij} \) can be equal to 1 only for role-based trust relationships established prior to deployment between \( i \) and other nodes or trust management authorities and for the associations derived directly from these.

The metric should satisfy the functional trust derivation criterion: Derivation of functional trust through referral trust, requires that the last trust arc represents functional trust, and all previous trust arcs represent referral trust [JGK06]. This implies that (1) only the referral trust \( R_{ix} \) of the recommenders and the recommended functional trust value \( T_{xj} \) should be used for the evaluation of \( T_{ij} \) and, at the same time, (2) for the evaluation of the referral trust \( R_{ix} \) of the recommenders (which is discussed in the following section) only referral trust metrics are utilised. Using \( N_i \) as the set of trusted nodes that \( i \) receives direct or indirect recommendations \( T_{xj} \) for node \( j \), for the evaluation of \( T_{ij} \) a function should be formulated as:

\[
T_{ij} = t(R_{ix}, T_{xj}), x \in N_i
\]

There exist several choices for the function \( t(\cdot) \), which are formed as combinations of the two trust transitivity operators: the link operator \( \otimes \) and the aggregation operator \( \odot \) (discussed in Section 3.1.1). Using the evidence that has been accumulated in the form of recommendations, the link operator for computing the trust derived from a single trust path essentially determines how \( R_{ix} \) and the recommended \( T_{xj} \) from each recommender \( x \) shall be combined. The link operator should satisfy the requirement that trust is weakened or diluted through transitivity [JGK06]. According to Sun et al. [SYHL06], concatenation propagation of trust should not increase trust: The trust value between the issuer and the target should not be more than the trust value between the issuer and the recommender, or the trust value between the recommender and the target. Moreover, the operator must be associative in order to allow computing trust along paths without considering the sub-paths evaluation order [MP07]. The link operator that we use is the product operator, so that the functional trust value computed through a single trust path \( i \leftarrow x \leftarrow j \) is:
4.6. TRUST EVALUATION METRICS

\[ T_{ij}^x = R_{ix} * T_{xj} \]  (4.2)

The product operator satisfies the aforementioned requirements: It is associative, it enforces trust to be weakened through transitivity, since both \( R_{ix} \) and \( T_{xj} \) are in \([0, 1]\), and the resulting trust value \( T_{ij}^x \) will not be higher than \( R_{ix} \) or \( T_{xj} \).

The aggregation operator determines how the contributions from different recommenders shall be combined by aggregating the measures \( T_{ij}^x \) that are computed separately from each trust path. The aggregator operator should also be associative. Moreover, trust should not be reduced because of multipath propagation \( [\text{SYHL06, JP05}] \); if \( T_{ij}^x = T_{ij}^y \) then \( T_{ij} \geq T_{ij}^x \). We use a straightforward approach that satisfies this requirement for an associative aggregation operator, the weighted average:

\[ T_{ij} = \sum_{x \in N_i} (T_{ij}^x * w_x) \]  (4.3)

The weight \( w_x \) that is associated to each recommendation represents the influence that each recommendation will have to the final outcome. It is formulated as the referral trust towards the recommender divided by the total referral trust:

\[ w_x = \frac{R_{ix}}{\sum_{x \in N_i} R_{ix}} \]

The trust metric is formulated as:

\[ T_{ij} = \sum_{x \in N_i} \frac{T_{ij}^x * R_{ix}}{\sum_{x \in N_i} R_{ix}} = \sum_{x \in N_i} \frac{R_{ix}^2 * T_{xj}}{\sum_{x \in N_i} R_{ix}} \]  (4.4)

This metric satisfies the requirement that trust is not reduced through multipath propagation. If \( N_i = \{x, y\} \) and \( T_{ij}^x = T_{ij}^y \) then the metric satisfies \( T_{ij} \geq T_{ij}^x \):

\[ T_{ij} \geq T_{ij}^x \iff \frac{\sum_{x \in N_i} R_{ix}^2 * T_{xj}}{\sum_{x \in N_i} R_{ix}} \geq R_{ix} * T_{xj} \]  (4.5)

\[ \frac{R_{ix}^2 * T_{xj} + R_{iy}^2 * T_{yj}}{R_{ix} + R_{iy}} \geq R_{ix} * T_{xj} \iff \]  (4.6)

Since \( T_{ij}^y = T_{ij}^x \Rightarrow R_{iy} * T_{yj} = R_{ix} * T_{xj} \), and therefore

\[ \frac{R_{ix}^2 * T_{xj} + R_{iy}^2 * T_{yj}}{R_{ix} + R_{iy}} \geq R_{ix} * T_{xj} \iff \]  (4.8)

\[ \frac{R_{ix} * R_{ix} * T_{xj} + R_{iy} * R_{ix} * T_{xj}}{R_{ix} + R_{iy}} \geq R_{ix} * T_{xj} \iff \]  (4.9)

\[ \frac{R_{ix} * T_{xj}(R_{ix} + R_{iy})}{R_{ix} + R_{iy}} \geq R_{ix} * T_{xj} \iff \]  (4.10)

\[ \frac{R_{ix} + R_{iy}}{R_{ix} + R_{iy}} \geq 1 \iff \]  (4.11)

\[ 1 \geq 1 \]  (4.12)

The metric in 4.4 does not therefore reduce trust because of the aggregation of trust
paths. As an operator, however, the weighted average is not idempotent. This would be a desirable property for the aggregation operator \[MP07\], meaning that multiple applications of the aggregation operation would not change the result in the case of overlapping trust paths. An idempotent aggregation operator, that could be used as an alternative to the weighted average, is the maximum. The functional trust metric using the maximum as the aggregation operator would be

\[ T_{ij} = \max(T_{ix} \cdot T_{xj}) \], \( x \in N_i \). However, we consider that this metric is too optimistic; it selects the trust path that is the most favourable for the trust target, and does not take into account the other trust paths that may even contain negative recommendations from highly trusted recommenders. It is therefore provided as an alternative, but the metric in 4.4 is the one that is used in the model and was implemented for the simulation analysis.

### 4.6.2 Referral Trust Metric

The referral trust metric \( R_{ij} \in [0, 1] \) is the second part of a trust association, essentially indicating the trust that node \( i \) assigns to \( j \) for providing recommendations. This metric is also used as the means to control trust evolution and spreading according to the level of distrust that each node should exhibit towards unknown parties since, as explained in Section 4.3.3.2, the degradation parameter \( d_i \in [0, 1] \) is utilised for its evaluation.

The evaluation of the referral trust of node \( j \) based on the accumulated recommendations from nodes \( x \in N_i \) should utilise only the recommended referral trust values and not the functional ones. This decision is made in order to satisfy the functional trust derivation criterion \[JGK06\]: In any valid transitive trust path, only the last arc should represent functional trust. Therefore, functional trust should not be used for the evaluated \( R_{ij} \), so that it can be used as an intermediate arc for the construction of future trust paths. Moreover, referral trust is essentially functional trust for the function of providing recommendations. Using this interpretation, the last arc that should be used for its evaluation is the recommended referral trust \( R_{xj} \). The referral trust metric \( R_{ij} \) is therefore evaluated by a function accepting as parameters \( d_i \), the referral trust assigned to the nodes in \( N_i \), and the recommended referral trust \( R_{xj} \):

\[ R_{ij} = r(R_{ix}, R_{xj}, d_i), x \in N_i \] (4.13)

The function \( r(.) \) should enforce the degradation of the value \( R_{ij} \) according to \( d_i \), so that when \( d_i = 1 \) no explicit degradation is performed to referral trust according to the length of the recommendation path from a node that \( i \) is pre-configured to trust, while when \( d_i = 0 \), then the referral trust \( R_{ij} \) assigned to any unknown node \( j \) during trust establishment should be 0. A possible optimistic \( r(.) \) can be formulated as:

\[ R_{ij} = \max(R_{ix} \cdot R_{xj}) \cdot d_i, x \in N_i \Rightarrow T_{xj} \geq T_{ij} \] (4.14)

This function uses for the computation of \( R_{ij} \) the maximum referral trust value, from the nodes whose recommendations are greater than or equal to the trust value computed for \( j \). Since, however, referral trust is essentially functional trust for the function of providing recommendations, we chose to use the same function for referral trust, incorporating the degradation parameter:
4.6. TRUST EVALUATION METRICS

\[ R_{ij} = \frac{\sum_{x \in N_i} R_{ix}^2 \ast R_{xj}}{\sum_{x \in N_i} R_{ix}} \ast d_i \]  \hspace{1cm} (4.15) \]

The advantage of utilising this metric is that it has all the desirable trust transitivity properties that the functional trust metric has: the link and aggregation operators are associative, referral trust is weakened through transitivity and, at the same time, it is not reduced through the aggregation of alternative trust paths. Unlike the function in 4.14, it utilises all the recommendations for the calculation of the final result. Moreover, it incorporates the degradation parameter so that, when \( d_i = 1 \), it does not affect the evaluation, while setting \( d_i = 0 \) makes \( R_{ij} = 0, \forall j \in N_i \), and thus \( i \) will not accept recommendations from nodes except the ones it is pre-configured to.

4.6.3 Trust Re-evaluation

The trust value \( T_{ij} \) is re-evaluated based on the set \( \{BEvidence_j\} \) of recommendations from supervision nodes. We do not distinguish these new recommendations from the ones provided during the initial evaluation; they should contribute to the trust value \( T_{ij} \) as if they were initially obtained. However, Algorithm 4.2 does not store the recommendations initially received, but only the calculated \( T_{ij} \) and \( R_{ij}^{total} \) of the total referral trust of the initial recommenders. For this reason, trust re-evaluation is made exactly as the initial trust evaluation using only the recommendations in \( \{BEvidence_j\} \) as input to Equation 4.4. It \( M_i \) are the nodes that provided the recommendations in the \( \{BEvidence_j\} \) set. \( T_{ij}^{new} \) is computed using only these recommendations as:

\[
T_{ij}^{new} = \frac{\sum_{x \in M_i} R_{ix}^2 \ast T_{xj}}{\sum_{x \in M_i} R_{ix}}
\]  \hspace{1cm} (4.16) \]

The obtained value \( T_{ij}^{new} \) is then combined with the current trust value \( T_{ij}^{current} \) for the update of \( T_{ij} \). The current trust value \( T_{ij}^{current} \) and the obtained value \( T_{ij}^{new} \) are interpreted as trust values resulting from different trust paths. The aggregation operator for their combination is the one used for the initial calculation of the functional trust metric in 4.3, the weighted average:

\[
T_{ij} = T_{ij}^{current} \ast w_{current} + T_{ij}^{new} \ast w_{new} = \frac{R_{ij}^{total} \ast T_{ij}^{current} + R_{ij}^{BEvidence} \ast T_{ij}^{new}}{R_{ij}^{total} + R_{ij}^{BEvidence}}
\]  \hspace{1cm} (4.17) \]

The value of \( R_{ij}^{BEvidence} \) in the formula is the total referral trust of the recommenders in \( \{BEvidence_j\} \), obtained following Algorithm 4.3. It should be noted that the interpretation of \( T_{ij}^{current} \) and \( T_{ij}^{new} \) as trust values resulting from different trust paths, which enables the use of the aggregation operator for their combination, may not be accurate: There may exist overlappings between the nodes that provided the evidence of the two sets. This can not be verified or taken into account during the calculation, since the evidence is not stored. However, this interpretation does not have a negative effect on the updated trust values as long as the trust resulting from multiple recommendations from a single source is not higher than that from independent sources [JP05]. This holds for Equation 4.17 because (1) the recommenders are unique in the
{BEvidence}_j set, and (2) if there is overlapping between the previous recommenders and these in the {BEvidence}_j set, it results from a trust re-evaluation operation which, according to Algorithm 4.3, accepts recommendations that can only be negative in relation to the current trust value.

The referral trust value \( R_{ij} \) is similarly updated using the recommendations in \( \{BEvidence\}_j \) as input to Equation 4.15 in order to obtain \( R_{ij}^{new} \). The new referral trust value is then combined with the current value for the update of \( R_{ij} \) by substituting the functional with the corresponding referral trust values in Equation 4.17.

All evaluation metrics that are used in the trust establishment process satisfy the requirement that the more trusted a node is, the more influence its opinion has on the final result. This also applies on trust re-evaluation, both for the combination of the recommendations from supervision nodes and for the recalculation of the trust values. Moreover, due to \( R_{ij}^{total} \) and \( R_{ij}^{BEvidence} \), the effect of re-evaluation on the trust values is greater the stronger the newly acquired evidence is in comparison to the evidence that was used for the calculation of the current values.

### 4.7 Analysis of the Model

In this section the trust management model and its processes and metrics are evaluated both through analysis against the requirements that were initially set, and through the study of the security and trust properties that the model exhibits. Recognising the criticality of the requirement for lightweight operation in WSN deployments, in Section 4.7.4 we estimate and discuss the resource requirements of the model and the resource overheads that it would impose on the network nodes.

#### 4.7.1 Evaluation Against Requirements

The trust management model consists of processes, metrics, configuration parameters and trust associations and evidence, all defined to fulfil the trust management requirements of Section 4.1.1 and realise our approach for addressing them. Table 4.5 summarises the aspects that are included in the trust management model towards fulfilling each of the requirements initially set.

Regarding the application of the model to WSN deployments, configuration parameters have been defined to enable restricting and distributing the trust management operations executed by the nodes, and the trust management processes are designed to minimise the message exchanges and the certificate validations according to the pre-configuration of the nodes. Through the configuration options and the trust management processes, the model enables the restriction of the alternative options for trust establishment for constrained sensor nodes so that they do not need to perform certificate validations or combinations of recommendations.

The model can support the heterogeneity in the trust evaluation needs of the nodes and the deployments through the configuration parameters that enable expressing the distrust that each node should exhibit after deployment, and affect both trust evolution and the resource overheads that the trust management operations impose. Moreover, it enables the utilisation of the pre-deployment knowledge on the roles and the trust associations of the nodes through the pre-configuration of trust associations that are given priority in all trust management processes.
### 4.7. ANALYSIS OF THE MODEL

Table 4.5: Analysis of the model against the trust management requirements. The model aspects are Associations (A), Evidence (E), Interfaces(I), Configuration parameters (C), Processes (P), and Metrics (M).

<table>
<thead>
<tr>
<th>Requirement to WSN deployments</th>
<th>Supporting model aspects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C: Locally stored public keys of offline trust managing authorities for local certificate verification.</td>
</tr>
<tr>
<td></td>
<td>C: $R_{req}$ used for restricting the required evidence and for lightweight configuration of the model.</td>
</tr>
<tr>
<td></td>
<td>C: {Propagation} set for expressing the delegation of resource consuming operations to pre-determined nodes.</td>
</tr>
<tr>
<td></td>
<td>C: Separate $R_{thresh}$ for the configuration of role-based associations with trusted nodes that should not be used as recommenders.</td>
</tr>
<tr>
<td></td>
<td>P: Trust associations resolution rejects recommendation requests from unknown/distrusted nodes to preserve resources.</td>
</tr>
<tr>
<td></td>
<td>P: Trust establishment gives priority to obtaining certificates to preserve the resources of possible recommenders.</td>
</tr>
<tr>
<td></td>
<td>P: Trust establishment minimises the certificate requests and validations and the collection of recommendations.</td>
</tr>
<tr>
<td></td>
<td>P: Passive approach for the collection of accusations for trust re-evaluation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement to WSN deployments</th>
<th>Supporting model aspects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C: Support for semantically independent role-based and certificate-based associations.</td>
</tr>
<tr>
<td></td>
<td>P: Trust re-evaluation only decreases the trust values based on behaviour-based evidence.</td>
</tr>
<tr>
<td></td>
<td>P: Trust associations resolution and establishment give priority to role-based associations and certificates.</td>
</tr>
<tr>
<td></td>
<td>I: TRUST-QUERY interface provided to the client services for obtaining both the trust class and the trust value.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Supporting model aspects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application to WSN deployments</td>
<td>C: $d_i$ for configuring the explicit degradation of referral trust.</td>
</tr>
<tr>
<td>Support for heterogeneity</td>
<td>C: $R_{req}$ for expressing the required amount of evidence for trust establishment and revocation.</td>
</tr>
<tr>
<td>Support for heterogeneity</td>
<td>C: $R_{thresh}$ for configuring the level of distrust that the nodes should exhibit.</td>
</tr>
<tr>
<td>Utilisation of pre-deployment knowledge</td>
<td>A, E: Role-based associations with other nodes, certification authorities, and supervision nodes can be configured and used as evidence.</td>
</tr>
<tr>
<td>Utilisation of pre-deployment knowledge</td>
<td>E: Certificates supported to allow for flexibility in the representation of pre-deployment knowledge.</td>
</tr>
<tr>
<td>Utilisation of pre-deployment knowledge</td>
<td>P: Trust associations resolution does not invoke trust establishment if role-based association exists.</td>
</tr>
<tr>
<td>Utilisation of pre-deployment knowledge</td>
<td>P: Trust establishment gives priority to obtaining certificates and recommendations from the most trusted nodes.</td>
</tr>
<tr>
<td>Utilisation of pre-deployment knowledge</td>
<td>M: $T_{ij} = 1$ only for pre-established trust relationships and for the associations derived directly from these.</td>
</tr>
<tr>
<td>Trust revocation due to malicious behaviour</td>
<td>A: Role-based associations can be configured with supervision nodes.</td>
</tr>
<tr>
<td>Trust revocation due to malicious behaviour</td>
<td>E: Behaviour-based trust evaluation results used as evidence for trust re-evaluation.</td>
</tr>
<tr>
<td>Trust revocation due to malicious behaviour</td>
<td>I: TRUST-EVIDENCE interface enables the receipt of accusations for malicious nodes.</td>
</tr>
<tr>
<td>Trust revocation due to malicious behaviour</td>
<td>P: Trust re-evaluation process for revoking trust due to malicious behaviour accusations.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>E: Combination of different types of evidence for trust establishment.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>C: $R_{thresh}$ for configuring the strength of the required evidence for trust revocation.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>C: $R_{thresh}$ enables the rejection of the recommendations from distrusted recommenders.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>P: Trust associations resolution rejects recommendation requests from unknown/distrusted nodes to preserve the secrecy of the trust associations.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>P: Trust establishment recognises supervision nodes only when there exists such evidence that results from role-based trust associations.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>P: Trust re-evaluation determines the accusations’ acceptance, temporarily stores them and filters them before re-evaluation decisions.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>M: Trust metric not too optimistic, takes into account all available trust paths.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>M: All metrics satisfy the requirement that the more trusted a node is, the more influence its opinion has on the final result.</td>
</tr>
<tr>
<td>Robustness against attacks on trust management</td>
<td>M: Trust re-evaluation affects the trust values according to the strength of the newly acquired evidence in comparison to the evidence utilised for the current trust value.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Supporting model aspects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generic scope and utilisable results</td>
<td>A, E: Support for semantically independent role-based and certificate-based associations.</td>
</tr>
<tr>
<td>Generic scope and utilisable results</td>
<td>P: Trust re-evaluation only decreases the trust values based on behaviour-based evidence.</td>
</tr>
<tr>
<td>Generic scope and utilisable results</td>
<td>P: Trust associations resolution and establishment give priority to role-based associations and certificates.</td>
</tr>
<tr>
<td>Generic scope and utilisable results</td>
<td>I: TRUST-QUERY interface provided to the client services for obtaining both the trust class and the trust value.</td>
</tr>
</tbody>
</table>
CHAPTER 4. A HYBRID TRUST MANAGEMENT MODEL

Trust revocation due to malicious behaviour is enabled through the trust re-evaluation process that receives behaviour-based evidence by trusted supervision nodes through the interface that the model includes. To increase the robustness of the trust management model against attacks, different types of evidence are utilised, the strength of the evidence is explicitly represented, the recommendations are selected and filtered, and the trust metrics take into account the referral trust of the recommenders. In Section 4.7.3 the robustness of the model against different types of attacks is further discussed. To sustain the generic scope of the model, trust is build on types of evidence whose semantics can be instantiated according to context for which the model is used, and the only type of evidence that is not semantically independent (the behaviour evaluation results) are utilised only for identifying and revoking trust from malicious nodes. Moreover, the solution has been designed to operate as an independent security service that provides the appropriate interface to its client services or applications.

4.7.2 Security and Trust Properties

The trust management model exhibits a set of trust and security properties of more generic scope than the specific requirements that it was designed to fulfil. These are summarised to:

**Respect to trust transitivity properties** The basic trust transitivity properties that any trust management solution should exhibit (discussed in Section 3.1.1) specify the types and scopes of trust that should be used and the types of operations that should be applied for the derivation of valid trust values through transitivity. The model respects these properties by separating functional from referral trust, by allowing the construction of trust paths where only the last trust arc represents functional trust, by enforcing referral trust to be weakened through transitivity and not reduced through the aggregation of alternative trust paths, and by utilising the referral trust of the recommenders for the computation of the trust values of the targets. The metrics do not, however, utilise idempotent operators for the aggregation of trust paths, which is a desirable, but not a necessary property.

**Controlled trust evolution according to pre-deployment knowledge and network purpose** As discussed in Section 4.5, the model is targeted at purpose-specific networks, where node cooperation decisions should be taken based on the deployment needs, and good node behaviour cannot compensate for the distrust that the deployment needs may dictate. Through the role-based trust associations and the configuration parameters of Section 4.3.3, the model enables the network designer to control trust evolution. Simulation results in Section 4.8.2 demonstrate the model’s effectiveness in establishing trust relationships with unknown nodes according to the pre-configuration decisions.

**Semantic coherence of trust evidence** For the model to remain generic, the semantics of trust have not been specified, and the function that functional trust represents has not been instantiated. According to the context for which the model is used, functional trust may represent trust for routing decisions, for key exchange, for data disclosure, or for other network operations that require node cooperation. The model
enables semantic coherence, if all the trust evidence that is used is coherent: Both explicit and implicit recommendations must contain semantically coherent recommended functional trust metrics $T_{xj}$. For this to be accomplished, certificates should be issued and role-based trust associations should be defined according to the context for which the trust model is used.

**Controlled trust revocation** While the model can be intergraded with most behaviour evaluation models, as discussed in Section 4.3.2, trust revocation is controlled: The supervision nodes providing the evidence must be trusted as recommenders, sufficient evidence must be accumulated, and trust will be revoked according to the strength of the newly obtained evidence in comparison to the evidence that was used for the evaluation of the current trust values. Sections 4.8.4 and 4.8.5 include simulation results that demonstrate the ability of the model to handle trust revocations in the presence of malicious recommenders.

**Distributed trust computation** The trust management processes of Section 4.5 enable all operations to be performed in a distributed manner, without the need for online trust management authorities. Trust evidence is maintained and exchanged cooperatively by the nodes, and each node computes and maintains its own local trust values.

**Controlled distribution of trust evaluation overhead** Through the $\{\text{Propagation}\}$ parameter and the restriction of the operations that can be performed by each node, the model enables controlling the distribution of the trust evaluation operations between the network nodes. It essentially enables controlling which operations shall be performed locally at each node and which shall be delegated to other, possibly more powerful, nodes. Section 4.8.5 of the simulation analysis includes an example of how this parameter can be used to delegate the resource consuming evaluation operations of a cluster to its cluster head.

**Resilience against trust saturation and low trust valuation problems** The first problem can appear due to a long history of positive node behaviour, leading to trust saturation, while the latter when trust stays low because of the even distribution of positive evidence, which can lead to the isolation of good nodes. Both problems, presented in reference [LV07], mainly apply to behaviour-based evaluation models. Our model avoids both issues by combining hard and soft trust relationships on common evaluation processes. Trust saturation cannot appear because behaviour evaluation results cannot increase trust. Low trust valuation in the presence of positive evidence can appear only because of deployment decisions, through the assignment of a low value to the explicit trust degradation parameter.

**4.7.3 Robustness Against Attacks**

Trust management models can be subjects to attacks aiming to affect trust evolution or disrupt the trust management operations. These attacks either target the reputation of legitimate nodes and try to make them appear malicious, or aim to increase the trust assigned to malicious nodes so that bad reputation is not spread in the network.
For the trust management model to operate securely, it has to be robust against the following types of attacks:

**Attacks against evidence authenticity and integrity**  Attackers may attempt to impersonate trusted network nodes and send false replies to recommendation requests that they have overheard, or impersonate trusted supervision nodes and send accusations for other trusted nodes. They may also attempt to modify the trust evidence reply messages, especially if they have compromised nodes that are within the multi-hop routes that the messages traverse. These attacks can affect how trust evolves both towards the attacker’s advantage and towards the isolation of legitimate nodes. For its protection against these attacks, the model relies on the lower level security mechanisms that provide authentication and integrity protection services for the evidence exchange messages. Moreover, it does not disclose information about the in-network trust associations to distrusted or unknown nodes which, if the exchanged evidence messages are encrypted, hinders attackers from learning the identities of the nodes that they would have to impersonate.

**Defaming attacks**  Compromised network nodes may launch attacks aiming to spread negative evidence about legitimate nodes. To increase the robustness of the model against such attacks, a number of measures have been taken: different types of evidence are utilised for trust evaluation, the strength of the evidence affects the computation of the trust values, and the accusations that are received are temporarily stored and filtered before they are utilised. Moreover, since the model accepts the behaviour-based trust evaluation results as evidence, it allows the network nodes to be informed about malicious nodes that have been identified by the supervision network. The effect of defaming attacks on the trust values is further studied through the experiments in Section 4.8.4.

**On-off attack**  During the on-off attack malicious entities behave well and badly alternatively so that their reputation does not degrade [SHL08]. The vulnerability of the model to this attack is limited, since it does not allow trust to be build solely on good behaviour evidence when there are other types of evidence, like role-based trust associations or certificates, available. Through the trust re-evaluation process, the behaviour-based evidence can only decrease the trust value initially computed, so the alteration of the behaviour of the malicious nodes can only have a negative effect on their trust values. The on-off attack may affect the operation of the supervision deployment, limiting its ability to identify the malicious nodes and provide the negative evidence, which is, however, outside the scope of our work since it depends on the behaviour-based trust evaluation model that is being applied.

**Conflicting behaviour attack**  During the conflicting behaviour attack the malicious entities behave inconsistently in the user domain instead of the time domain [SHL08]. Similarly to the on-off attacks, the affect of this attack on the trust management model is limited, and the identification of malicious nodes mainly depends on the behaviour-based trust evaluation model that is being applied.
4.7. ANALYSIS OF THE MODEL

Sybil attack and newcomer attack A malicious node can operate using several faked identities or register using a new identity when its trust has been revoked [SHL08]. For protection against faked identities, the model relies on the authentication mechanisms that are provided by the security components. It is only partially vulnerable to the newcomer attack; since malicious nodes using new identities can not provide valid or verifiable certificates, and role-based trust associations will not exist for their new identities, the only evidence that may become available for them is their behaviour evaluation results from the supervision network. The model is thus vulnerable to this attack only if the nodes are configured to accept the results of behaviour evaluation, and if the supervision network is vulnerable to this attack.

4.7.4 Resource Requirements and Overheads

The computational requirements of the model and the energy consumption for executing the trust management processes depend on the type of each node and its pre-configuration. For example, nodes that do not have any pre-configured trust associations with certification authorities shall never request or accept certificates as trust evidence and will not perform any certificate validation operations. If, at the same time, they have been configured with an $R_{req}$ value that is lower than their referral trust for their most trusted node, and this node has them included in its $\{\text{Propagation}\}$ set, then the only operation they will need to execute during each trust establishment operation is to request and receive one recommendation. Unlike this case, nodes that have been configured with a high $R_{req}$ will need to accumulate large amounts of trust evidence for each trust establishment or re-evaluation operation, which affects both the communication and the memory overheads that the model imposes.

The memory requirements of the model depend on $R_{req}$ and on the number of the trust associations of the nodes. Each entry in the trust records contains the identity of the trustee, the two trust metrics, the total referral trust, two flags and the $\{B\text{Evidence}_j\}$ set. If the set is empty, the network uses the 64-bit extended addresses of the IEEE 802.15.4 standard, and the trust values are 32-bit floating point numbers, then each trust association entry requires $64 + 32 + 32 + 32 + 1 + 1 = 162$ bit $\approx 21$ bytes memory space. The $\{B\text{Evidence}_j\}$ set is composed of $\text{(Recommendation)Evidence}$ constructs, each containing the identifiers of the recommender $x$ and of the recommended node $j$, $T_{xj}$, $R_{xj}$, and the flag $\text{Supervision}_j$, which result to a size of approximately 24 bytes. The set is, however, emptied when $\text{sum}(R_{xj}) \geq R_{req}$, so the maximum number of entries that it may hold depends on the value of the $R_{req}$ configuration parameter.

Although the energy requirements of the model depend on its configuration, we can estimate the requirements per trust management operation. The experimental studies that were discussed in Section 3.3.1.3 have measured the energy costs for message transmissions and for cryptographic operations on a variety of sensor devices. Based on these studies, we estimate that the most expensive of the trust management operations included in the model is the certificate communication and validation. As shown in Table 3.4, Wander et al. measured that for Mica2dot sensors, with 4MHz AT-mega128L 8-bit microcontroller, the energy required for signature verification is 11.9$mJ$ for RSA-1024 and 45.09$mJ$ for the equivalent ECDSA-160 certificates [WGE+05]. The same authors estimate that a simplified RSA-1024 certificate is 262 bytes long and an ECDSA-160 certificate can be reduced to 86 bytes, while the energy to transmit
is $59.2 \mu J/\text{byte}$ and the energy to receive is $28.6 \mu J/\text{byte}$. This means that, without taking into account the communication protocol, packet size and structure, the cost for transmitting only (Certificates)Evidence when ECDSA-160 certificates and 64-bit network addresses are used would be $(8 + 8 + 86) \times 59.2 \mu J = 6.01 mJ$, while for receiving it $(8 + 8 + 86) \times 28.6 \mu J = 2.9 mJ$.

The communication costs for the exchange of recommendations are considerably lower. Each (Recommendation)Evidence contains the identifiers of the recommender $x$ and of the recommended node $j$, $T_{xj}$, $R_{xj}$, and the flag $\text{Supervision}_j$. With 64-bit addresses and 32-bit trust values, the cost for transmitting only the data payload ($\approx 24$ bytes) of a recommendation message would be $24 \times 59.2 \mu J = 1.4 mJ$ and for receiving it, $24 \times 28.6 \mu J = 0.68 mJ$. However, the total cost for exchanging recommendations also depends on other factors: the underlying security mechanisms for protecting the authenticity and integrity of the recommendation messages (experiments on MICA2 nodes have approximated a 14% increase on the energy required to transmit a packet in TinyOS authentication and encryption mode in comparison to the default mode, mainly attributed to the number of extra bytes transmitted due to the security services [GSSK05]), and the length of the multihop routes that they may need to traverse. It is because of these factors that in the simulation analysis of the following section we examine separately the inter-cluster recommendations from the intra-cluster ones. The inter-cluster recommendations will need to traverse shorter routes; the nodes may even be 1-hop neighbours. Moreover, the nodes of the same cluster are more likely to be equipped pre-shared symmetric keys, thus the communication of recommendations would incur lower security overheads than in the case of intra-cluster recommendations that may require the establishment of cryptographic keys.

The most expensive of the operations that the model includes, the certificate communication and verification, is considered acceptable for the sensor nodes if ECC-based certificates are used [AEOQ05, GKOS05, WGE+05]. The model does, however, provide the means to control the type and minimise and distribute the required operations according to the trust evaluation requirements of the nodes, so that it can be uniformly applied to WSN deployments where the nodes that are designated for the roles of cluster heads or gateways are computationally more powerful than the sensor nodes.

### 4.8 Experimental Evaluation

In order to validate the proposed model and the algorithms and metrics that it includes, we utilised the simulation environment that was introduced in Section 4.2. The experiments are made using a simple simulation setting, similar to the one of the working scenario, in order to evaluate how trust evolves in the network and what operations are required. In this section, we describe the simulation setup and parameterisation, the network model and the node configuration of the simulation scenario. We then present and discuss the results that were obtained in terms of the established trust relationships, the required operations, their distribution among the network nodes, and the results of the trust revocation operations.

It should be noted that these types of results were selected because they were the ones that we considered valuable for this trust model. Since behaviour monitoring and malicious node detection is out of its scope, robustness against collusion or defaming
attacks in the supervision deployment is not evaluated. Instead, in Sections 4.8.4 and 4.8.5, we evaluate how effectively the model can use behaviour-based trust evaluation information once it is provided by supervision nodes, in order to isolate the nodes that were found to be misbehaving.

Moreover, since the model offers a rich set of parameters for controlling trust evolution according to the pre-deployment knowledge that may exist, in Section 4.8.5 we study the effects of an enriched initial configuration. We vary the parameters and the pre-established trust associations of the initial scenario, and examine how the modified configuration affects the trust evaluation operations and results.

4.8.1 Simulation Setup

For our experiments we assume a scenario that can combine all types of trust evidence that the model utilises, including certificates, locally stored information on role-based trust associations, and recommendations from third parties and supervision nodes. Figure 4.9 depicts the logical grouping of the nodes in the scenario. The group that is the focus of our experiments is a WSN composed of 100 nodes, of which the 10 WSNc nodes are certified by an offline trust managing authority (CA). Behaviour-based trust evaluation results are provided as an independent network service by the supervision deployment SvN, which is composed of 100 nodes. A separate SvN deployment was assumed instead of designated WSN nodes for providing this service, to enable distinguishing and examining the interactions between the sensor nodes and the supervision nodes.

Figure 4.9 also shows the initial parameterisation and trust associations that were set for the simulation. For each WSN node, these include an association with the fully trusted CA and associations with 10 randomly selected highly trusted (T_{ij} and R_{ij} between 0.7 and 0.9) SvN nodes. The initial configuration does not include any trust associations between the WSN nodes. Regarding the configuration parameters, \( R_{req} \) is set to 2, so that each newly established trust association requires evidence from two fully trusted or more than two marginally trusted third parties. The degradation parameter \( d_i \) is set to 1, so that trust does not degrade according to the distance from pre-established trust associations.

The simulation proceeds in simulation rounds in three phases. In the first phase, we assume that the supervision nodes have not gained enough experience from the behaviour of the WSN nodes, so they can only provide recommendations for the rest of
the SvN nodes. In each of the 200 rounds of the first phase, request messages are sent towards random WSN nodes. The number of the requests in each round is 100, equal to the size of the WSN. The sender of each request message is also randomly selected, to simulate the effect of the dynamically changing physical position of the WSN nodes. With the receipt of each request message, the recipient resolves its trust relationship with the sender and, if the sender is unknown, initiates the trust establishment process.

The second phase proceeds like the first for an equal number of rounds, but the SvN nodes are assumed capable of providing recommendations for the WSN nodes. In both phases, the recommendations are provided only on demand, as responses to recommendation requests generated during the trust establishment process. The trust values that are recommended by the SvN nodes for the nodes of both the SvN and the WSN that are behaving legitimately are between 0.7 and 0.9, while the trust values for those that are detected to be misbehaving are between 0.2 and 0.4. We assume that the SvN nodes have detected that 10% of the WSN nodes and 10% of the SvN nodes act maliciously. The malicious nodes may include nodes with which pre-established trust associations have been defined. In order to examine separately the effects of the trust establishment and of the trust re-evaluation operations, recommendations are not sent proactively, and no trust re-evaluation operations are performed during the first two phases.

In the third phase, the next 400 rounds of the simulation focus on trust re-evaluation due to the propagation of negative behaviour evaluation evidence from the SvN nodes. In each round, a random legitimate supervision node sends to all WSN nodes a recommendation about a random malicious node. At the same time, we assume that the malicious SvN nodes recommend each other as highly trusted, and try to defame other nodes by continuously recommending a trust value between 0.1 and 0.3. For the defaming attack to be more effective, they collude on defaming a particular set of nodes: the 10 WSNc nodes. Moreover, their tactic is aggressive: in each round, a random malicious supervision node sends negative recommendations about all falsely accused WSNc nodes to all WSN nodes.

During the three phases, statistics are accumulated at each node for its trust relationships and the operations it performs. The simulation is run several times, and for most performance metrics the results are averaged.

### 4.8.2 Established Trust Relationships

As performance metrics for the first two phases of the simulation, we are interested in the number of trust relationships of each type, the values of the functional and referral trust metrics for each relationship, and the required number of operations. The results obtained are illustrated in Figures 4.10, 4.11, and 4.12 respectively.

Figure 4.10 depicts the number of trust relationships of each type collectively for all WSN nodes, averaged from 20 simulation runs. The results suggest that, until the second phase of the simulation (round 200), only a small number of trust associations can be resolved, which is attributed to the insufficient evidence available during the first phase. The WSN nodes can evaluate their trust relationships only with the SvN nodes through combining two types of evidence, namely their locally stored information on the pre-established trust associations, and the behaviour-based trust evaluation results that the trusted SvN nodes make available. They can not resolve the trust relationships
within WSN, since the only available evidence during the first phase is the certificates of the $WSN_c$ nodes, which do not suffice as evidence, because $T_i,CA = 1 < R_{req} = 2$.

This changes during the second phase, since more evidence becomes available by the $SvN$ nodes and the WSN nodes can resolve the inter-network trust relationships by combining certificate-based with behaviour-based evidence. As expected, the number of trust relationships only decreases after the second phase, since only trust re-evaluations occur.

It is of special interest to examine which exactly trust relationships were established, and what the results for the trust metrics were. Figure 4.11 (a) illustrates the pre-established trust associations of one of the WSN nodes, while (b) the relationships that were established at the end of the first phase. Most of the $SvN$ nodes are less trusted than the 10 pre-known $SvN$ nodes, although positive recommendations were provided, because their trust values diluted through transitivity. The trust associations at the end of the second phase are shown in (c). Most WSN nodes are assigned trust values similar to the ones of the $SvN$ nodes, since similar evidence was used for the evaluation. The $WSN_c$ nodes were assigned higher functional and referral trust values, since their evidence included a valid certificate from the fully trusted $CA$.

Some of the nodes are distrusted because they have been recognised as malicious and negative recommendations were used as evidence. Some of the WSN nodes had pre-established trust associations with $SvN$ nodes that are malicious. It was observed that the nodes that highly trusted malicious $SvN$ nodes as recommenders included the falsely positive trust values that the malicious $SvN$ nodes recommended in the evidence that they accumulated for other malicious $SvN$ nodes, and they assigned relatively high values to other malicious nodes. Node $SvN_{92}$ in Figure 4.11 (a) is a trusted malicious node, which was not given the opportunity to affect the result because its referral trust is not high enough to be selected as a recommender.
Figure 4.11: Trust relationships of node $WSN_0$

(a) Node $WSN_9$, Round 0

(b) Node $WSN_9$, Round 200

(c) Node $WSN_9$, Round 400

(d) Node $WSN_9$, Round 800
4.8. EXPERIMENTAL EVALUATION

Figure 4.12: Operations per round by WSN nodes

4.8.3 Required Trust Evaluation Operations

During the first phase, we also examined the types and number of operations that were required to reach the point where most trust relationships with the WSN cluster have been resolved. Under examination were the operations that are the most costly in terms of computation and power consumption: the certificate validations and the recommendations exchanges. The number of operations of each type was examined separately because each type of operation is estimated to incur different costs in terms of computation and power consumption, as explained in Section 4.7.4.

In Figure 4.12, we plot the number of certificate validations and the number of recommendations exchanged, averaged from 20 simulation runs. During the experiments, certificates were received and validated by the WSN nodes when they had to evaluate their trust associations with the 10 WSN nodes. However, since the certificates alone do not suffice as evidence, the trust relationships were not resolved—and the number of certificate validations did not decrease—until the second phase when more evidence became available.

Most operations were exchanges of intra-cluster recommendations, i.e., recommendations sent from SvN nodes to WSN nodes. This was anticipated, since the majority of the evidence that is used in this scenario is the behaviour-based trust evaluation results. Moreover, since \( R_{req} = 2 \) and no SvN node is fully trusted as a recommender, the evidence required for each new trust association includes 3 implicit or explicit recommendations from the mostly trusted third parties—each WSN node is pre-configured to trust 10 SvN nodes with \( R_{ix} \geq 0.7 \), so there exist at least 3 recommenders to accumulate \( R_{ij}^{total} \geq 3 \times 0.7 > R_{req} \). A significant amount of evidence was thus required, which explains why the number of intra-cluster recommendations is high. After the first phase, some recommendations were also exchanged between the WSN nodes. These
4.8.4 Trust Revocation Operations and the Impact of Colluding Malicious Recommenders

During the third phase of the simulation, the SvN nodes send to the WSN nodes their recommendations for the trust relationships to be re-evaluated. The effect of these recommendations is illustrated in Figure 4.11 (d). Node SvN92 was one of the nodes that were found exhibiting malicious behaviour. Notice that the degradation of trust through the re-evaluations is gradual; in the last round, the trust values for SvN92 are not as low as the supervision nodes recommend. This is attributed to two reasons. The first is the value assigned to $R_{req}$ for the WSN nodes, which makes it necessary to accumulate 3 or 4 negative recommendations —depending on the referral trust to the SvN nodes providing them— before each trust re-evaluation. The second reason is that trust has to be re-evaluated more than once in order to be revoked, since the trust re-evaluation process takes into account the initial trust values and the strength of the evidence that was used for their calculation. As, however, negative recommendations continue to be received, the trust values gradually converge to the recommended ones.

In Figure 4.13 we plot the number of legitimate and malicious trust re-evaluations per round, averaged from 20 simulation runs. Malicious re-evaluations are the ones concerning the $WSN_c$ nodes, which are falsely accused by the malicious SvN nodes. Notice that the number of legitimate re-evaluations at the very first rounds of the third phase is relatively small, which is also attributed to the value of $R_{req}$. Since a number of negative recommendations has to be accumulated before trust re-evaluation, and the legitimate supervision nodes send recommendations about a single malicious node in each round, the recommendations are only stored at the first rounds.

Examining the effect of the defaming attack, the number of malicious revocations is large at the first rounds compared to the number or legitimate revocations. This is the result of the focused defaming attack —it targets only the 10 $WSN_c$ nodes, and of the
aggressive tactic of the malicious SvN nodes, that in each round send recommendations about all falsely accused WSNc nodes to all WSN nodes.

However, malicious trust re-evaluations are gradually reduced as negative evidence is received for the malicious recommenders, their trust values drop below the thresholds (Figure 4.11 (d)), and the sets \{BEvidence\} are filtered to exclude their recommendations. Minimising the effect of malicious recommendations thus depends on how quickly the nodes are notified for the malicious nodes. Using the trust re-evaluation process of the model, the effectiveness of revoking trust from malicious nodes depends on the effectiveness of the supervision node in exposing their negative behaviour. The simulation results suggest that so does the effectiveness of protecting from defaming attacks.

4.8.5 The Effect of the Configuration Parameters

The scenario that was used for the experiments discussed in the previous sections assumed little pre-deployment knowledge on the internal structure of the WSN, and did not utilise most of the parameters that the model offers for controlling both trust evolution and the distribution of the required operations among the nodes. In order to examine how the configuration affects the trust evaluation operations and results, in this section we vary the parameters \(R_{req}, d_i, \{Propagation\}\), and the pre-established trust associations, to define a richer pre-configuration based on the pre-deployment knowledge that may exist.

It is assumed that some knowledge on the structure of the WSN exists. It is composed of clusters of 10 sensor nodes, each having a computationally more powerful cluster head, selected from the set WSNc. This structure was selected because it is considered that it approximates real-world WSN scenarios. For example, it could practically apply to a vehicular or body sensor network, where sensor nodes are collecting data that their cluster head is aggregating and transmitting. The initial configuration of the sensor nodes aims to make the cluster heads responsible for expanding their list of trust associations and for undertaking the computationally expensive trust establishment operations.

Specifically, the changes in the initial configuration (Figure 4.9) of the 90 WSN nodes include:

- Pre-established trust associations for each node are defined only for with cluster head in WSNc. The cluster heads are fully trusted, with \(T_{ic} = R_{ic} = 1.0\). No trust associations are defined with certification authorities or supervision nodes.
- The value of parameter \(R_{req}\) is set to 1.0, in order for the evidence provided by the cluster heads to be sufficient for expanding the trust associations of the WSN nodes.
- The value of parameter \(d_i\) is set to 0.8, so that the referral trust will explicitly degrade for newly established trust associations.

For the 10 WSNc nodes, the only change compared to their initial configuration was their \{Propagation\} set, that includes the WSN nodes within their cluster. As in the first scenario, their pre-established trust associations include the CA and 10 SvN nodes. Apart from these changes on the pre-configuration of the nodes, the simulation experiment was executed exactly as described in Section 4.8.1.
Figure 4.14: Trust relationships at the end of the second phase in the clustered scenario

(a) Node $WSN_{c1}$, Round 400

(b) Node $WSN_9$, Round 400
4.8. EXPERIMENTAL EVALUATION

At the end of the second phase, the established trust relationships of the WSN\(_c\) nodes are similar to the ones obtained in the previous scenario. Comparing the results in Figure 4.14 (a) with the corresponding results in Figure 4.11 (c), the only substantial difference is that higher trust values are assigned to malicious nodes. This is not however attributed to the configuration changes, but to the fact that the pre-configuration of WSN\(_{c1}\) included a malicious node in the mostly trusted recommenders.

For WSN\(_9\), however, the results are substantially different. Notice that, apart from the fully trusted cluster head WSN\(_c\), all other trust relationships are a subset of the ones of WSN\(_{c1}\). Since WSN\(_9\) was preconfigured to depend entirely on its cluster head for recommendations, and the referral trust of the cluster head is 1.0, the trust values computed for all other nodes are equal to the ones that the cluster head recommended. The same does not apply to the referral trust, which is lower due to the degradation parameter \(d_i\). Through assigning it a value less than 1.0, it was ensured that no node will ever be selected instead of the cluster head as a recommender, and that behaviour evaluation results would be accepted only from the mostly trusted supervision nodes. If \(d_i\) was set to an even lower value, like in the example scenario discussed in Section 4.3.3, then all newly established trust relationships could have \(R_{ij} < R_{thresh}\), and the node would depend entirely on its cluster head for recommendations.

The configuration that was applied also had significant impact on the operations required for trust establishment. The number of operations of each type per round, collectively for all WSN nodes, is plotted in Figure 4.15. In comparison to Figure 4.12 of the initial scenario, the operations that the sensor nodes had to execute are both less and lighter in terms of resource consumption. They are less because of the lower value assigned to \(R_{req}\). Since \(R_{req} = R_{ic}\), the sensor nodes require only one recommendation, the one from their cluster head. They are lighter because their most
trusted node includes them in its \{Propagation\} set, and is thus forced to execute the resource consuming trust establishment operations in order to provide the requested recommendations.

From the types of operations in Figure 4.15, the sensor nodes received only inter-cluster recommendations, equal in number to their established trust relationships. At the same time, the cluster heads received all recommendations from the \(SvN\) nodes and performed the certificate validations to resolve their relationships with the other \(WSN_c\) nodes. The intra-cluster recommendations are mainly performed during the first rounds of each phase, when the \(WSN_c\) nodes evaluate their trust relationships following recommendation requests. The inter-cluster recommendations, serving to distribute this information to the sensor nodes, are the ones that are gradually reduced until the end of each phase.

This layered approach to the operations that can be performed by each node, and the operations that are propagated to be performed by other nodes in the network (in this case, \(WSN_c\)), allows distributing the computational cost of trust evaluation operations in a controlled manner. The results of the simulations suggest that the initial configuration served not only to control how the trust relationships would evolve, but also to control the distribution of the computational cost and the resource consumption that the trust establishment operations would incur to each node.

The low value assigned to \(R_{req}\), however, made the WSN nodes more vulnerable to defaming attacks. Figure 4.16 shows that the number of malicious re-evaluations has increased substantially in the first rounds, compared to the results in Figure 4.13. Due to the value of \(R_{req}\), only 2 negative recommendations were required for trust re-evaluation, which lowered the resistance of the nodes to the aggressive tactic of the malicious recommenders. At the same time, however, the nodes were more quickly convinced to perform legitimate re-evaluations and, within first 50 rounds, trust for most malicious nodes was revoked, and the defaming attack stopped affecting the trusts relationships.
4.9 Summary

The requirements that were initially set for the proposed model were intended to address the perceived trust establishment needs of real-world WSN deployments. The trust establishment model proposed in this work has been shown to fulfil its main objectives that it should be applied uniformly throughout various deployments, support through proper configuration the diverse characteristics and needs of sensor nodes, utilise the pre-deployment knowledge that may be available, enable the distribution of the trust management overheads, and be generic and produce utilisable results. The model is hybrid, utilising all role-base trust associations, certificates and behaviour evaluation results as trust evidence, and combining them on common evaluation processes and metrics.

The specification of the model included its structure and interfaces, the trust evidence and trust associations that it supports, a rich set of configuration parameters, the trust management processes and the trust evaluation metrics. The analysis of the model included its evaluation against the requirements, the study of its security and trust properties and of its robustness against attacks, and the estimation of the resource requirements and overheads of the trust management operations. Our analysis and experiments demonstrated that the trust evaluation processes and metrics respect the basic properties of trust and, in addition, enable controlling both trust evolution and the resource overheads. We found that the resource requirements of the model depend highly on the type and configuration of each node, with larger overheads being imposed to highly adaptive or critical nodes that are pre-configured to support certificates as trust evidence or require large amounts of trust evidence for establishing new trust relationships. Simulation results show that the model can uniformly support from unstructured dynamic networks to clustered networks for which rich pre-deployment knowledge exists, and from highly adaptive nodes to static and restricted nodes, that will never during the network lifecycle need to perform certificate validations or combinations of recommendations.
CHAPTER 4. A HYBRID TRUST MANAGEMENT MODEL
Chapter 5

Trust–Based Data Disclosure

Privacy protection in WSNs primarily entails ensuring that sensed information is confined to the network and is accessible only to authorised parties. The notion of privacy does not, however, include only confidentiality and access control; it is related more to controlling the disclosure of personal information in exchange of some perceived benefit, than to ensuring complete secrecy or anonymity. In Section 3.2 the privacy protection mechanisms were categorised into mechanisms for the protection of the communications context, privacy sensitive information gathering schemes, and controlled information disclosure approaches. The category that our work falls into is the controlled information disclosure, which entails data access control and data granularity control, and essentially handles the tradeoffs between service information requirements and restrictions applied by privacy preferences. Controlled information disclosure can be addressed in the middleware and application layer, since it is related to protecting the content of the messages from disclosure to illegitimate entities, and it can be applied at different points of the information flow.

For WSNs, in settings with multiple data collectors, with varying levels of trust associated to them, that offer a variety of services in exchange of raw or aggregated information of varying levels of detail, controlling data disclosure is challenging. Any controlled data disclosure solution for WSNs has to address a number of issues: where to apply the data disclosure mechanisms depends on which entities in the information flow are assumed to be trusted; what data to disclose depends on the data granularity control mechanisms that can be applied; who to disclose data to is not as simple as identity–based access control, due to the WSNs’ dynamic nature and their vulnerability to node compromise; when to disclose accurate data instead of cloaked data depends on the entity requesting it and requires some decision mechanisms.

From the analysis of the related work in the area we found that most existing solutions use privacy policies and privacy tags for each type of data the sensor node handles [HL04] and introduce intermediaries between the data provider and the data requestor to perform data disclosure control or anonymisation operations [MFD03, CKK+08]. However, an assumption that needs to be made when introducing intermediaries is that all devices up to the level where the privacy operations are performed should be trusted. Our motivation for the proposed trust–based data disclosure solution was to exclude intermediaries and to use trust as the facilitator for data access and granularity control decisions that would be performed locally at each sensor node. The objective
was to utilise the metrics produced by the trust management model presented in the previous chapter, while sustaining the semantic consistency of the controlled information disclosure solution.

Controlled data disclosure and trust are closely related fields. The roots of trust management in conventional systems, such as databases and operating systems, are the identity-based access control mechanisms [BFS04]. Trust management was initially developed for the purpose of enforcing access control in open and dynamic environments [ADdV07]. Trust is well suited for data disclosure decisions in WSNs, because of their cooperative nature, their susceptibility to node misbehaviour, their mobility and dynamic topology characteristics, and their need for decentralized data access control. However, although various trust models have been proposed and applied for the provision of other security services in WSNs, such as secure routing [GBS08], secure node grouping, and trusted key exchange [LF07], to the best of our knowledge, trust has not been used for facilitating privacy-related decisions on sensor nodes. Research on trust models for WSNs has focused on behaviour-based trust evaluation, and we consider that the semantics that these models assign to trust are not appropriate for data disclosure decisions. The main reasons are that the trust models that are based solely on behaviour monitoring and evaluation do not exploit the pre-deployment knowledge on the network purpose, that is the most decisive factor for data access decisions, which should be taken based on the deployment needs, and that good node behaviour should not compensate for the distrust that the deployment needs may dictate.

Having studied the characteristics and the controlled data disclosure requirements of WSNs, and having analysed the related work on the area, we summarise the problem that we are trying to tackle to:

Addressing the need for a solution for performing localised data access and data granularity control at the points of data capture in distributed and dynamic WSN deployments according to the deployment needs, the network purpose, and the context of the data requests.

The solution should enable both data access and data granularity control, facilitating the decisions both about if and about what data will be disclosed. It should enable localised control, at the points of data capture, avoiding the use of trusted intermediaries. It should be applied in distributed and dynamic WSNs, and therefore be flexible enough for handling requests from unknown or compromised data requestors, excluding the use of privacy policies and tags, and using trust instead as the facilitator for the data release decisions. The context of the requests should be taken into account, so as to support special exceptions for emergencies in crisis situations, where safety outweighs privacy needs [HL04].

In this chapter we present an application level solution for controlling data disclosure at the points of data capture or generation. The solution utilises the hybrid trust management model. The main characteristic of the model that makes it suitable for data release decisions is that it enables the exploitation of the pre-deployment knowledge in order to control trust evolution according to the deployment needs and the network purpose. We utilise the model for controlled data disclosure by specifying the semantics of the trust metrics and the trust evidence, by defining distinct trust classes for the nodes requesting data, and by specifying how the trust status dictates
5.1 Scope and Objectives

5.1.1 Data Disclosure Requirements and Approach

In the following paragraphs we analyse the requirements for the controlled data disclosure solution as they result from the problem statement in the previous section, and we present and document our approach for addressing them.

Application to distributed and dynamic WSNs The solution should take into account the special characteristics and limitations of WSNs. It should therefore be lightweight, decentralised, flexible in handling requests from unknown or compromised data requestors, and able to control the disclosure not only of raw sensed data but also of aggregated data of varying levels of detail. We consider that the solution can not use only authentication and privacy policies enforcement mechanisms for data disclosure decisions, since these mechanisms can not sufficiently address the issues of handling requests from nodes that are not known during the pre-deployment phase, and of being able to identify and reject requests from nodes that have been compromised. Our approach for meeting this requirement is to utilise trust as the facilitator for data disclosure decisions. Trust can offer both the required flexibility in identifying and classifying the data requestor nodes, and increased expressiveness and scalability in comparison to traditional access control systems [ADdV+07]. Moreover, as demonstrated in the previous chapter, the hybrid trust management model can be applied to distributed and dynamic WSNs. Instead of utilising privacy tags to characterise if each type of raw or aggregated data that the nodes handle will be disclosed according to privacy policies, our approach is to utilise metrics that are comparable to the trust values for the classification of the data requests.

Control for both data access and data granularity The solution should facilitate the decisions both about whether the data will be released and about whether it should be cloaked before disclosure. Although WSNs are purpose-specific, the nodes may not be used only for the purpose they were deployed for, or may not only cooperate with the nodes they were preconfigured to trust. During their lifetime, sensor nodes might cooperate with partially trusted deployments for the provision of services that do not require accurate or fully identifiable data. In the example that is later used for the simulation experiments, a vehicular sensor network deployed by the manufacturer
for monitoring the engine and the status of the vehicle is also requested to provide information to a traffic control network. Our approach for enabling data access and data granularity control is using the trust assigned to each data requestor to determine if the data or only a sample of it will be disclosed, or if the request will be rejected. However, what sample of the data will be disclosed to partially trusted requestors depends on the type of the data and on the information needs of the service. For this reason, the solution allows the use of various mechanisms, including negative surveys and data generalisation mechanisms for data cloaking.

**Localised control at the points of data capture**  This requirement is set in order to avoid the use of trusted intermediaries and the communication the requested data in the network towards them. Our approach for addressing it is devising the controlled data disclosure solution to operate as an independent application-level security service, that can utilise the results of the hybrid trust management service and provide interfaces to other applications for requesting the data that will be disclosed. All data access and data granularity control operations that need to be executed to provide this result are performed internally.

**Decisions according to the deployment needs and the network purpose**  The main determinant for data release decisions is the knowledge on the legitimate information flows, which is essentially what the privacy policies of most policy-based solutions express. Since our approach is to use trust as the facilitator for data release decisions, this requirement relates to the semantics assigned to the trust metrics and the trust evidence utilised to derive them. Since the hybrid trust management model was devised to be generic, our approach for satisfying this requirement is to specify these semantics: The functional trust metric will refer to trust for releasing data if the role-based trust associations of the nodes are pre-configured and the trust managing authorities issue certificates according this specific trust scope. Once sufficient evidence of these two types is available in the network, then the third type of trust evidence, the behaviour evaluation results that are not semantically consistent with this trust scope, are used only as the means to revoke trust from malicious nodes.

**Taking into account the context of the requests**  In certain application fields, WSNs may be capable of identifying emergencies and crisis situations. In these cases, the safety of the entities or objects being monitored may outweigh the privacy needs. For this reason, the solution supports bypassing all the data disclosure control mechanisms when the situation is characterised as an exceptional case or as an emergency.

### 5.1.2 Relation to Other Approaches

The solution presented in this chapter addresses a similar issue as the privacy sensitive information disclosure approaches that were discussed in Section 3.2.3, and especially these utilising privacy policies and preferences for access control in Section 3.2.3.1. However, although the issue that is addressed is similar, the assumptions that are made and the approach that is undertaken for controlling information disclosure are different. Our solution utilises trust instead of privacy policies and privacy tags for access control, and does not assume any trusted intermediaries. Moreover, unlike most solutions in
this category that focus on location data, it is designed to handle any type of sensed or aggregated data and it is targeted for WSNs.

The alternative options for data cloaking that are presented later in Section 5.4 include Esponda’s negative surveys technique [Esp06] and mechanisms for the reduction of data accuracy. The mechanisms that can be used are similar to these for information granularity control presented in Section 3.2.3.2. Since, however, the mechanism that should be used for reducing the accuracy of the data depends on the type of the data and the application field of the network, we do not specify the exact mechanism that the corresponding component will implement.

5.1.3 Assumptions and Limitations

The proposed solution addresses a specific issue in the field of the privacy protection, namely the protection of the content of the messages from disclosure to illegitimate entities. It does not include any mechanisms for protecting the context of the communications from eavesdroppers overhearing the communications or for protecting against location or identity inference. Moreover, the controlled data disclosure service assumes that other security services are utilised to authenticate the data requestor nodes and that, once it has replied to the requesting application with the data that can be disclosed, other security services are utilised to protect the confidentiality of the data when it is communicated. In order to cover the complete spectrum of the privacy requirements of WSNs (as presented in Section 2.3.2), it should therefore be combined with other, lower level solutions.

The proposed solution does not specify the application-dependant mechanisms that it utilises for reduction of data accuracy and for emergency situation identification. Instead, we identify the alternative options and provide support for these by including the corresponding components and interfaces in the structure of the controlled data disclosure scheme. Moreover, we assume that a generic trust management mechanism, like the one presented in Chapter 4, is available to provide information on the trust metrics, and that sufficient trust evidence of the appropriate semantics had been made available during the network pre-configuration phase for the trust metrics to be calculated.

5.2 Structure, Components and Interfaces

The controlled data disclosure mechanism receives as input from the applications that utilise it the identifier of the requestor node and the requested data, and outputs the data that should be provided to the requestor, which may be the original data, cloaked data, or a notification for rejection of the request. The components of the mechanism are structured as in Figure 5.1. It is designed to operate as an independent application-level service, interfacing the hybrid trust management system for obtaining the functional trust metrics of the data requestor nodes through the TRUST-QUERY interface specified in Section 4.4, using the identifier of the requestor node in the TargetID field. It provides an applications interface for making its results available to requesting applications.

For each type of data that the nodes handle, the configuration parameters are used to specify the required trustworthiness of the data recipients and the data cloaking
operations that can be performed. The sensor nodes may produce more than one types of data when they have multiple sensing units, or when they participate in data aggregation operations. Entries should be added to the configuration parameters for each of the types that are known before deployment, since the parameters are utilised by the data disclosure decision process of Section 5.3 and by the data cloaking mechanisms described in Section 5.4.

The data disclosure control operations include:

1. Receiving and replying to applications requests through the DATA-DISCLOSURE-QUERY interface. The primitives that are included in this interface are the DATA-DISCLOSURE-QUERY.req, used by the applications to request data access and granularity control to be performed, and the DATA-DISCLOSURE-QUERY.res primitive that delivers the information that should be sent to the requestor:

```
DATA-DISCLOSURE-QUERY.req(
    RequestorID, /*The identifier of the data requestor node*/
    DataType,  /*The type of the requested data*/
    Data,      /*The requested data*/
    CloakingRestrictions /*Set with possible elements {sampling, generalisation}*/
)
```

```
DATA-DISCLOSURE-QUERY.res(
    RequestorID, /*The identifier of the data requestor node*/
    AccessType, /*Discrete value in {unmodified, sample, generalised, denied}*/
    DisclosedData /*The data that should be disclosed*/
)
5.3. **DATA DISCLOSURE CONTROL**

The CloakingRestrictions set in the request is filled according to the configuration of the requesting application, and contains the data cloaking operations that cannot be performed based on the application information requirements. The AccessType parameter in the response takes values according to the operation that has been performed in the data by the system and may be denied if the data request should be rejected. If a data cloaking operation was performed, the requesting application can utilise the value of this parameter to notify the requestor that the data is a negative sample or has been generalised.

2. Deciding upon the disclosure of the requested data. The operation is invoked after the receipt of each request and executes the data disclosure control process that is described in Section 5.3 to produce the response that will be forwarded to the requesting application.

3. Evaluating the criticality of the situation and identifying emergency cases. The operation is invoked by the data disclosure control process using a request message that includes the DataType and the Data, and replies with a flag denoting if an emergency case has been identified. The implementation of this operation, which is further explained in Section 5.3, depends on the types of the data that are handled by the node and the application field on the network. The operation may need to interface other services (like localisation) or it may need to communicate with other nodes if it applies collaborative context awareness mechanisms.

The data cloaking operations that are included in the system are invoked by the data disclosure control process using the DATA-CLOAK.req primitive that includes the DataType and the Data, and reply using the DATA-CLOAK.res primitive that includes the cloaked data. Section 5.4 describes the implementation of the negative samples generation operation and discusses the implementation of the data granularity control mechanism.

## 5.3 Data Disclosure Control

The data disclosure control process is invoked in node $i$ following an application request. It uses the TRUST-QUERY interface for obtaining the functional trust metric $T_{ij}$, representing the trust that $i$ assigns to the data requestor node $j$ for the function of disclosing to it data, and the DATA-CLOAK interface for applying data cloaking operations.

The process follows Algorithm 5.1 to determine if the data or only a sample of it will be disclosed, or if the request will be rejected, according to the functional trust value assigned to the data requestor $j$. Data requestors are categorised into three distinct trust classes: trusted, partially trusted, and distrusted. The trust classes are devised according to the trust thresholds $T_{\text{disclose,thresh}}$ and $T_{\text{cloak,thresh}}$, that are set in the configuration parameters for each type $t$ of data that the node produces. Each class is handled differently regarding data disclosure. Accurate data is sent only to trusted requestors, while partially trusted requestors are sent data which has been cloaked. Data requestors are categorised in the distrusted class either if their functional trust metric is below the cloaking threshold, or if their trust association remains unresolved after the trust establishment process that was initiated if $j$ was an unknown node following the TRUST-QUERY request.
Algorithm 5.1: Controlling data disclosure in node \( i \) upon receipt of an application request to release data \( D_t \) of type \( t \) to node \( j \) with data cloaking restrictions \( R \)

```plaintext
if isEmergency\((t, D_t)\) //Bypass data disclosure control in emergency cases
    return \((j, \text{"unmodified"}, D_t)\)
else
    Retrieve Configuration Parameters \( CP_t \) for data type \( t \)
    Issue a TRUST–QUERY.req \((\text{me.id}, j)\) to obtain \((j, \text{Class}_j, \text{T}_{ij})\)
    if Class\(_j\) = “unresolved” or \( T_{ij} < CP_t.T_{cloak} \) //Distrusted class
        return \((j, \text{"denied"}, \text{null})\)
    else if \( T_{ij} \geq CP_t.T_{disclose} \) //Trusted class
        return \((j, \text{“unmodified”}, D_t)\)
    else //Partially Trusted class, \( T_{ij} \in [CP_t.T_{cloak}, CP_t.T_{disclose}] \)
        //Determine and apply the appropriate cloaking operation
        if \( CP_t.NegativeSampling = \text{true and } \text{"sampling"} \notin R \)
            Issue request to Negative Samples Generation to obtain Sample\((D_t)\)
            return \((j, \text{“sample”}, \text{Sample}(D_t))\)
        else if \( CP_t.Generalisation = \text{true and } \text{“generalisation”} \notin R \)
            Issue request to Data Granularity Control to obtain Generalised\((D_t)\)
            return \((j, \text{“generalised”}, \text{Generalised}(D_t))\)
        else //No cloaking operation is applicable
            return \((j, \text{“denied”}, \text{null})\)
    end if
end if
```

CHAPTER 5. TRUST–BASED DATA DISCLOSURE
For partially trusted data requestors, the algorithm decides on the cloaking operation that is appropriate based on the type of the data (retrieved from the configuration parameters) and the information needs of the application (retrieved from the CloakingRestrictions set of the request). If no restrictions are applied by the request and any cloaking operation can be performed on the requested data type, then priority is given to the application of negative sampling mechanisms. The reason is that negative sampling is computationally simple for the sensor nodes, and it enables cloaking without requiring information on the anonymity set size. If no data cloaking technique is found to be applicable, then the request is denied, since the requestor is not trusted enough to receive accurate data.

The process enables ignoring the data access control operations and any trust establishment operations that may be invoked in emergency cases. The $\text{isEmergency}(t, D_t)$ is implemented by the Emergency Situation Identification component. The issue of how to characterise a situation as an emergency depends on the application space of the network and the type of the data, so we avoid specifying how the function is implemented. Instead, we identify two options. If the type of the sensed data and the type of the monitored subject enable defining a range of emergency values, then the value of the data itself can be used for this decision. Examples of this option are sensor readings of the heart rate of patients or of the temperature in sensitive products storage facilities, where an emergency case could be identified only by comparing the $D_t$ with the acceptable range.

A second, more complex option, is to combine context awareness techniques and emergency signals from other network nodes in order to cooperatively decide on the criticality of the situation. This is an issue of context awareness in the privacy protection mechanisms; context-aware privacy was described by Mitseva et al. as the application of privacy protection mechanisms which adjust to the current relevant contexts and take decisions accordingly [MWP08]. In their scheme, context attributes are used to describe a snapshot of the current context from the status of all relevant interacting entities. Context reasoning mechanisms enable the deduction of high-level context using high-level context rules and low-level context, i.e. basic contextual information directly measured by the sensors. We refer the reader to reference [MWP08] for more information on the formation and evaluation of context rules that could be used in order to cooperatively characterise emergency cases. If context awareness mechanisms are utilised, then the Emergency Situation Identification component will need to access other services and communicate with other nodes through the network interface for cooperatively identifying critical situations.

### 5.4 Data Cloaking Operations

The data cloaking operations enable the sensor nodes to provide information to partially trusted deployments for the provision of services that do not require accurate data. As shown in Algorithm 5.1, the scheme enables the application of alternative techniques data cloaking depending on the type of the data and the information needs of the service requesting it. In the following sections we describe the negative surveys technique, which enables the disclosure of negative samples of the data instead of the accurate measured data, and discuss the application of data granularity control mechanisms that
can be applied for reducing the data accuracy.

5.4.1 Negative Surveys

The negative surveys technique was introduced in reference [Esp06] and used for anonymous data collection in WSNs in reference [HGFE07]. It can be applied for any type of data that can take discrete values from a finite set or range of values. Using this technique, the nodes transmit negative samples of their data, i.e. random false values from the finite set, instead of their actual sensed data. The data requestors can use the false values collected from a number of nodes to retrieve the distribution of the actual data. In reference [HGFE07] it is derived that, for each discrete value $i$, if $n$ is the total number of reporting nodes, $t$ is the size of the values set, $R_i$ is the number of nodes that falsely reported $i$, and $C_{i,j}$ is the expected number of sensor nodes in category $j$ that report $i$, the estimate of the number of nodes that would truthfully report $i$ is:

$$A_i = \sum_{j \neq i} \left( R_j - \sum_{k \neq i, j} C_{j,k} \right)$$

Since each node has selected its false reported value from the values set with probability $\frac{1}{(t-1)}$,

$$\sum_{k \neq i, j} C_{j,k} = \frac{1}{(t-1)} \sum_{k \neq i, j} A_k$$

By observing that $\sum_{j \neq i} R_j = n - R_i$ and $\sum_{j \neq i} A_j = n - A_i$, it is finally derived that:

$$A_i = n - R_i(t - 1)$$

This technique therefore enables estimating the number of nodes that sense each discrete value without requiring the truthful data. The accuracy of the estimates was found to increase the more nodes participated and the smaller the size of the values set.

For partially trusted data requestors, this technique is applied by the Negative Samples Generation operation of Figure 5.1 upon the receipt of DATA-CLOAK requests. The configuration parameters for the DataType of each request contain the Negative Sampling Parameters, which include the range of all possible values that the requested data type can take. The implementation of this operation is simple for the sensor nodes: they have to randomly select a value from the set that is different than the Data in the request, and reply with the Sample($D_t$) that is used by Algorithm 5.1.

The negative survey technique is computationally simple for the sensor nodes, since it only requires reporting a false value. However, it is appropriate only for the provision of services for which the distribution of data is important rather than the answers from specific sensor nodes, and for types of data that take discrete values from finite sets.
5.4.2 Reduction of Data Accuracy

The reduction of the accuracy of the data is the alternative approach that can be used for partially trusted data requestors either when the data values are not discrete or constrained in a finite set, or when the data requestors need the readings from specific sensor nodes.

In the database field, this approach mainly involves data generalization, i.e., replacing a value with a less specific but semantically consistent value in order to achieve \( k \)-anonymity \([\text{Swe02}]\). The in-network reduction of the accuracy of dynamic and real-time data, however, raises the issue of how to determine when data is sufficiently generalised. For some types of data, like body temperature, mechanisms for the reduction of data accuracy could produce a temperature range instead of the actual measurement, with the size of the range depending on the trust value towards the requestor. For spatial and temporal information, however, that can make the subject directly identifiable, the reduction of data accuracy is not trivial, since it should depend on the anonymity set size \([\text{GG03}]\), which is not distinguishable to the sensor nodes that have to perform this operation.

The technique that is used for reducing the accuracy of the data therefore depends on the type of the data and the application field of the network. This is the main reason why the solutions for information granularity control for ad hoc and sensor networks, including those presented in Section 3.2.3, are rarely data and application-independent. We do not specify the exact mechanism to be implemented by the Data Granularity Control operation to provide the Generalised(\( D_t \)) value to Algorithm 5.1. Depending on the implementation, this operation may need to access other services, like localisation, or communicate with other nodes through the network interface, in order to derive information on the anonymity set size.

The reduction of the accuracy of the data can be computationally simple for the sensor nodes only if its performed naively, without ensuring \( k \)-anonymity. Both approaches for data cloaking therefore have their limitations in terms of their application on sensor nodes, on the types of data they support, or in addressing service information needs: If the service requires the data from specific sensor nodes, then the reduction of data accuracy may be performed naively by the sensors; the negative surveys technique enables cloaking without requiring information on the anonymity set size, but is appropriate only for the provision of services for which the distribution of data is important. The solution enables the use of any of the two approaches, as long as the data is flagged as shown in Algorithm 5.1, so that the recipient can identify what type of cloaking was performed.

5.5 Experimental Evaluation

In order to evaluate the effectiveness of the proposed data disclosure scheme in classifying data requests, we implemented the decision mechanisms in the simulation environment that was introduced in Section 4.2. A simulation scenario was devised to approximate a realistic WSN scenario, so as to examine how the established trust relationships affect the data disclosure operations. For our experiments we assume a network with three groups of nodes, each having a different purpose and belonging to a different deployment. Figure 5.2 depicts the logical grouping of the network nodes in
the simulation scenario. The group that is the focus of our experiments is a cluster of sensor nodes that we assume is a vehicular sensor network (VSN) consisting of 5 mutually trusted nodes ($T_{VSN,VSN_i} \in [0.9,1.0]$), of which one, $VSN_h$, is the cluster head. For simplicity, we assume that each VSN node can provide information of only a single type, and that negative sampling is supported for all types of data and accepted by all data requestor nodes.

The other two groups are assumed to be environmental sensor networks. The first, the ServiceESN, is deployed by the vehicle manufacturer in one of the vehicle service points. It is unknown to the VSN, but certified by the manufacturer’s trust managing authority. 10 of the 50 mutually trusted ServiceESN nodes hold a valid certificate. The second group, TrafficControlESN, also consists of 50 mutually trusted nodes, of which only a subset (10 nodes) are known and partially trusted by the $VSN_h$ node, with $T_{VSN,hj} \approx 0.6$. Regarding the trust management service configuration of the VSN nodes, it is set that $R_{\text{threshold}} = 0.5$, $R_{\text{req}} = 0.9$, and $d = 0.9$. The simulation is run for 100 rounds. In each round, thirty data request messages are sent from a random requestor to a random VSN node. Algorithm 5.1 is executed with the receipt of each request message, and statistics are accumulated at each node for its trust establishment and data disclosure operations.

As performance metrics for the trust establishment operations, we are interested in the types of the established trust relationships and the value of the trust metric for each relationship. Figure 5.3 depicts the number of trust relationships of each type collectively for all VSN nodes, averaged from 5 simulation runs. The results suggest that the number of unresolved trust relationships decreases gradually at each round as requests are received. The reason for this is the configuration of the VSN nodes: They are configured to have acceptable referral trust $R_{ij}$ only for their cluster head, $VSN_h$. Since the $\{\text{Propagation}\}$ parameter is not being used, they have to wait for $VSN_h$ to resolve its trust associations before it can provide them with recommendations for the rest of the nodes. This configuration was made to avoid the resource consuming certificate validations and intra-cluster recommendations at the rest of the VSN sensor nodes.
Figure 5.3: Types of trust relationships of VSN nodes per round

Figure 5.4: Trust relationships of node VSN\textsubscript{2} on round 300
Figure 5.4 illustrates the resulting trust values for the relationships that were established at the end of the simulation by one of the VSN nodes, and the classification that was made for data disclosure operations according to the thresholds $T_{\text{disclose}}^\text{thresh}$ and $T_{\text{cloak}}^\text{thresh}$. High trust values are assigned to the 10 ServiceESN nodes that provided as evidence to VSN the valid certificate from the fully trusted trust managing authority. The rest of the ServiceESN nodes are less trusted because of the length of the recommendation path $VSN_2 \leftarrow VSN_h \leftarrow ServiceESN_{x \in \text{Certified}}$. To illustrate the effect of the degradation parameter on relationships with a recommendation path like this, the crosses on the ServiceESN bars depict the results of the simulation when $d$ was set to 0.7. Most TrafficControlESN nodes are considered partially trusted, since 10 of them were partially trusted by the cluster head, and their trust values also diluted through transitivity.

Figure 5.5: Types of replies of VSN nodes to data requests per round

It is of special interest to examine how the established trust associations affected the data disclosure operations. Figure 5.5 depicts the percentage of each type of reply (disclosure of accurate data, disclosure of cloaked data, or denial) out of the total replies, collectively for all VSN nodes, averaged from 5 simulation runs. The figure also shows what the percentage of disclosures (of both accurate and cloaked data) would be for the pre-known requestors only, without the trust establishment operations. During the first rounds, the results demonstrate that most data requests are denied, since the trust associations can not be resolved yet by the VSN nodes. Until the end of the simulation, the nodes gradually gain enough information to determine when to disclose either accurate or cloaked data, according to the trust values of Figure 5.4. When the trust relationships are resolved with the ServiceESN nodes, they receive the accurate sensor readings that they request, while the TrafficControlESN nodes receive only cloaked data.
5.6 Summary

The application-level controlled data disclosure solution proposed in this chapter positions trust as the facilitator of the data access and the granularity control decisions of sensor nodes. It utilises the results that the hybrid trust management model produces, in order to benefit from the flexibility that it offers in handling unknown and malicious nodes. It fulfils its main objectives that it should be applied to distributed and dynamic WSNs, support both data access and data granularity control at the points of data capture, and control disclosure according to the deployment needs and the network purpose.

The specification of the solution included its structure and interfaces, the data disclosure control operations for classifying the data requestors and determining the data cloaking operations to be performed, and alternative options for cloaking the data before publishing it to partially trusted requestors. These options include negative surveys and data granularity control mechanisms, to be applied according to the type of the data and the information needs of the service requesting it. Limitations were, however, identified in both approaches: The first is appropriate only for the provision of services for which the distribution of data is important rather than specific answers from sensor nodes, while the latter can only be performed naively by the sensors, without taking into account the size of the anonymity set. As a result, data cloaking is an aspect of the scheme where a generic, application-independent, solution could not be provided.

The solution was evaluated through simulation experiments on a scenario with both trusted and partially trusted nodes. We examined the established trust relationships and their affect on the data disclosure operations. The simulation results demonstrated that, through proper configuration, the hybrid trust management solution provides adequate information for identifying how requests from unknown data requestors should be handled.
Chapter 6

Security Management for Heterogeneous WSNs

The trust management model and the controlled data disclosure scheme that were presented in the previous chapters assume the provision of core security services to guarantee the authenticity, integrity, and confidentiality of the data generated and communicated through the network. For this reason, in this chapter they are integrated in a security management framework that aims to provide a complete security solution for WSNs. The analysis of the security requirements of WSNs in Section 2.3.3 highlighted that security services management for these types of networks is not trivial. The integration of WSNs with other networks and platforms and the heterogeneity between the capabilities and the security, trust and privacy requirements of the nodes and the deployments necessitate the provision of flexibility and adaptability properties.

From the analysis of the related work on the area we found that the issue of security management in WSNs has not been sufficiently addressed. Although there exist security management solutions offering flexibility [PR04], adopting a configurable toolbox approach [WGS06], exploiting the heterogeneity of the nodes [MWY08], or enabling the integration of WSNs with other platforms [FDM+05], we did not find any solution for WSNs combining all these characteristics, offering flexibility and adaptability, and including all security, privacy and trust services.

Having studied the characteristics and the security requirements of WSNs, and having analysed the related work on the area, we summarise the problem that we are trying to tackle to:

Addressing the need for a localised, flexible and adaptive security management solution for heterogeneous and integrated WSN deployments, that facilitates the provision of core security functionality along with trust management and privacy protection services.

The solution should be localised, applied in each network node, without requiring the existence of centralised management points. It should be flexible and adaptive in order to support the heterogeneity in the capabilities and the security requirements of the nodes and the deployments and the integration of WSNs with other networks. Finally, it should be a complete security management solution, facilitating the provision of all security, trust and privacy services.
In this chapter we present a generic security management framework that integrates the trust management and controlled data disclosure mechanisms, and allows customisation of WSNs to a diverse set of application spaces. The framework is designed to be flexible and adaptive through following a modular approach for the integration of the various configurable protocol and control elements into a complete toolbox solution. We present performance evaluation results that approximate the benefits of the security framework, mainly in terms of resource consumption, for a variety of scenarios. Moreover, we provide guidelines for matching the security requirements of the nodes and the deployments to the security framework configuration.

Throughout this chapter, we utilise a working WSN scenario from the medical field for demonstrating the application of the security management components. The scenario involves body sensor networks (BSNs), which are deployed for remote monitoring of the patients’ condition and report their data through mobile terminals. The terminals communicate with the hospital’s base station through either external networks or environmental sensor networks (ESNs). The data that is collected is both related to the vital functions of the patients and to their relative location. This scenario was selected because it entails nodes with diverse capabilities (sensor nodes and gateway nodes), which imposes the need for a customisable security solution, and data with diverse security requirements according to its type and to the context of each communication, which imposes the need for context awareness and adaptability in the security mechanisms.

It should be noted that the framework was developed in collaboration with other researchers (explained in Section 1.4) during the course of the e-SENSE research project. The main objective of the project was the integration of ubiquitous WSNs in the future ambient intelligent mobile systems beyond 3G by following a toolbox approach [GPS+06]. The solution that is presented in this chapter introduces some modifications to the one reported in references [MAM+08, AMS+07b]. The trust management components and the processes and metrics that they implement are different, and the controlled data disclosure components are the ones presented in Chapter 5.

In the following section, we specify the scope and objectives of our work, including the requirements for the security management solution, our approach, the limitations of our work and the assumptions that we have made. In Section 6.2 we present the security management framework and specify its structure, the mechanisms for providing adaptability and flexibility, the components and their interfaces, and the framework versions with their corresponding sets of security protocols and mechanisms. Section 6.4 describes how the configuration of the security framework with appropriate elements can be performed according to the security requirements of WSN deployments. Finally, Section 6.5 includes the analysis and evaluation of the solution.

6.1 Scope and Objectives

6.1.1 Security Management Requirements and Approach

In the following paragraphs, we analyse the requirements for the security management solution as they result from the problem statement in the previous section, and we present and document our approach for addressing them.
6.1. **SCOPE AND OBJECTIVES**

**Application to WSN deployments** The security management framework should be applied locally at each network node, enabling the enforcement of pre-defined settings and policies, without requiring the existence of centralised management points. At the same time, the security mechanisms that it requires the sensor nodes to execute should entail acceptable resource consumption. In order to meet this requirement, we select security primitives that have been evaluated and have been found to be applicable on sensor nodes, and include only these in the framework components. For the selection of the primitives, we utilise the experimental evaluations that were discussed in Section 3.3.1.3. Moreover, through the adaptability property that the framework features, the security protocols and mechanisms that are used for each communication after the network deployment are selected to be the most lightweight for providing an adequate level of security according to the context of the communication.

**Complete security management solution** The security management framework should cover the complete set of security requirements of the devices within the network by facilitating the provision of all security, trust and privacy services. The basic issues that it needs to address are data confidentiality and integrity, authentication and data access control, and management of the trust relationships. Our approach for meeting this requirement is to integrate in the solution core security services along with the data disclosure control mechanism and the trust management model that were introduced in the previous chapters. Since trust management is utilised for the purpose of data disclosure control, the semantics that are assigned to trust are the ones described in Section 5.1.1.

**Flexibility** This requirement stems from the diversity that exists in the capabilities and the security requirements of the nodes in WSN deployments. As explained in Section 2.3.3, diversity exists in the types and roles of sensor nodes utilised, their computational capabilities, their mobility model, the possibility of their regular maintenance, and the type of information they collect. The framework should be flexible in providing lightweight security services while effectively covering the diverse capabilities and security needs of the nodes and the deployments. Our approach for meeting this requirement is designing the framework as a modular, toolbox solution, composed of configurable components. This way, it can be configured to provide the best suitable levels of security, trust and privacy functionality for different node architectures, hardware limitations, security requirements, and application spaces. Essentially, this approach enables different versions of the framework to be deployed, providing varying levels of security functionality, according to the role, capabilities and security requirements of the nodes.

**Adaptability** Changes in the environment in which communications take place may affect their security requirements. After the network deployment, different levels of security may need to be applied to the communications according to their context. For example, mobile nodes may require stronger security mechanisms to be applied when they are in certain high-risk locations. This imposes the need for security mechanisms which can adapt to the context of each communication. The adaptability property is set to ensure that the system works at the best of its capability, taking into account the
trade-off among device constraints, change in context, and different policies and preferences. The adaptability property of the security management framework is attained through defining different security levels with their corresponding primitives, which can be simultaneously supported by each node, and enabling the use of context awareness mechanisms for selecting the appropriate security level for each communication. Essentially, this approach enables different security protocols and primitives to be supported by each node and applied according to the context of the communications.

Support for integration with other networks The security management framework should enable the secure integration of WSNs with other networks, supporting communication flows that are not only between sensor nodes of the same network, but also with external entities through network gateways. In the working WSN scenario, the boundaries between the BSNs and the external networks they are interconnected with are the mobile terminals gathering the data from the patients’ BSNs. In order to address this requirement, the security management framework should be applied both in the WSN nodes and in the gateways, to secure the communications up to the boundaries with the other networks. The flexibility property of the framework enables different versions of it to be deployed in the gateways, given that they have both different hardware capabilities, and communication and security requirements.

6.1.2 Relation to Other Approaches

The approach that we adopt for the adaptability property of the framework was inspired by the adaptive security solution proposed for low data rate networks in reference [PR04]. This work introduced a security manager component which integrates the security elements, handles three security levels, decides on the security level that is adequate for each service, and enables the network to adapt to the different needs of devices and services. Moreover, similarly to the security architecture proposed for WSNs in reference [WGS06], we adopt a modular toolbox approach for the flexible integration of various security services. The security management framework that is presented in this chapter combines these characteristics to make the enforced security mechanisms dependant both on the context of the communications and on the capabilities and requirements of the nodes.

Similarly to the PLUS framework [YKD08], we include a trust management component in the solution, but we specify that its purpose is to facilitate data disclosure control operations. Moreover, as in reference [MIWY08], the gateway nodes in the framework may perform more computationally complex operations than the sensor nodes. However, neither of these solutions aims to cover the complete set of security requirements of WSNs, supporting the provision of all security, controlled data disclosure and trust services. Moreover, the security management framework is based on the one reported in references [MAM+08, AMS+07b], but it is modified as explained at the introduction of this chapter.

6.1.3 Assumptions and Limitations

The security management framework was developed to operate as a component in the management subsystem of the e-SENSE protocol stack architecture [GPS+06]. It could,
however, be integrated in other architectures, if some conditions are satisfied. The framework assumes that other services, like location and positioning, exist to provide information to the context awareness mechanisms. Moreover, the architecture should be able to integrate security primitives and protocols, and should provide the appropriate interfaces to the security management component for controlling their enforcement.

Regarding the scope of the proposed solution, it should be noted that it concerns only the WSN security issues that can be addressed by a security management service. While it provides support for controlled data disclosure, trust, and core security services including authentication, confidentiality and integrity protection, is does not address security issues that would require interference with the routing protocol or the network design. The security management framework does not introduce new security mechanisms or protocols, but is rather a solution for the integration of existing mechanisms. Moreover, the framework provides support for the application of context awareness mechanisms but, since these mechanisms are application-dependant, it is out of its scope to specify how they are implemented.

6.2 Structure, Components and Interfaces

The proposed security management framework is implemented by a security manager component, positioned in the management subsystem of the e-SENSE protocol stack [GPS+06], as depicted in Figure 6.1. The e-SENSE protocol stack architecture is decomposed into four logical subsystems, namely the application subsystem, hosting one or several sensor applications, the middleware subsystem, providing data transfer services for the transport of the application data packets, the connectivity subsystem, consisting of functions required for operating the physical layer, the medium access control, and the network and transport layer, and the cross-layer management subsystem, responsible for the configuration and initialization of the stack. Each subsystem comprises various protocol and control entities, which offer services and functions at service access points to other subsystems.

The cross-layer security manager includes services at different levels of the protocol stack related to data security, disclosure control and trust management. A modular toolbox approach is adopted, with some of the components being optional or having limited functionality for scaled-down versions of the framework, that can be applied to resource constrained nodes that have limited security requirements.

The security protocols and mechanisms component positioned in the connectivity subsystem is the most fundamental, since it is the one required for all types of nodes in the network. It contains the security primitives necessary to implement several security protocols in a way that is transparent to the layers above it in the protocol stack. The adaptability of the security mechanisms is ensured by the security agent, responsible for determining the most suitable security mechanisms and protocols for every message exchange, as explained in Section 6.3.1.

The controlled data disclosure agent integrates the data disclosure control and the data cloaking operations of the trust–based data disclosure system of Figure 5.1, and makes its results available to the applications through the DATA-DISCLOSURE-QUERY interface defined in Section 5.2. The hybrid trust management agent integrates the trust management operations and the trust evidence management operations of Figure
Figure 6.1: The security framework within the e-SENSE protocol stack — scaled-down and extended versions

4.8 interfaces the middleware subsystem for communicating trust evidence, and makes the trust metrics available through the TRUST-QUERY interface defined in Section 4.4.

The privacy and trust assistant is an application support component, to be implemented only in user-related full-function nodes (for example smart phones or PDAs), to provide an interface for the configuration of the trust and controlled data disclosure parameters, described in Sections 4.3.3 and 5.2 respectively.

The profile and rule agent interfaces the security, controlled data disclosure and trust agents for the provision of related information that has been stored in the node. It interfaces the data store to get or modify the trust records or the profile or service information. The information in the profiles and rules is utilised by the security agent to configure the security mechanisms that are applied. The profiles may be related to the scenario, the node status, the service, or the related context of the communications. The rules concern the security mechanisms that should be applied according to the current profile. The context information that may be required by the security agent to evaluate the current profile comes from service and node discovery, location positioning and the applications.

Within the protocol stack implemented in any sensor node, cluster head or gateway, the security manager interfaces various layers. The security agent interfaces the connectivity layers for the configuration of the security protocols and mechanisms component. The trust management agent interfaces the middleware layers for the exchange of evidence-related messages with peer nodes. The controlled data disclosure agent interfaces the application subsystem to receive requests and publish its results.
6.3 Flexibility and Adaptability of the Security Manager Components

Flexibility is ensured by enabling the role, capabilities and security needs of each network node to determine the subset of security manager components which shall reside on the node, as shown in Figure 6.1. Extended versions of the security framework apply to the coordinators and the gateway nodes, as well as to sensor nodes that are dynamic or produce data that is sensitive, and require the provision of trust-based controlled data disclosure functionality. The scaled-down version has a simpler controlled data disclosure component, and misses the trust management components, the privacy and trust assistant and the data store. Instead of the data store, the scaled-down version has an access list in the profile and rule agent, that contains the minimum required policies, profiles and data access rights.

Except from the components that might be omitted, others are customisable and allow for lighter versions to be deployed, to provide only a subset of the services defined. The security protocols and mechanisms component may contain the mechanisms for only a subset of the Security Levels (SLs) supported by the framework. For highly constrained nodes with strictly defined role, one SL may suffice for its communications with its cluster head. Similar customisations can be made to the trust management and the data disclosure components. The versions presented on Figure 6.1 are therefore not strict regarding their components, since additional intermediate versions can be deployed.

Adaptability is attained through enabling the security protocols and mechanisms component of each node to be equipped with different mechanisms, which are selected by the security agent according to the context of the communications. Three SLs are defined, as presented in Table 6.1:

1. The low SL provides non-privileged services and allows exchange of non-sensitive data with weak integrity protection.
2. The medium SL provides limited protection, but is lightweight enough to be used even if the data exchanged within the WSN is not sensitive.
3. The high SL provides privileged access to service and secures the exchange of highly sensitive data with freshness protection.

The SL that is applied to each communication is the minimum accepted by both parties, agreed upon by their security agents. In the following sections we describe the security, controlled data disclosure, and trust components and the mechanisms that each uses to attain the flexibility property and, for the security agent, the adaptability property.

6.3.1 Security Agent

The security agent enables configuring the SL that will be applied for each communication. The SL is re-evaluated whenever there is a change in the network state, the device state, the surrounding context or when the user requests other level of data protection then the current one. It is determined by applying rules to the current scenario, node,
service and context profiles. The rules and the context awareness mechanisms that are used for evaluating the current profiles are not defined in the framework, because they are application-dependant. For example, in the working WSN scenario, where a mobile terminal acts as the gateway of each BSN, the profiles of the terminal could include the remaining energy, its relative location, the health status of the patient, and the service or network role of the node requesting data. Rules could be defined to assign the low SL for the communications with the BSN when the patient is at home and the high SL when he is in public places and the remaining energy of the sensor nodes is above a threshold\(^1\).

The SLs determine the mechanisms and protocols that are used to provide authentication, encryption, message freshness and integrity for each request. Table 6.1 includes a list of mechanisms that can be used for each security level. The list is indicative, since the mechanisms for each security level should be decided according to the criticality and the security needs of the deployment scenario. The characterisation of the sensor devices in the table is made according to their classification in reference [RAL07]. The protocols and mechanisms that the list includes for all security levels have been found applicable for sensor nodes through the experimental studies that were discussed in Section 3.3.1.3.

For low levels of security, symmetric network and group keys could be used for authentication, similarly to the use of keys with different scopes in the LEAP protocol [ZSJ03]. For medium levels of security, symmetric link keys are proposed for two-party authentication, together with \(\mu\)TESLA for authenticated broadcast through delayed key disclosure [PST+02], while the highest security level includes the use of ECC. An approach for introducing flexibility in the encryption process is to use combination of different parameters of RC5 as presented in reference [RDL+06]. The assumption is that shorter the key lengths are and the smaller the number of rounds is, the weaker

---

1We refer the reader to reference [AMS+07b] for a detailed analysis of the profiles and rules that can be set in the example scenario.
6.3. FLEXIBILITY AND ADAPTABILITY

The three SLs could be differentiated through a combination of block length, different number of rounds and different key length. For example, the medium SL could apply RC5 in CTR mode with 32 bits block length, 6 rounds and 24-bit key, while the high SL could apply RC5 with 32 bits block length, 12 rounds and 40-bit key.

The security agent and the components that are related to it are configurable and allow for lighter versions to be deployed. The security protocols and mechanisms component may support only a subset of the SLs, while the complexity of the security agent decision mechanisms depends on the profiles and rules that have been defined. Constrained nodes with strictly defined network role, like the sensor nodes of the BSNs in the working scenario, could be configured to apply the low SL by default and include only a limited set of profiles and rules. Since the SL that is applied to the communications is the minimum accepted by both parties, the sensor nodes could depend entirely on the mobile terminals for increasing the applicable SL according to their own context evaluation outcomes.

6.3.2 Controlled Data Disclosure Agent

The controlled data disclosure components address the flexibility requirement of the security management framework by being customisable and allowing for lighter versions to be deployed. The controlled data disclosure agent obtains information from the profile and rule agent using the PRIV-INDICATION interface, and makes its results available through the DATA-DISCLOSURE-QUERY interface defined in Section 5.2. In the scaled-down version, its functionality is simple: it compares the identity of the requestor node and the type of the requested data against the entries in the access list to decide if the requested data will be disclosed, and does not enable any data cloaking operations.

In the extended version of the framework, the controlled data disclosure agent integrates the data disclosure control and data cloaking operations of the trust–based data disclosure system of Figure 5.1, while the configuration parameters depicted in the figure are stored in the profile and rule agent. The data cloaking operations and the emergency situation identification mechanism that the controlled data disclosure agent includes may require input from other services, which can be obtained through the interfaces that the security manager has with the other components of the management subsystem. For nodes that are constrained but require more flexibility than the access list can offer, or require the extended version of the framework, the controlled data disclosure agent can be configured to be more lightweight. If the data disclosure threshold is set to be equal to the data cloaking threshold for all types of data, as discussed in Section 5.3, no data cloaking mechanisms need to be included in the node.

6.3.3 Hybrid Trust Management Agent

The trust management components are required for nodes that, during the network lifecycle and without reconfiguration, will need to provide information to nodes other than these that they were initially configured to disclose data to. In the scaled-down version of the framework, the hybrid trust management agent is omitted, since the controlled data disclosure agent utilises the access list instead of trust values.

In the extended version of the framework, the hybrid trust management agent integrates the trust management operations and the trust evidence management operations.
Table 6.2: Security management framework configuration issues and affecting parameters

<table>
<thead>
<tr>
<th>Configuration Parameters</th>
<th>Configuration Decisions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node role in the network</td>
<td>Security manager version (^1)[Scaled down/Extended]</td>
</tr>
<tr>
<td>Node capabilities</td>
<td>Supported security levels (^1)[H/M/L]</td>
</tr>
<tr>
<td>Node security, privacy and trust requirements</td>
<td>Security mechanisms for each security level</td>
</tr>
<tr>
<td>Scenario</td>
<td>Profiles and rules</td>
</tr>
<tr>
<td>Context building block</td>
<td>Controlled data disclosure components parameters</td>
</tr>
<tr>
<td>Network/cluster type</td>
<td>Data cloaking operations</td>
</tr>
<tr>
<td>Scope of communications and data</td>
<td>Trust management components parameters</td>
</tr>
<tr>
<td>Scenario security requirements</td>
<td>Access list data for the scaled down version</td>
</tr>
</tbody>
</table>

of Figure 4.8 while the configuration parameters and the trust records included in the figure are stored in the profile and rule agent and the data store respectively. It makes the trust metrics available through the TRUST-QUERY interface defined in Section 4.4 and it utilises the TRUST-INDICATION interface for obtaining information from the profile and rule agent and the middleware subsystem for communicating trust evidence. The prerequisite that was set for the existence of authentication, confidentiality and integrity protection mechanisms for the trust evidence messages is therefore satisfied by enforcing the application of security mechanisms in the underlying connectivity subsystem. For nodes that are constrained but require the provision of trust management services, the hybrid trust management agent can be configured to be lightweight using the configuration parameters as discussed in Section 4.3.3.

6.4 Security Framework Configuration

The configuration of the security management framework and its components is made according to the security requirements and the capabilities of the deployments and the nodes. Table 6.2 summarises the configuration issues that should be addressed and the parameters that should be taken into account. From these issues, the ones that are not discussed in this section are the application-dependant configuration of the profiles and rules, and the configuration of the trust management, the controlled data disclosure, and the data cloaking components, since they have been discussed in Sections 4.3.3 and 5.2.

While most of the issues included in the table are decided upon for each node, the issue of the security mechanisms that are used for each SL is resolved at network level. The mechanisms included in Table 6.1 are indicative: for each deployment, the mechanisms should be selected according to the security requirements, the criticality of the scenario, and the purpose of the network. For example, unlike the working WSN scenario, WSNs deployed for environment monitoring may have weak or no data confidentiality requirements, so the encryption mechanisms can be weaker for all SLs.

The security manager version is resolved at node level. The extended version of the framework applies to nodes that are dynamic and may need to provide information
to requestors other than those that they were initially configured to disclose data to, nodes whose communications scope extends beyond the WSN, being gateways or sink nodes, or nodes that can provide interface to the end users for reconfigurations. The configuration of the supported SLs at each node is made according to the security requirements of the node’s communications, by taking into account the scenario, the type of network the node belongs to, the scope of its communications, and the criticality of the data produced by the node. Sensor nodes that have static pre-defined network roles and finite known sets of communicating parties may need to support only a subset of the available SLs.

In the working WSN scenario, if the mobile terminals that act as the gateways of the BSN clusters are known before deployment, then the sensor nodes of the BSNs could have the scaled-down version of the framework, with the mobile terminals being the only entities represented by entries in the access lists of the sensor nodes. The extended version of the security framework could reside only on the mobile terminals, so that they are responsible for defining the SL to be applied for the communications within the BSN on one side (low or medium SL), and with the hospital’s ESNs or external networks on other side. The high SL could be supported only by the mobile terminals, since these are the only nodes that accumulate and communicate aggregated, highly sensitive information. At the same time, only the terminals will perform data disclosure control operations that may require data cloaking or trust establishment with unknown external nodes.

### 6.5 Analysis and Evaluation

This section evaluates the security management framework against the requirements that were presented in Section 6.1.1 and examines the benefits received by the adaptability property of the framework. We do not discuss the trust management or controlled data disclosure components, since these have been analysed in Sections 4.7 and 5.5 respectively.

The security management solution addresses the requirement for application to WSN deployments since it (1) operates as a local security component on each node, (2) enforces pre-defined settings and policies through the profiles, rules, and configuration parameters, (3) allows its components to be simplified or even left out, and (4) selectively applies security mechanisms that have been evaluated and have been found to be applicable for sensor nodes. It provides a complete security management solution by combining the functionality of the security, controlled data disclosure, and trust management components. It supports the integration of WSNs with other networks by being uniformly applied to the sensor and to the gateway nodes and by enabling different versions of it to be deployed in the gateways to provide stronger security functionality.

The flexibility requirement is fulfilled both for the framework and for its components. The framework enables the deployment of extended and scaled-down versions, as depicted in Figure 6.1 while the components are customisable and allow for lighter versions to be deployed. The interdependencies between the components for the different versions were identified in Section 6.3 while guidelines for the application of the framework were provided in Section 6.4. The adaptability requirement is fulfilled
through the security agent component, that adjusts the security mechanisms that are applied as explained in Section 6.3.1.

The adaptability property of the framework essentially allows minimising the processing overheads through the selection of the optimal mechanisms and security primitives for each communication. However, the benefits gained from this property depend on the frequency of the SL changes, which require the exchange of configuration messages and increase the communication overheads. The effects of the adaptability property of the framework on battery power consumption were approximated through a prototype implementation on a testbed with Mica2 nodes. The aim of the performance evaluation was to quantify the difference in energy consumption between having and not having the ability to adapt the SL. A simplified version of the framework was deployed, with the security mechanisms implemented using TinySec, and the low, medium and high SLs represented by the TinySec transmit modes security disabled, authentication only, and encryption and authentication respectively.

Figure 6.2 shows the variation in power consumption according to the SL change rate. The curves labelled 100%H, 100%M and 100%L show the power consumption per node with the SL fixed at high, medium and low respectively. The points labelled 50%L 40%M 10%H represent the experimental values of the power consumption per node with the adaptive security framework switched on and with the respective proportions of time spent in each SL. The chart shows that, in comparison to constantly using the medium SL, benefits can be achieved by adapting the security behaviour, if the SL changes less frequently than every minute.

2The prototype implementation of the security framework, the results of which are used for its analysis, was performed during the course of the e-SENSE project by Thales Research and Technology Ltd, UK. Details on the implementation of the framework and further results can be found in reference [D2306]. The results presented in this section were reported in reference [MAM+08].
The benefits of the adaptability property were also approximated for different scenarios. Figure 6.3 plots the power consumption per 1000 sensor data messages, without including the power consumed in SL changes. The numbers indicate the number of configuration messages that can be sent per 1000 sensor data messages before the power consumed using the adaptive security framework exceeds that without the framework. The curves labelled Medium Only and High Only show the power consumption for fixed SLs of medium and high respectively. For scenarios using only the medium and high SLs (the first three in the figure), the reduction in power consumption achieved by the adaptive framework is limited. Using the adaptive framework provides the greatest benefits for scenarios that use the low SL most of the time.

The results of the experimental evaluation demonstrate that the power savings from the adaptability property depend on the proportion of the total time spent in each SL during network operation and the frequency of SL changes. The adaptability property is mostly beneficial for scenarios that use the low SL for a large proportion of the time and that change SL infrequently. The benefits in terms of resources therefore depend on the dynamicity and the security requirements of the application and the scenario.

6.6 Summary

The security management framework presented in this chapter integrates the trust management solution and the controlled data disclosure scheme with core security components in order to provide a complete security solution for WSNs. It enables the customisation of WSNs to a diverse set of application spaces, supports the heterogeneity in the capabilities and the security requirements of the nodes, and allows the secure integration of WSNs with other networks by being flexible and adaptive. The flexibility property is attained through following a modular approach for the integration of various configurable components in a toolbox solution. The adaptability property is attained
through defining different security levels, which can be simultaneously supported by
each node, and enabling the adjustment of the security level for each communication
according to pre-defined profiles and rules.

The specification of the framework includes its structure and components, the de-
pendencies between them, the mechanisms for providing adaptability and flexibility in
the framework and in each component, and the framework versions with their corres-
ponding sets of mechanisms. Extended and scaled down versions of the framework were
defined to provide the most suitable levels of security, controlled data disclosure and
trust functionality for the different nodes.

Performance evaluation results approximated the benefits of the adaptability prop-
erty of the security framework in terms of resource consumption. It was found that,
while the adjustments of the security mechanisms that are applied can lead to reduced
processing overheads, communication overheads increase because of the exchange of
configuration messages. The property was found to be mostly beneficial for scenarios
that are relatively static and do not require frequent reconfigurations of the security
mechanisms that are applied.
Chapter 7
Conclusions and Future Work

In this dissertation we have introduced a generic trust management model for WSNs, applied it for controlling data disclosure operations on the sensor nodes, and integrated it in a complete security management solution. In this chapter, we summarise our findings and describe potential directions for future research.

7.1 Summary of Results

Our work examined the fields of trust, privacy and security in WSNs. The first step that we undertook was to specify and document the problem that we are aiming to tackle and the objectives and boundaries of our work. The problem statements in Chapters 4, 5 and 6 were derived by examining (1) the characteristics of WSNs in node, network and data level, (2) their trust, privacy, and security requirements, (3) the properties and characteristics of each associated security concept, (4) the approaches that exist for addressing trust and privacy requirements in other types of systems and networks, and (5) the related work on trust, privacy, and security management in WSNs.

The objectives that we set for our work were to provide:

1. A generic and secure solution for dynamically managing the trust relationships within and between heterogeneous WSN deployments according to the pre-deployment knowledge, the network purpose, and the available feedback on malicious behaviours.

2. A solution for performing localised data access and data granularity control at the points of data capture in distributed and dynamic WSN deployments according to the deployment needs, the network purpose, and the context of the data requests.

3. A localised, flexible and adaptive security management solution for heterogeneous and integrated WSN deployments, that facilitates the provision of core security functionality along with trust management and privacy protection services.

For each of these objectives, we formulated a set of detailed requirements for our solution, stated our assumptions and limitations, and identified the works that are mostly related to ours. We then specified our solution, justified our main design decisions, and analysed and evaluated it both in terms of how it addresses the requirements, and through experimental evaluation results. Throughout the dissertation, we
used three different WSN scenarios to assist in the explanation and analysis of the solutions: a generic WSN scenario to which pre-deployment knowledge was added in Chapter 4, a vehicular sensor network scenario in Chapter 5, and a body sensor network scenario in Chapter 6. The following sections summarise our contribution and main findings with respect to each of the objectives.

7.1.1 Trust Management

Trust management is an important security service for WSNs, mainly because of their cooperative nature and their vulnerability to node compromise and misbehaviour. The analysis of the characteristics and the security requirements of WSNs highlighted that the applicable trust management solutions should be lightweight in terms of computational and communication requirements, yet powerful in terms of flexibility in managing trust between nodes of heterogeneous deployments. At the same time, the solutions should utilise the pre-deployment knowledge on the roles of the sensor nodes and their trust associations.

In the discussion of the related work in the area, we categorised the trust management solutions into certificate-based, behaviour-based and hybrid models, according to their scope, purpose and admissible types of evidence. The certificate-based models were analysed according to the entities that provide certification authority functionality, while the behaviour-based according to their approach for modelling and evaluating trust. We also discussed the hybrid models, which combine different approaches on trust establishment to benefit from the properties of trust that each individual approach offers. The main conclusions that we have drawn from the related work analysis include:

- The certificate-based and the behaviour-based trust models adopt different perspectives of trust, each approach having its own limitations and drawbacks in terms of the characteristics and properties of trust that it supports, its resource requirements, and its applicability on WSNs. The two approaches are not alternative but supplementary, and a hybrid approach can leverage the drawbacks identified in the individual approaches.

- The drawbacks of the certificate-based trust models include their lack of feedback potential on the behaviour of the nodes and their resilience on asymmetric cryptographic operations, which increases their computational complexity and limits their applicability on resource constrained sensors.

- The drawbacks of the behaviour-based trust models include that they do not support role-based trust, they do not utilise the pre-deployment knowledge that may be available in WSN deployments, and they have high energy requirements for accumulating behavioural evidence through observations.

- The hybrid models in the related bibliography are all proposed for distributed systems or multiagent systems, and do not take into account the special requirements and limitations of WSNs, like their resource limitations and the varying connectivity, while they do not specify concrete metrics and processes for obtaining and combining the different types of evidence that they support.
Based on these findings, and having studied the trust management requirements of WSNs, we set the following detailed objectives for our solution: It should be generic, apply to WSNs, support the heterogeneity in the trust evaluation needs of the deployments and the nodes, enable the utilisation of the pre-deployment knowledge that may be available, allow trust revocation due to malicious behaviour, and be robust against attacks to the trust management model itself. The trust management model that we introduced is hybrid, utilising all role-base trust associations, certificates and behaviour evaluation results as trust evidence, and combining them on common evaluation processes and metrics. Its specification included its structure and interfaces and a number of novel aspects, including the different types of trust evidence and trust associations that it supports, a rich set of configuration parameters, the trust management processes and the trust evaluation metrics.

For the design of the solution we assumed the existence of authentication and integrity protection mechanisms for the exchanged messages. Moreover, the solution was designed to utilise behaviour evaluation results provided as an independent network service by supervision deployments, but it was out of the scope of our work to specify the behaviour evaluation mechanisms. Instead, we specified which existing behaviour-based solutions the model can be integrated with and receive their feedback. Another limitation of our work is that the model was not applied and evaluated on WSN test beds or specific WSN deployments. Instead, the model and the processes and metrics that is includes were validated through simulation experiments with varying parameters. Moreover, the analysis of the model included its evaluation against the requirements and the analysis of its security and properties, of its robustness against various types of attacks, and of its resource requirements and overheads.

The main conclusions that we have drawn from our work on trust management and from the analysis and evaluation of our solution are:

- The trust management model fulfils the requirements that were set and respects the trust transitivity properties, while its robustness against some attacks depends on the robustness of the behaviour-based trust evaluation mechanism implemented in the supervision network.

- The model can support the needs of deployments and nodes with diverse trust management requirements. Simulation results demonstrated that it can support from unstructured dynamic networks to clustered networks for which rich pre-deployment knowledge exists, and from highly adaptive nodes to static and restricted nodes, that will never during the network lifecycle perform certificate validations or combinations of recommendations.

- The resource requirements of the model depend on the type and configuration of each node, with larger overheads being imposed to highly adaptive or critical nodes that are pre-configured to support certificates as trust evidence or require a large amount of evidence for establishing new trust relationships.

- The model provides the means to control the type of operations executed by the nodes, and to distribute the required operations according to their trust evaluation requirements and their capabilities.
• WSNs can benefit from behaviour evaluation with minimal cost in resources if it is performed by supervision networks that make their behaviour evaluation results available, and if the trust management solution applied in the sensor nodes enables the utilisation of these results.

• Trust management models can exploit the pre-deployment knowledge on the network purpose, topology and information flows for controlling not only trust evolution, but also the distribution of the trust management overheads among the network nodes.

The trust management model is generic in terms of the semantics of trust, since the semantics of the trust evidence that it utilises can be defined according to purpose the model is applied for. This enables its use not only for controlled data disclosure, but also for other security-related services that are included in Section 7.2 as future research directions.

7.1.2 Privacy Protection and Controlled Data Disclosure

WSNs collect and communicate information of varying granularity and sensitivity with respect to the privacy of the entities that they directly or indirectly monitor. Analysing the privacy requirements of WSNs, we identified three main issues that need to be addressed: the protection of the communications context, privacy sensitive information disclosure, and privacy sensitive information gathering. The same categorisation was made for the discussion of the related work in the area. Some works aim to protect against adversaries overhearing the communications, while others aim to protect against illegitimate access of information from service providers. Different assumptions are made about the entities that are considered trusted, with some schemes trusting only the WSN deployment and other schemes assuming that there exist trusted intermediaries to enforce privacy mechanisms. Some schemes address user privacy concerns related to the information captured and transmitted by his carried devices, while others address concerns with respect to environmental sensor networks that capture information about people in their proximity. Some schemes interfere with the routing protocol, by requiring modifications of the message headers or introducing routing path selection strategies, others interfere with the information flow through introducing intermediaries, while others interfere with the data through or enforcing adjustments to its granularity.

The main conclusions that we have drawn from the related work analysis include:

• Privacy issues in WSNs are addressed at multiple levels of the network stack and at different points of the sensed information flow to meet most of the privacy requirements, by solutions that have different aims and are complementary.

• Most of the solutions that have been proposed for protecting the context of the communications and for privacy sensitive information gathering influence the system design at the routing protocol and information flow levels, which can complicate their actual integration to the deployments.

• Privacy comes at a cost in resources and a number of techniques, like phantom routing, dummy traffic, and message buffering, have increased resource requirements and could impose negative effects in the operation of the network.
7.1. SUMMARY OF RESULTS

A number of the solutions that have been proposed for controlled data disclosure and data granularity control were found to have limited applicability, being either too application-specific, or requiring the existence of trusted intermediaries or middleware services, thus assuming that all devices up to the level where the privacy operations are performed are trusted.

Based on these findings, especially for the controlled data disclosure issues, and having studied the privacy requirements of WSNs, we set the detailed objectives for our solution: It should apply to distributed and dynamic WSNs at the points of data capture, enforce both data access and data granularity control, respect the deployment needs and the network purpose, and enable taking into account the context of the data requests. The controlled data disclosure solution that we introduced positions trust as the facilitator of the data access and granularity control decisions of sensor nodes, utilising the results that the hybrid trust management model produces. The specification of the solution included its structure and interfaces, novel data disclosure control operations for classifying the data requestors and determining the data cloaking operations to be performed, and alternative options for cloaking the data before publishing it to partially trusted requestors, which include negative surveys and data granularity control mechanisms.

For the design of the solution we assumed the existence of mechanisms for the reduction of data accuracy and for emergency situation identification, and we identified alternative options for their implementation. A limitation of our work is that the solution was not applied and evaluated on WSN test beds or specific WSN deployments. Instead, it was validated through simulation experiments on a scenario with both trusted and partially trusted nodes.

The main conclusions that we have drawn from our work on controlled data disclosure and from the evaluation of our solution are:

- The hybrid trust management model provides adequate information for identifying how data requests from unknown requestors should be handled at each sensor node, while respecting the deployment needs and the network purpose.

- Both approaches that the solution supports for data cloaking, namely the reduction of data accuracy and negative surveys, have limitations in terms of the types of data they support and in addressing service information needs, thus a generic and application-independent solution could not be provided.

- Data cloaking mechanisms enable the provision of services that do not require accurate or fully identifiable data but, in practice, it is unlikely that WSNs will be deployed for providing only cloaked data during their lifetime. A mechanism for deciding when they should be applied is therefore required.

- Trust management is valuable the field of controlled data disclosure, especially for the flexibility that it offers in handling unknown or compromised data requestors.

The proposed solution does not include specifications for its application-dependant components, and does not aim to cover the complete spectrum of WSNs’ privacy requirements. Possible future work towards these directions is included in Section 7.2.
7.1.3 Security Management

The provision of core security services for authentication, confidentiality and integrity protection is a prerequisite for secure WSNs. Other services that are especially important for WSNs include the detection of compromised nodes and secure data aggregation. Apart from the security services, the requirements analysis highlighted the need for flexible and adaptable security management to address the need for self-organisation and to support the integration of WSNs with other networks and platforms and the heterogeneity between the nodes and the deployments in the capabilities and the security needs.

In the discussion of the related work in the area, we included the key exchange and distribution solutions that have been designed to serve as the basis for other security services such as authentication and encryption of sensor node communications, the secure data aggregation and verification mechanisms that have been proposed to deal with compromised nodes within the network, and the findings of the studies that have been made on the applicability of security primitives and protocols on the sensor nodes. Moreover, we discussed the solutions that have been proposed for security services integration and management.

The main conclusion that we have drawn from the related work analysis is that, while there exist solutions to address most of the core security requirements of WSNs, the issue of security management has not been sufficiently addressed. Few works were found on the integration and management of security services, none of which was targeted to WSNs and offering flexibility, adaptability and context-awareness properties and including all security, privacy and trust services.

Based on these findings, especially for the security management issue, and having studied the security requirements of WSNs, we set the detailed objectives for our solution: It should be a complete and localised security management solution, apply to WSN deployments, offer flexibility and adaptability properties, and support the integration of WSNs with other networks. The security management framework that was presented integrates the trust management model and the controlled data disclosure scheme along with core security services to guarantee the authenticity, integrity, and confidentiality of the data generated and communicated through the network. Its specification included its structure and components, the dependencies between them, the mechanisms for providing adaptability and flexibility in the framework and in each component, the framework versions with their corresponding sets of security mechanisms, and guidelines for matching the security requirements of the nodes and the deployments to the security framework configuration.

For the design of the solution we assumed the existence of context awareness mechanisms which, being application-dependant, could not be specified, and we limited the scope of the solution to the WSN security issues that can be addressed by a security management service without interfering with the routing protocol or with the network design. Another limitation of our work is that the framework was not applied and evaluated on specific WSN deployments. Instead, performance evaluation results derived through a prototype implementation of the framework approximated the benefits of the adaptability property in terms of resource consumption.

The main conclusions that we have drawn from our work on security management are:
7.2. FUTURE WORK

• The integration of the trust management model and the controlled data disclosure scheme in the security framework enabled fulfilling the assumptions that were made about the existence of core security functionality.

• Security management in WSNs needs to be flexible, in order to enable the customisation of WSNs to a diverse set of application spaces and to support the heterogeneity in the capabilities and the security requirements of the nodes. This property can be attained by following a modular approach for the integration of various configurable security components in a toolbox solution, and specifying their dependencies. This approach enabled the definition of extended and scaled down versions of the framework to provide the most suitable levels of security, controlled data disclosure and trust functionality for the different nodes.

• Security management in WSNs can be adaptive, by enabling pre-defined profiles and rules to adjust the security mechanisms that are applied for each communication. The adaptability property can lead to reduced processing overheads. At the same time, however, communication overheads increase because of the exchange of configuration or synchronisation messages. The property was found to be mostly beneficial for scenarios that are relatively static and do not require frequent reconfigurations of the security mechanisms that are applied.

• In resource constrained environments like WSNs, the selection of lightweight security mechanisms is only one aspect of the solution for reducing the security overheads. The other aspect is controlling how and when the security mechanisms are applied, so that (1) only the minimum required security functionality is provided and (2) the heterogeneity of the nodes in terms of their capabilities is exploited for distributing the security overheads.

The last observation applies both to the security management solution and to the trust management model, and this is the reason why both were designed to be highly configurable and to support controlling the restriction or the delegation of operations. The benefits of this were analysed and approximated, but not quantified for specific WSN deployments, which is a possible direction for future work utilising WSNs deployed for specific applications.

7.2 Future Work

The possible directions for future research work on trust, controlled data disclosure, and security management are in three dimensions: towards the realisation of the assumptions that we have made about application-dependant issues by applying the solutions that we proposed in specific WSN deployments in various application spaces, towards addressing open research issues in areas that are related to our work, and towards the application of the concepts and the solutions that we proposed in other environments or for addressing other security requirements.

The trust management model that we proposed is generic and allows for the semantics of trust to be defined according to purpose it is applied for. This enables utilisation not only for controlled data disclosure but also for other security-related services. Trusted key establishment and management is one of the most interesting
applications that we believe the model is suited for. The key management schemes
that have been proposed for WSNs, although they are cooperative and vulnerable to
node compromise, do not explicitly utilise trust concepts. For example, the key pre-
distribution solutions that were discussed in Section 3.3.1.1 implicitly assume that,
onece valid common keys are existing in the nodes’ key pools, then the nodes are mu-
tually trusted and can establish a symmetric key. A possible future research direction
is the application of our trust management model for trusted key exchange and man-
agement, in order to benefit from the flexibility that it offers in handling unknown and
compromised nodes.

The hybrid trust management concepts can be applied to other environments except
from WSNs. The integration of various types of trust evidence, including these derived
from certificates, policies, reputation, or direct observations, in a unified evaluation
model has a number of advantages, discussed in Section 3.1.5. In the model that we
proposed for WSNs we have restricted the types and the amount of admissible evidence,
for the trust management processes to be lightweight and match the characteristics of
the sensor nodes. In order to receive the full benefits of the integration of different types
of trust in common processes and metrics, the same concepts can be applied to other
types of cooperative networks and environments where the resource and connectivity
restrictions do not apply, pre-deployment knowledge is available, and nodes are similarly
vulnerable to compromise or misbehaviours.

The solution that we proposed for controlled data disclosure does not define the
application and data-dependant mechanisms that can be applied for reducing data
accuracy. Both approaches that it supports for data cloaking, namely negative surveys
and reduction of data accuracy, have limitations in the types of data they support and in
addressing service information needs. The first is appropriate only for the provision of
services for which the distribution of data is important rather than specific answers from
sensor nodes, while the latter can only be performed naively by the sensors, without
taking into account the size of the anonymity set. As a result, data cloaking is an aspect
of the scheme where an application-independent solution could not be provided, and a
possible future research direction is towards devising a generic, probably cooperative,
method to handle partially trusted data requestors.

Moreover, the controlled data disclosure solution does not aim to cover the com-
plete spectrum of WSNs’ privacy requirements, which have to be addressed at lower
levels as well. The design of a complete, cross-layer privacy protection solution is a
challenging possible future research direction. It would have to integrate the controlled
data disclosure mechanism with identity and pseudonym management and with loca-
tion privacy mechanisms, and it would need to be applied both to the sensor nodes and
to the aggregators and the intermediaries that may exist in the information flow.

In the field of security management, the usability of the proposed solution is another
possible direction for future research. The framework includes an application support
component, to be implemented only in user-related full-function nodes, but does not
specify how this component is implemented. This component should be designed to
provide an interface to the users both for the configuration of the trust and controlled
data disclosure parameters, and for receiving feedback or intervening with the trust
and data disclosure decisions. This research direction would also entail examining how
the services that are provided to the end users are affected by the trust establishment
and the trust revocation operations in WSN deployments of various application spaces.
Moreover, the security management solution that was proposed is highly configurable, and the benefits of its adaptability property were approximated through a prototype implementation. However, although its analysis and evaluation demonstrated that it can be effectively applied to WSNs of various application spaces, it remains a theoretical solution. As a possible direction for future work, it would be interesting to apply and evaluate it on specific WSN deployments. The deployments that we identify as the most suitable candidates are those designed to support healthcare applications in the medical field, since they combine different types of nodes and clusters, while having increased security and privacy requirements.

7.3 Concluding Remarks

The focus of this dissertation was trust management and its application for secure and privacy preserving WSNs. We introduced a generic trust management model for WSNs, and validated it in terms of the supported trust characteristics, its robustness, its resource requirements, and through experiments, that demonstrated that it can uniformly support nodes with highly diverse roles and capabilities while distributing the cost of trust evaluation operations. We applied the model for controlling data disclosure operations on the sensor nodes, and integrated it in a complete security management solution. The related work analysis that preceded showed that the notions of trust and reputation have received various interpretations. Most trust management models that have been proposed for WSNs implicitly assign strict semantics to trust, which limits their applicability for security services other than secure routing. The hybrid nature of the model we proposed was inspired by the observation that WSN deployments are purpose-specific, and trust has to respect the network purpose and not be solely built on behavioural evidence. At the same time, this characteristic was what made its metrics generic, allowed its application for controlled data disclosure, and opened new research directions towards applying it for other services and in other environments.
Bibliography


