# Handbook of Research on Design, Control, and Modeling of Swarm Robotics

Ying Tan
*Peking University, China*

A volume in the Advances in Computational Intelligence and Robotics (ACIR) Book Series

Information Science REFERENCE
An Imprint of IGI Global

Published in the United States of America by
Information Science Publishing (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA, USA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com



Library of Congress Cataloging-in-Publication Data

Handbook of research on design, control, and modeling of swarm robotics / Ying Tan, editor.
pages cm
Includes bibliographical references and index.
ISBN 978-1-4666-9572-6 (hardcover) -- ISBN 978-1-4666-9573-3 (ebook) 1. Robotics--Research. 2. Swarm intelligence--Research. 3. Robots--Control--Research. I. Tan, Ying., editor.
TJ211. H2645 2016
629.8'9263824--dc23

2015032768

This book is published in the IGI Global book series Advances in Computational Intelligence and Robotics (ACIR) (ISSN: 2327-0411; eISSN: 2327-042X)

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

# Chapter 2
# Security in Swarm Robotics

**Thalia May Laing**
*Royal Holloway, University of London, UK*

**Allan Tomlinson**
*Royal Holloway, University of London, UK*

**Siaw-Lynn Ng**
*Royal Holloway, University of London, UK*

**Keith M Martin**
*Royal Holloway, University of London, UK*

## ABSTRACT

*Inspired by social animals, such as ants, bees and fish, which appear to exhibit what has been dubbed 'swarm intelligence', swarm robotic systems aim to accomplish goals that are unachievable by an individual robot. Swarm robotics have a large number of potential uses, including applications in the military, monitoring, disaster relief, healthcare and commercial applications. To be able to achieve their goals, it is of utmost importance that communications between agents are secure in the presence of possibly malicious interruptions and attacks from adversaries. The authors will discuss the issues surrounding the provision of secure communications in swarm robotics: what secure communications mean, how the characteristics of swarm robotics present a security challenge, the relationship between security issues for swarm robotics and other network technologies, and how different adversarial models demand different types of solutions. It will then be discussed what the important open research questions are in secure communications in swarm robotics.*

## INTRODUCTION

Swarm robotics is concerned with the coordination of large numbers of relatively simple robots. Although there is no universally accepted definition for swarm robotic systems, Şahin (2005) proposes the following working definition: *swarm robotics is the study of how a large number of relatively simple physically embodied agents can be designed such that a desired collective behaviour emerges from the local interactions among agents and between the agents and their environment*.

Swarm robotic systems aim to accomplish goals that are unachievable by an individual robot. In a number of situations, having numerous simple robots forming a swarm, rather than an individual complex robot, could be beneficial as it may be cost effective or achieve the set goal more effectively. Because of this, swarm robotics have a large number of potential uses, including applications in the military, medical scenarios, disaster relief, monitoring and commercial applications (Şahin, 2005).

DOI: 10.4018/978-1-4666-9572-6.ch002

Şahin (2005) explicitly puts forwards five characteristics as criteria for distinguishing swarm robotics from other multi-robot research in Section 3 of his paper. Şahin suggests these characteristics as a way to differentiate swarm robotics from other multi-robot systems. His five defining characteristics of swarm robotics are:

1. **Autonomous Robots**. Robots are able to act without the direct intervention of humans and have control over their own actions and internal state.
2. **Large number of Robots**. Closely linked to the idea of scalability, there should be a large number of robots, or studies should be applicable to the control of large robotic swarms.
3. **Consist of a Few Homogeneous Groups of Robots**. The swarm network should consist of relatively few groups of homogeneous robots.
4. **Relatively Incapable or Inefficient Robots**. On an individual level, the robots should be relatively simple and either incapable of completing tasks individually, requiring cooperation amongst the swarm to achieve the global goal, or working as a group should improve the performance and robustness of the handling of the task.
5. **Local Sensing and Communication Capabilities**. The robots should have local and limited sensing and communication abilities to ensure distributed coordination amongst the swarm. A global communication channel may be used to download a common program onto the swarm, however this should not be used for coordination amongst the robots (as this is likely to be unscalable) and the communication is considered to be one way, in the direction from the channel to the swarm.

Other characteristics not explicitly listed by Şahin (although some are implicit) include:

6. **Co-Operate to Accomplish Tasks:** As the robots are relatively incapable or inefficient, they are required to, or would benefit from, cooperating to complete any given tasks.
7. **Mobile:** The robots are mobile. It is generally assumed their movement is not predictable. Sometimes, however, the robots may be bounded to movement within a predetermined boundary.
8. **Self-Organising:** Swarms should be self-organising, defined by Camazine, Deneubourg, Franks, Sneyd and Theraulaz (2002), as *the process in which pattern at the global level of a system emerges solely from numerous interactions among the lower-level components of the system*.
9. **Collective Emergent Behaviour:** A collective behaviour emerges from the local interactions among agents and between the agents and their environment.
10. **Decentralised Control:** The individual robots must operate on local information obtained to accomplish global goals. There is no central point of control in the system and coordination is completely distributed. This characteristic contributes to achieving robustness, as there is no common node failure point or vulnerability (Winfield & Nembrini, 2006).
11. **No Individual Identity:** In a swarm, there are relatively few groups of homogeneous robots. In each homogeneous group, the robots can be identical, as they do not need to be individually identified. Thus individual identification is not necessarily required.
12. **Lack of Synchronicity:** As described by Beni (2005), the units of the swarm do not move synchronously or sequentially, but interact dynamically.
13. **Range of Communication:** Swarms use both explicit and implicit communication methods. Explicit communications are where one robot communicates directly with another. Such methods of communication include

radio frequency and infra-red technologies, which have previously been well studied. Implicit communication includes interaction via sensing of other robots, their behaviour and their interaction with the environment. Communication such as this is known as *stigmergy*. An example of stigmergy in nature is that of foraging ants that lay pheromones to communicate and create foraging patterns. These are believed to increase their foraging performance (Şahin, 2005, Section 3.4).

These characteristics contribute to properties of swarm robotics that are desirable in many applications, especially in unpredictable or hostile environments:

1. **Resilience:** The loss of individual agents has little impact on the success of the task. Many agents collect the same data so the functioning of individual agents is not critical.
2. **Decentralised control:** The coordination of the swarm is distributed. Again, this ensures there are no single points of failure.
3. **Adaptability:** The swarm will adapt itself dynamically to the environment, allowing changes in topology.

## Chapter Outline

Much research on swarm robotics assumes a benign, or at least a non-malicious, environment where failure of agents thus arises only due to malfunction or accidents. However, many proposed deployments of swarm robotics are in hostile or unpredictable environments - there may be adversaries who actively attempt to sabotage the swarm, or who attempt to eavesdrop on important and confidential information collected by the swarm. As demonstrated by Berghmans and Deckers (2013) in a simulation, adversaries were able to successfully slow down or freeze collective exploration with a number of attacks. It is important that attacks such as these are protected against.

In the next section, we will provide an overview of basic security services identified in the ISO standard for the security architecture of the OSI reference model (ISO, 1989), and consider what security services cryptography can provide.

In the section 'Secure Communications in Swarms', we will focus on secure communications in swarm robotics by first considering some examples of use. We will then examine how some of the characteristics of a swarm present a challenge to its security. Swarm robotics uses a range of communication methods: direct and indirect communication through the environment (in particular, stigmergy and local sensing). These communication methods and their impact on swarm communications security will also be discussed.

From a security perspective, swarm robotics can be regarded as a special type of computer network with additional characteristics. As such, swarm robotics share many characteristics with a number of other technologies that have already been subjected to extensive security analysis, such as mobile sensor networks (MSNs) (Walters, Liang, Shi & Chaudhary, 2007), mobile ad-hoc networks (MANETs) (Akbani, Korkmaz & Raju, 2012), vehicular ad-hoc networks (VANETs) (Raya & Hubaux, 2007) and other multi-robot systems and software agents (Wooldridge, 2009). In the section 'Related Technologies', we will briefly describe some of the main security challenges for these technologies and examine how these compare with the potential security challenges faced in swarm robotics. Following this, the issue of how to model stigmergy in order to secure this indirect form of communication is discussed.

In most security research, adversaries and attackers are classified according to both the information they have access to and their capabilities. We will discuss both internal and external adversaries in detail and consider what threats they pose to the security of swarm robotics. After this, some challenges unique to the swarm will be highlighted and discussed.

As little research into security in swarm robotics has been conducted thus far, there are a number of open questions. A discussion of a selection of these will conclude the chapter.

## BACKGROUND

In this section, we give a short overview of the general issues surrounding secure communications. Many authors have provided an overview of the subject, such as Akbani et al. (2012) and Wooldridge (2009). We refer the reader to them for more background and details.

### Introduction to Secure Communications

A communications system provides a channel for transmission of information from point A to B. Swarm robotics uses wireless transmission links, as well as local sensing (such as infra-red) and stigmergy. Messages sent over these channels need to be encoded for transmission. This is summarised in Figure 1.

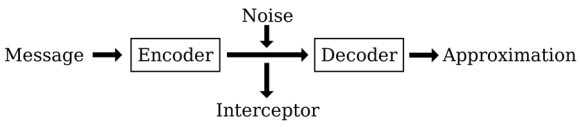
With the possibility of noise or interference in the transmission channel, error-correcting codes are deployed to determine whether the message was received correctly, and whether it can be fully recovered. The theory of error-correcting codes deals with these issues (Trappe, Washington, Anshel & Boklan, 2007).

It is also generally assumed that the communication channel is insecure - that is, an adversary may intercept, read and modify messages sent through this channel. There are several different aspects to "security" that need to be considered, the most important of which are commonly identified as (ISO, 1989):

1. **Confidentiality:** The assurance that no one other than the intended recipient(s) can read the data; in other words, the data is kept secret between the sender and recipient.
2. **Integrity:** The assurance that the data has not been altered, either maliciously or accidentally, in an unauthorised way.
3. **Entity Authentication:** Sometimes called *identification*, entity authentication serves to identify specific entities in isolation from any other activity the entity may want to perform.
4. **Data Origin Authentication:** This service identifies a specific entity as the source or origin of a given piece of data.
5. **Availability:** Availability ensures that accessibility and usability are available upon demand by an authorised entity. The loss of availability is commonly referred to as *denial-of-service*.

In a swarm, it may be desirable to apply any number of these services to the data being exchanged between individual robots. Some security services may be more important than others and the services required are dependent on the application of the swarm. Before we consider the provision of security services specific to swarm communications, we will briefly discuss how cryptography can be used to deal with some of these issues.

*Figure 1. Communications system*




### Cryptography

*Cryptography* is the art (and science) of designing systems to transform messages in such a way that two entities can communicate securely over an insecure channel.

The information to be communicated is commonly referred to as the *plaintext*. In a basic communications scenario, the sender encrypts the plaintext using a predetermined *encryption key*. The result is known as the *ciphertext* and it is the ciphertext that is sent over the insecure channel. The receiver, who knows the *decryption key*, is able to decrypt the ciphertext to obtain the plaintext. A basic *cryptosystem* is summarised in Figure 2.

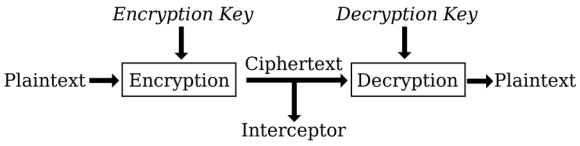
In traditional (*symmetric-key*) cryptosystems the communicating parties first need to agree on a randomly chosen key, which is used for both encryption and decryption and must be kept secret. If the key is compromised then so is the entire system. Being able to efficiently agree upon and exchange a key securely is known as the *key distribution* problem. An overview of this can be found in Menezes, Van Oorschot and Vanstone (1996).

However, there also exist cryptosystems where the encryption key and the decryption key are not identical. These *public-key* (or *asymmetric*) cryptosystems have the property that the encryption key can be made public, and only the decryption key needs to be kept secret. These cryptosystems have the advantage that encryption keys can be obtained from public directories, thus avoiding the need to preshare secrets in advance of communication. In general, public key cryptosystems are slower than symmetric-key cryptosystems. However they can be used to establish secret keys for a symmetric-key cryptosystem, which can then be used to encrypt the data. Again, an overview and details can be found in Menezes et al. (1996).

All ciphertexts are sent over the communication channel and are hence observable. It is generally also assumed that the encryption process (the algorithm used) is known by any adversary and hence security of a cryptosystem relies entirely on the security of the decryption key.

Note that Figure 2 describes the basic model for using a cryptosystem to provide confidentiality. Similar models exist for the use of cryptography to provide integrity and authentication. It should be noted that cryptography cannot readily be used to support availability.

*Figure 2. A cryptosystem*




## SECURE COMMUNICATIONS IN SWARMS

Before we consider the security of communications in swarm robotics we consider some scenarios where swarm robotics may be used and examine the security issues that might arise in these scenarios. An example of how important it is to secure communications in a robotics swarm is illustrated by Berghmans and Decker (2013), where they simulate and analyse six different attacks on collective exploration. They conclude that the collective exploration algorithm used was vulnerable to threats from enemies and that, before swarms are used in real world applications, security measures should be taken into account.

### Applications and Behaviours of Swarm Robotics

Swarm robotics have applications in many areas, for example, military (mine clearance and surveillance), environmental monitoring, disaster relief and health care (medication provision, monitoring, cleaning, guidance of patients and intruder or emergency detection).

Consider the medication provision aspect of a healthcare application investigated by a European Union 6$^{th}$ framework programme project, Intelligent Robot Swarm for Attendance, Recognition,

Cleaning and Delivery (iWARD) (2007). This swarm consists of robots with a secure physical storage box and an input/output screen. A health practitioner inputs the medication required, the robot finds its way to the pharmacy, the pharmacist puts the requested medicine in the secure storage box, then the robot takes the medicine back to the healthcare practitioner. Clearly there are many potential security concerns: the availability of a robot from the swarm, the authentication of the healthcare practitioner and the robot, the confidentiality and integrity of the information being transmitted/transferred, the correct routing of the robot, the authorisation and authentication of the pharmacist and the robot, and the security of the storage box.

*Figure 3. An iWARD robot*
*Source: Thiel, Habe and Block, 2009*

The Guardians Project (2007) considered the example of a fire in an industrial warehouse. Robots accompany a human squad leader. The robots are connected wirelessly and are self-organising. They explore ahead of the human leader and send data to the leader and to central control. In this case the important security services are the availability of the channels between all parties and the authenticity of the information.

More generally, the application of swarms can be reduced to a number of behaviours. In (Navarro & Matía, 2012) a list of basic behaviours and tasks typical of swarm robotics is formulated. The tasks include pattern formation, collective movement, task allocation, source search, collective transport of objects and collective mapping. Generally, the behaviours key to performing these tasks are aggregation and dispersion (an example of a behaviour that is not is object clustering (Brambilla, Ferrante, Birattari, & Dorigo, 2013)).

## Identification of Security Services Needed

In both the iWARD (2007) and the Guardians project (2007), confidential and authenticated communication between agents of the swarm is a prerequisite to the functioning of the swarm. These are well-studied security requirements in other networking environments.

In both aggregation and dispersion behaviours, agents must be able to sense the proximity of another agent from their swarm, either to then

*Figure 4. The team of robots used in the Guardians Project*
*Source: Saez-Pons, Alboul, Penders and Nomdedeu, 2010*

draw closer or disperse. This requires an agent to be able to somehow authenticate another agent of the same swarm.

## Challenges to Security Posed by the Swarm

Higgins, Tomlinson and Martin (2009a) list a number of challenges to providing security in a swarm robotic environment. These were then further considered by Berghman and Deckers (2013). Most of these challenges are common to other technologies, while some appear to be unique to swarm robotics and require further consideration.

### Challenges Found in Other Technologies

1. **Resource Constraints:** A small device, such as an agent in a swarm, may feature a number of different resource constraints, such as storage, communication bandwidth, computational restrictions and energy. If any of these resources were to become unavailable, the robot might become inaccessible.
2. **Scalability:** Within swarms, there is the potential to have a large number of robots. As a result, any security solutions posed should be suitable for a large number of robots.
3. **Physical Capture and Tampering:** A robot could be captured, leading to the loss of availability of the seized robot and the data held. Furthermore, an adversary might obtain all data on the commandeered robot and be able to tamper with the robot before reintroducing it back into the swarm.
4. **Control:** Due to the distributed nature of swarms, there are no points of control. Any security solutions must bear in mind that swarms should consist of relatively few groups of homogeneous robots in which the individuals make decisions and operate based on local information obtained to accomplish global goals. This presents the risk that the system becomes out of control. As Şahin (2005, Section 3.5) describes, a global communication channel may be used to download a common program onto the swarm, but this channel must not be used for coordination amongst the swarm. This global communication channel may be the only way to communicate with the swarm post deployment, but this communication is only in one direction.
5. **Real Time:** Ideally, the robots must act and communicate in real time. Because the robots' actions are dependent on each other and their environment, if communicating and validating messages is not immediate then the environment around them may have changed before they are able to act. Therefore solutions should be efficient to enable robots to react in real time.
6. **Swarm Mobility:** Security can be hard to achieve in mobile environments such as swarm robotics. Additionally, constraining the movement of swarm members to, for example, stay within set boundaries, may pose further complications.
7. **Key Management:** Cryptographic keys form a core component of any security mechanism suitable for swarms (Dolev, Lahiani, & Yung, 2007), and the dynamic nature in which robots join and leave the swarm could present problems for key management.
8. **Intrusion Detection:** If a foreign entity joins the swarm, or if a robot is reprogrammed and released back into the swarm, the swarm will need to be able to detect an intrusion. Intrusion detection systems work by targeting anomalous behaviour. However, because robots are autonomous entities, apparently anomalous behaviour may in fact represent suitable reactions to their environment and could diffuse through the swarm and eventually become desirable, collective behaviour

that would enable the swarm to efficiently achieve their goal. Trying to identify and stem anomalous behaviour may result in making the swarm inefficient. However, there is a chance that anomalous behaviour is initiated by an intruder and is malicious, hence the correct identification and appropriate response is vital. Telling the difference between malicious anomalous behaviour and desirable, pioneering anomalous behaviour is a difficult problem. These issues will be discussed further when addressing challenges unique to the swarm.

9. **Managing Learning:** Robots have the ability to learn and react to environmental changes. An adversary may be able to take advantage of this and change the pattern of behaviour in the swarm in order to achieve its goal.

## Challenges Unique to the Swarm

1. **Identity and Secrets:** In order to provide a number of security services, the identification of robots is vital; as noted by Higgins et al. (2009a, Section 3.2). Robots could either be identified as individuals, with each individual robot having a unique identity, or as part of a swarm, where the distinction between robots in a swarm is unnecessary and ensuring that the robot belongs to the swarm is sufficient. However, there are settings where individual identity is undesirable (Flocchini, Prencipe, Santoro & Widmayer, 2005) which may make implementing some security solutions difficult.
2. **Stigmergy:** Stigmergy refers to the communication of agents via the environment: robots modify their environment, which is noted by other robots and affects their behaviour. As an example, ants lay down pheromones on the way home to their nest and other ants follow this trail. When applying stigmergy to swarms, agents are required to both 'read' and 'write' to their environment. Just like with direct communication, when communicating using stigmergy, we need to ensure that
   a. Only agents in the vicinity can communicate;
   b. Messages left in the environment can only be read by legitimate agents;
   c. Messages read in the environment are indeed from legitimate agents;
   d. Messages are only valid for a certain time period;
   e. Messages left in the environment have not been altered, either maliciously or accidentally.
3. **Local Sensing:** Local sensing refers to how an agent's behaviour is affected not by direct communication with other agents, but by what the agent can observe in its immediate surroundings, such as the configuration or behaviour (velocity of movement, for example) of its neighbours. Whereas stigmergy requires one swarm member to actively leave a message in the environment and another robot to read it, local sensing involves only a robot sensing its environment.

# RELATED TECHNOLOGIES

A number of technologies that have similarities to swarm robotic networks will now be discussed (see also (Higgins, Tomlinson & Martin 2009b)). These technologies have already been subjected to a degree of security analysis and, after relating them to swarm robotic networks, any unique features of robotic swarms that should be further considered in terms of security are identified.

## Mobile Sensor Networks

A *sensor network* is a collection of devices, sometimes called *nodes*, with sensors that typically communicate over a wireless network (Higgins et al., 2009b). A *mobile sensor network* (MSN) is

a sensor network in which the nodes are moving; the nodes themselves may be mobile or placed on mobile objects.

MSNs share a number of properties with swarm robotics. Both systems assume a potentially large number of mobile nodes that are relatively simple and restricted to local sensing and communication capabilities.

However, swarms exhibit a number of characteristics that MSNs do not, namely emergent behaviour and self-organisation. Furthermore, in most wireless sensor networks nodes will take readings and attempt to communicate this data back to a more powerful device, called a *sink,* that will connect to the network periodically and request data. In contrast, robots distribute their data amongst the swarm, not to a sink, and thus their objective is slightly different.

There is already a considerable amount of research on security in MSNs and exploring the application of existing security solutions to swarms, given the characteristics common to swarms and MSNs, should provide some relevant solutions.

## Mobile Ad-Hoc Networks

An *ad-hoc network*, as defined by (de Morais Cordeiro & Agrawal, 2002, Section 1), is an autonomous system of mobile hosts (also serving as routers) connected by wireless links, the union of which forms a communication network modelled in the form of an arbitrary graph. In a *mobile ad-hoc network* (MANET), the network topology may dynamically change in an unpredictable manner since the nodes are free to move. MANETs have a number of characteristics that are similar to those found in swarm robotics.

1. MANETs lack infrastructure and trusted third parties. Like swarms, they are distributed systems.
2. MANETs can consist of a large number of nodes in the network. Because of this, any security protocols for use in MANETs are applicable to a large number of nodes. Scalability is also important in swarms where solutions should be able to be applied to a large number of robots.
3. Both robots in a swarm and MANETs have limited communication capabilities. It is assumed it is not possible to have all nodes within range of each other (de Morais Cordeiro & Agrawal, 2002, Section 1).
4. Nodes in a MANET are free to move arbitrarily with varying speeds, meaning the network topology may change randomly and at unpredictable times (Duggirala, 2000). A dynamic topology is also a characteristic of swarms.

However, there are some key differences between MANETs and swarms, some of which have been pointed out by Higgins et al. (2009b).

1. Swarms use a wider range of communication methods than MANETs, including stigmergy.
2. The individual identification of nodes in a MANET is normally an important feature, but this is not always the case for swarms. In many applications, individual robots of the swarm are mainly concerned with whether or not another robot is a member of the swarm and not who they are on an individual level. There may, however, be some applications in which individual identity in the swarm is important. One example of this may be when internal adversaries are present, as is described in the internal adversaries section of this chapter.
3. MANETs are not designed to have a collective emergent behaviour like robotic swarms do.
4. MANETs can consist of many types of mobile devices (Higgins et al., 2009b, Section 2) and can be either heterogeneous or homogeneous. Swarms, on the other hand, consist of few homogeneous groups of robots.

There is a considerable amount of work on the security of MANETs which, given the similarities between MANETs and swarms, may be applicable to, or provide an insight into, securing swarm robotic systems.

## VANETs

*Vehicular ad-hoc networks* (VANETs) allow vehicles to wirelessly exchange information about vehicle, road and traffic conditions to other vehicles in order to improve road safety and efficiency.

VANETs are often considered to be a subset of MANETs, with the vehicles acting as the nodes in the network. Despite this, there appear to be some characteristics unique to VANETs and assumptions of MANETs that are not necessarily true in VANETs. Firstly, VANETS do not assume limited computational ability; as noted in (Choffnes & Bustamante, 2007, Section 2), vehicles allow generous limits on power consumption and size for system components. MANETs, conversely, appear to be restricted by heavier constraints. MANETs are also assumed to move in an unpredictable manner, whereas VANETs have a fairly constrained mobility and their trajectories are comparably predictable, mainly because vehicles follow roads. Additionally, VANETS have the ability to detect road and traffic conditions and do so by using sensors, whereas MANETs lack sensors and only communicate between themselves.

Taking into account the sensoring capabilities of the nodes, it may be reasonable to suggest that a VANET be considered as a type of MSN. Moreover, an MSN could be interpreted as a MANET with sensing abilities and a VANET as an MSN but with more predictable movements. A swarm has sensoring abilities and they are able to 'read' from the environment and so VANETs are similar to swarms in this respect.

Furthermore, the fact that VANET nodes have more predictable trajectories than most other MANETs does not necessarily rule them out as being regarded as a MANET; solutions that work for unpredictable trajectories would also work for predictable trajectories, but they are unlikely to utilise the predictable movements of the nodes and are therefore likely to be less efficient than could be achieved. Robots in swarms cannot be assumed to have predictable movements.

An initial impression could lead us to believe that robotic swarms have all the aforementioned abilities of MSNs and MANETs, but also further capabilities and behaviours such as stigmergy and emergent behaviour. However, these behaviours may also be modelled in some other systems. For example, VANETs could be seen to exhibit emergent behaviour; as described in (Choffnes & Bustamante, 2007, Section 2); collections of vehicles favour particular paths based on a number of different factors, such as the number of lanes in a road, traffic-signal behaviour and the proximity of certain roads to motorways etc. These emergent mobility patterns could be used to model emergent behaviour.

VANETs are an interesting technology, with an already significant amount of research conducted into security and the reliability of messages sent. It is worth further considering the extent to which VANET security systems are applicable to swarm robotics and consider the consequences of transferring such solutions. This is done later on in the chapter, when the presence of an internal adversary is scrutinised.

## Multi-Robot Systems

Like robotic swarms, multi-robot systems are collections of robots working together to achieve a common goal (Lima & Custodio, 2005). However, they differ from robotic swarms because they are typically managed by a well-defined command and control structure, which is generally centralised or hierarchical, rather than a distributed system as found in a robotic swarm.

Some obvious similarities between these systems exist. They both consist of mobile, autonomous robots co-operating to achieve a common

goal. In both cases the robots have local sensing and communication abilities and are relatively incapable; having multiple robots rather than a single robot will enable the system to either achieve a goal one robot would not have been able to succeed at individually, or to achieve the goal more efficiently.

There are, however, a number of disparities. As noted previously, an important difference is that robotic swarms are fully distributed and decentralised, compared to having the well-defined command and control structure of a multi-robot system. Multi-robot systems are also not designed to be self-organising or to have an emergent behaviour, which are both important features found in robotic swarms.

Fault tolerance in multi-robot systems with a hierarchical command and control system has been explored previously, most notably by Parker (1998). Such studies and architectures may be applicable to, or when applied may highlight characteristics unique to, robotic swarms given the similarities.

## Software Agents

Although there is no single universally accepted definition of software agents, Wooldridge (2009) defines an *agent* to be a computer system that is situated in some environment and is capable of autonomous action in this environment in order to meet its design objectives.

Borselius (2003, Section 2.2) highlights some widely accepted characteristics of software agents: situatedness, autonomy and flexibility. *Situatedness* is defined as meaning that the agent receives sensory input from the environment and can perform actions that change the environment in some way. Note that the meaning of the term 'environment' varies somewhat, but could be the Internet or a host on which the agent is executing. *Autonomy* has been described previously as the agent having the ability to act without direct intervention of humans and having control over their own actions and internal state. *Flexibility* is defined to include the following properties:

1. **Responsive:** The agent has the ability to perceive and react in a timely manner to its environment.
2. **Pro-active:** The agents have the ability to exhibit opportunistic, goal-driven behaviour and take the initiative where appropriate.
3. **Social:** Agents can interact with other agents and humans in order to solve their problems and help others.

In addition to situatedness, autonomy and flexibility, there are a number of other attributes defined by Borselius (2003) that software agents exhibit. These include: *rationality*, meaning the agent will act in a manner that helps to attain its goals; *veracity*, meaning the agent will not knowingly communicate false information; *benevolence*, meaning an agent cannot have conflicting goals, and *mobility*, meaning the agent has the ability to move across networks and between different hosts to fulfil its goals. An example of a mobile agent would be an agent with the ability to migrate during execution from one host to another where it can then resume execution.

Similar to robotic swarms, software agents have no global system control and the data is decentralised (Borselius, 2003). The individual robots are relatively incapable and computation is asynchronous. Furthermore, if the agents are mobile, a dynamic network is another feature that multi-agent systems have in common with swarms. Because of these similarities, software agents could offer a number of security solutions already existing in the literature that may be applicable to swarms.

However, there are differences that may limit the translation of solutions from software agents to swarms. These differences include the physical nature of swarms, rather than the nature of software, and the use of stigmergy in swarms. However, agents do have the ability to interact with

their environment, but not in a physical manner that is required in the definition of swarms given by Şahin in 2005.

Furthermore, the environment in the software agent setting is different to that in the swarm robotic setting. In software agents, the environment could be active and hostile. A swarm's environment on the other hand is likely to be passive; the robots trust the environment they are in compared to a software agent who may have to be aware of the environment potentially acting as an adversary.

## EXTERNAL ADVERSARIES

In a swarm, there are two kinds of adversary: an *external attacker*, sometimes known as an *intruder*, and an *internal attacker*, sometimes called a *traitor*, such as a robot that has been corrupted. In this section, we focus on protecting the swarm network against an external adversary.

An external adversary is an agent who is assumed to originate from outside the swarm. External adversaries are generally not considered to be present during the deployment of the swarm robotic network. The external adversary is able to watch and analyse the swarm post-deployment and does not have knowledge of any information stored on each individual robot.

It is assumed there are potentially a large number of external attackers who all have similar restrictions to the robots in the swarm; i.e. the adversaries are mobile, autonomous, physically embodied agents with local sensing and communication abilities. They also have limited power and memory and are able to observe, communicate with and analyse members of the swarm.

In the presence of an external adversary, members of the swarm are only concerned with whether another entity is a legitimate member of the swarm or not, and not with individual identification. Therefore the swarm is only required to

*Table 1. Comparison of characteristics of swarms with relevant technologies*

| | | Swarms | MSN | MANET | VANET | Multi-Robot | Software Agent |
|---|---|---|---|---|---|---|---|
| Şahin's explicit Characteristics | Autonomous | ✓ | | | | | ✓ |
| | Large number | ✓ | ✓ | ✓ | ✓ | | |
| | Homogeneous groups | ✓ | ✓ | ✗ | ✓ | ✓ | |
| | Relatively incapable | ✓ | ✓ | | ✗ | ✓ | ✓ |
| | Limited local sensing and communication abilities | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Other characteristics | Co-operate to achieve goals | ✓ | ✓ | | | ✓ | ✓ |
| | Mobile | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | Self-organising | ✓ | | | | | |
| | Collective emergent behaviour | ✓ | ✗ | ✗ | ✓ | | |
| | Decentralised | ✓ | ✗ | ✓ | | ✓ | ✓ |
| | No individual identity necessary | ✓ | ✗ | ✗ | ✗ | | |
| | Lack of synchronicity | ✓ | ✓ | ✓ | | | ✓ |
| | Range of communication | ✓ | ✗ | ✗ | ✓ | | |
| | Active and hostile environment | ✓ | | | | | ✓ |
| | Physical nature | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ |
| | Dynamic network | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

use some form of group identity as a method of distinguishing between external adversaries and swarm members. After our discussion considering challenges found in other technologies, it is clear that communication between legitimate members of the swarm will require some form of secret in order to achieve any level of security.

An external attacker could have a number of goals and can be either active or passive:

1. A *passive* adversary may eavesdrop on the communication between swarm agents to gather information.
2. An *active* adversary may:
    a. Introduce its own agents to masquerade as agents of the swarm to gather information, plant false information or change the swarm's behaviour through its agents' behaviour;
    b. Capture swarm agents in order to extract information from them, remove information from the system, or modify and reintroduce them as its own agents;
    c. Attempt to attack the availability of the system by removing stigmergic messages from the environment.

The first threat can be dealt with by the use of appropriate encryption.

Threat 2a can be dealt with by deploying an authentication mechanism. Note that this does not require authenticating the individual identity of the robot, but rather authenticating a robot as a member of the same swarm.

Capturing agents to remove information from the swarm should have little impact on the swarm due to the high robustness provided by multiple agents gathering similar information. Agents being modified and re-introduced to the swarm can be mitigated by revoking agents that are compromised. Some *key predistribution schemes* (discussed further when addressing challenges unique to the swarm) are designed with this in mind. If we assume that compromised agents can be identified and that a global communication channel for updating software exists, then compromised agents can be revoked by updating keys via this channel.

Denial-of-service is similar to attacks in other types of network (Wood & Stankovic, 2002). This is difficult to prevent in general. However, due to the redundancy of the swarm, there will be other agents leaving the same stigmergic messages, giving some assurance that some messages will reach agents.

Two security solutions are highlighted here that appear to be applicable.

Robots in a swarm may have severe restrictions on resources such as memory, power, computational abilities etc. One solution will be suggested for securing swarms where the robots are not resource constrained, and a second solution will be suggested for swarms where the robots are restricted.

## Robots without Constraints

Robots in a swarm without severe constraints may have large memory capacity, substantial computational ability and abundant power. In a swarm consisting of such robots, one method of securing communications against external adversaries is to use public key cryptography (PKC). Using public key encryption, the data passed between robots can be kept confidential. With the use of a *digital signature* (which is an electronic code attached to a message), integrity and authentication of the transmitted data could also be provided. Hence PKC can provide all the vital security services apart from availability.

The application of PKC in wireless sensor networks (WSNs) has been discussed by Lopez (2006). Since wireless sensors have similar resource constraints to swarm robots, we suggest that the arguments for using PKC in swarms are similar to those for using PKC in WSNs.

As explained by Lopez (2006), the use of PKC would increase the security of the network (here,

swarm), decrease the complexity of key distribution and authentication protocols (compared to using symmetric-key cryptography), and increase the resilience of the network (swarm) as any nodes (robots) captured would only enable the adversary to gain access to keying material of those nodes.

There are several PKC systems that could be used for a swarm, depending on the requirements. For example, the Diffie-Hellman key exchange protocol (Diffie & Hellman, 1976) could be used to establish a session key between robots that can then be used to encrypt data in a more lightweight symmetric cryptosystem. Alternatively, a public key system such as RSA (Rivest, Shamir & Adleman, 1978) or elliptic curve cryptography (Kapoor, Abraham & Singh, 2008) could be deployed.

Despite the advantages of PKC, its application in swarms may be impractical in some settings. PKC is not as efficient as symmetric-key cryptography and so, in a setting with modest computational power and memory available, PKC may not be desirable. Furthermore, as highlighted in (Lopez, 2006), computations such as those used in PKC demand comparatively more energy; the creation and verification of signatures for PKC are also energy intensive operations. When the energy of the robots has been drained, agents will become unavailable and so, eventually, will the swarm. Schemes that contribute to maximising the life of the agents and the swarm may be desirable. However, some advances on increasing the efficiency of PKC have been made (Li & Shen, 2011; Chapin & Skalka, 2010).

The use of PKC tends to require a public key infrastructure (PKI), which creates and manages *digital certificates* that bind public keys to the respective user identities (Ellison & Schneier, 2000). This can be a hard process to implement in a decentralised swarm infrastructure.

We consider how PKC can address the challenges raised by the unique characteristics of the swarm (identity and secrets, stigmergy and local sensing).

## Robots with Constraints

In a swarm where the robots have modest resources, PKC may be too demanding and therefore undesirable. One alternative solution is to use symmetric-key cryptography with keys that have been distributed using a KPS prior to deployment.

Symmetric-key cryptography can be used to secure swarm communications in a way similar to PKC. Symmetric encryption can provide confidentiality, while integrity and authentication can be provided using a *Message Authentication Code (MAC),* a short piece of information generated based on the message and a secret key; Bellare, Canetti and Krawczyk (1996) provide more information on MACs.

Key predistribution in WSNs have certain characteristics that may make them applicable to swarms.

In particular, swarm robots, similar to wireless sensors, are distributed by a trusted central authority that is able to pre-load the agent with information before distribution. In a KPS a central authority distributes (pre-installs) keying information securely prior to deployment. Pairs of agents can derive secret keys later on based on this information for subsequent cryptographic use. The surveys (Martin & Paterson, 2008; Walters et al., 2007) give many examples of KPSs, varying in designs depending on what the network topologies are and on what the focus of the WSNs application is.

One major characteristic of swarm robotics is emergent behaviour, in which local interactions give rise to global behaviour. Because of the redundancy features of a swarm, the loss or compromise of individual agents has little impact on the success of the task. This has similarities with the deployment of WSNs: while WSNs are generally not concerned with emergent behaviour, KPSs used for WSNs are regularly designed so that the loss of a small proportion of the sensors has minimal effect on global connectivity.

In a KPS, a subset of keys are preinstalled onto each node before the swarm is deployed. If two robots have a certain number of keys in common, they are able to communicate. If not, they must establish a path using intermediate agents. KPSs can be designed to prioritise memory, connectivity or resilience. In general, the larger the number of keys given to each agent, the higher the connectivity but the lower the resilience. Hence KPS design revolves around a compromise between these three criteria.

There are two main types of KPS: probabilistic schemes (Eschenauer, Gligor, 2002) in which the subset of keys allocated to each robot is chosen at random, or deterministic schemes, which uses objects such as combinatorial designs as building blocks.

Probabilistic and deterministic KPSs have different advantages and disadvantages, see Martin (2009), Camptepe and Yener (2004) and Lee and Stinson (2008).

Once keys have been distributed using a KPS, some form of lightweight cryptosystem could be used for data encryption. A survey of potential lightweight cryptosystems is given by Eisenbarth, Kumar, Paar, Poschmann and Uhsadel (2007).

## INTERNAL ADVERSARIES

An internal adversary is an attacker within the swarm: a traitor, or a robot that has been captured, re-programmed and re-introduced into the swarm. It is much stronger than an external adversary because it has access to any secret material established prior to deployment and can thus potentially send authenticated messages, even if the messages are false or unreliable.

The goals of an internal adversary are similar to those of an external adversary, as described in the previous section. Any of these attacks may alter the emergent behaviour of the swarm. Thus it is crucial to be able to identify internal adversaries. Once an internal adversary has been identified, their key material can be *revoked*, meaning that their keys can no longer be used for communication within the swarm, thus making them incapable of communicating with their peers and essentially removing them from the swarm robotic system.

This section will consider two different methods for dealing with an adversary inside the system. There are a number of different methods that may be used, but threshold schemes (as used in VANETs) and intrusion detection systems (as used in MSNs) are detailed here because of their applicability to swarms. Note that this is not a comprehensive survey of defence against internal adversaries, but rather, we aim to point out some possible avenues of investigation.

One important issue raised by the presence of an internal adversary is that of individual identity. When the adversary is external, agents in the swarm need only concern themselves with whether or not whom they are communicating with is a member of the swarm, so group identity suffices. However, if it is a member of the swarm that is malicious, this means that agents cannot trust their peers. Some form of individual identity is thus necessary. Without this, the agents would need to continually reassess the individuals around them, as they would otherwise be unable to link the identities of neighbours that leave their neighbourhood and then return. The provision of individual identity is discussed in challenges found in other technologies. Most of the following schemes assume individual identity.

Ideally, one would like to be able to identify internal adversaries and to revoke them. Furthermore, it would be desirable to stop genuine members of the swarm accepting unreliable messages from an internal adversary to prevent their behaviour being swayed by unreliable information.

The use of PKC in a swarm has been discussed when considering external adversaries with robots without constraints. Although a PKC scheme would not be able to help identify an internal adversary, once the adversary has been identified the scheme could remove the public key certificate for

the adversary, stopping them from communicating with other members of the swarm. However, applying this in practice to a distributed swarm poses a number of problems with certificate management. Key predistribution also does not help in identifying internal adversaries. However, one could deal with the risk of internal adversaries by prioritising resilience in the KPS chosen.

There are other methods that could be used to identify and revoke internal adversaries. We describe a threshold method used in VANETs, which could be adapted for swarms, and an intrusion detection systems used in MSNs.

## Threshold Schemes in VANETs

There exists a plethora of systems proposed for VANETs that address how to identify a traitor vehicle that is sending unreliable messages.

As described by Chen, Li, Martin, and Ng (2013), VANETs allow vehicles to exchange information about vehicle, road and traffic conditions wirelessly. A *reliable* message is a message that reflects reality. An *unreliable* message, which may be sent either intentionally (by an internal adversary) or due to a hardware malfunction, can have undesirable consequences, such as journey delays or accidents, if the recipient wrongly considers the message to be reliable. There are a number of solutions to evaluating the reliability of messages in VANETs in the literature.

The majority of suggested schemes for VANETs are *threshold schemes*; this is where a message is believed to be reliable if a number of distinct vehicles above a threshold announce the same message within a given time interval. A number of threshold schemes have been proposed, such as in (Chen, Ng, & Wang, 2011; Daza, Domingo-Ferrer, Sebé, & Viejo, 2009; Raya, Aziz, & Hubaux, 2006; Kounga, Walter, & Lachmund, 2009; Wu, Domingo-Ferrer, & González-Nicolás, 2010). Many of these schemes are complicated by the requirements of privacy and anonymity in the VANET system (which, while important requirements in a VANET are not necessarily vital in a swarm), and many of these schemes use resource-intensive public key cryptographic primitives such as variants of digital signatures. However, many assume that while there is a central authority to help set up the scheme (legitimising vehicles, issuing certificates to entities and performing annual health checks of vehicles and other such procedures), this authority is not online in the running of the scheme. This reflects the situation of swarm robotic systems - before deployment, the robots are able to be preloaded with some information, such as keys and certificates, but after being deployed there may be no method of global communication with the authority. However, there may be some applications of swarms where a global channel of communication with the authority is available, and thus this would not be a problem.

As VANETs generally consist of a large number of vehicles, the majority of solutions tend to be scalable and so do not pose a problem with respect to scalability when used in swarm robotics. Most of the schemes mentioned here also discuss the possibility of revocation, a feature that is important to have in a scheme in order to prevent traitor robots swaying the emergent behaviour.

Finally, it is noted that the schemes here all rely on each robot having an individual identity, rather than just a group identity. Designing a successful threshold scheme in which there is no individual identity appears to be an interesting and difficult problem as, without an identity, the adversary would be able to send the same message enough times to pass the threshold, resulting in their messages being accepted. One suggestion may be to use some form of temporary pseudonym, but one would require the robot to not be able to change pseudonym and send the same message under a different pseudonym.

While a threshold scheme allows robots to act only on reliable messages, it would also be possible to build into the scheme a detection system to record robots that send unreliable messages. However, the

central authority has to be informed in order for it to perform a revocation. Without a communication channel back to the central authority, there does not seem to be a straightforward solution.

## Intrusion Detection Systems

Another method for identifying an intruder that is regularly used in MSNs is an *intrusion detection system* (IDS). Intrusion detection systems consist of an audit collection agent that collects information about the system being observed. This data is then either stored or processed directly and presented to the site security officer (SSO), who will take the necessary action.

There are two generally accepted types of intrusion detection: *anomaly-based detection* and *misuse-based detection* (Axelsson, 2000; Sun, Osborne, Xiao, & Guizani, 2007).

In anomaly detection, all behaviour that is abnormal to the system is flagged. This is achieved by the creation of normal profiles of system states and comparing this to the system's current behaviour. According to Axelsson (2000), anomaly-based detection schemes can be either self-learning or programmed. Self-learning systems typically observe the system for a period of time and build a model of what is considered normal. On the other hand, a programmed system requires someone to program the IDS to detect specified anomalous events.

For an *anomaly-based detection scheme* to be applicable to swarm robotics it must be a real-time system so the agents are able to recognise an intruder as soon as possible. The data should be sourced from the network and both the data collecting and data processing should be distributed, due to the decentralised nature of the swarms. When an intruder is identified, either the swarm responds passively to the intruder (by reporting back to a central authority, which may be difficult given the inability to communicate with the central authority) or the swarm takes an active approach where the intruder is disabled or hindered.

According to (Axelsson, 2000), there are very few schemes that fulfil these requirements. However, there is one scheme that may be applicable. Proposed by Porras and Neumann in 1997, EMERALD is an IDS that is a real-time, continuous, distributed system that has an active response to an intruder and can collect data from both the host and network.

However, anomaly based IDSs may be of limited use when applied to swarms. One reason for this is due to the problem of establishing what behaviour is normal. This is likely to be especially hard in a swarm, where the agents have an emergent behaviour that arises from local interactions. This emergent behaviour cannot always be predicted and may not be obvious. What may appear to be an anomalous behaviour may in fact be the first sighting of a desirable behaviour that could develop into global behaviour and help the swarm to become more efficient. Stemming anomalous behaviour on a local level could prevent the emergent behaviour of the swarm from evolving so that any goals are achieved more efficiently.

In a *misuse-based detection scheme* (Sun et al., 2007), alternatively called a *signature scheme* (Axelsson, 2000), behaviour that is sufficiently close to some previously defined pattern signature of a known intrusion is flagged. In the taxonomy proposed by Axelsson (2000), it is said that signature detection schemes are programs with an explicit decision rule that contains what can be expected to be observed in the event of an intrusion. The programmers can list what can be seen as illegal behaviour; any behaviour that is not on the list can be considered to be legal.

A misuse-based detection scheme may be of limited use in a swarm robotic network because of its inability to detect new attacks. However, it could be possible to make a list of illegal behaviours that would obviously conflict with the goal of the system. For example, if the desired collective behaviour is that of immediate aggregation, any robots that are obviously showing signs of dispersion could be highlighted as adversaries

because their behaviour is obviously not helping the swarm robotic network achieve its goal. However, behaviours such as this in a more complex system may be hard to find and limit the flexibility of the system.

In conclusion, IDSs may not be the best way to detect internal adversaries due to the difficulty of establishing what behaviour is normal in an anomaly based IDS, and an inability to protect against novel attacks in a misuse-based IDS.

Most of the schemes here appear to have a number of limitations when applied to a swarm. The nature of the infrastructure of swarms makes it hard to apply these schemes. In order to successfully apply some of these schemes, a number of assumptions about the swarm may have to be relaxed. For example, if the swarm were allowed some form of command and control centre, there would be a lot more flexibility as to what schemes can be applied.

## ADDRESSING CHALLENGES UNIQUE TO SWARM ROBOTIC NETWORKS

We previously identified three security challenges unique to swarm robotic networks: identity and secrets, stigmergy and local sensing. Here, we will discuss a number of ways of addressing these challenges and demonstrate how existing security solutions may be applicable.

### Identity and Secrets

Recall that, in order to provide a number of security services, the identification of robots is vital. Robots can either be identified as individuals, or as part of a swarm.

There are a number of methods of identification that could be used, both on an individual or group level. Using some form of public identification (ID) code or a name is an obvious, but naïve, approach; it would be easy for an external adversary to read the ID code of a legitimate member (or similarly the swarm code or name) and then impersonate them. This is sometimes called a *spoofing* attack. As well as being resistant to spoofing, identities must also be unforgeable. ID codes and names are neither, and are thus not secure enough for identification in this setting.

Identifying a member of the swarm by a set of common behaviours (Schmidt, Leinmüller, Schoch, Held, & Schäfer, 2008), physical traits (Russell, 2004) or other public descriptors have been suggested. Again, however, characteristics such as these can be observed and copied by external adversaries. It would thus seem necessary to use some form of secret as identification. This secret, for example a secret key, could be used for either group or individual identification.

#### Group Identification

To protect against an external adversary, it is sufficient for each robot to authenticate other robots as members of the swarm and not individually. One possible way to provide group identity is by using symmetric-key cryptography. This could be a way of identifying the robots in a resource constrained swarm in the presence of an external adversary, as has been done previously when discussing external adversaries. A common key could be distributed amongst the swarm prior to deployment that will be stored and used for encryption and decryption by each robot. If a member of the swarm receives a message encrypted using the common key in the system, then they conclude that the sender is a legitimate member of the swarm. Similarly, a sender could assume that any messages they send will only be read by legitimate members of the swarm as they are the only ones in possession of the necessary key to decrypt the message.

However, this use of a single common key would not be resilient; if a robot is captured, all the stored key material is available to the adversary and so any communications using this common key will be insecure. There are other ways that

keys could be distributed that would provide different levels of resilience whilst compromising on memory and connectivity. The robots can authenticate each other as members of the same swarm due to having common keys. One way to distribute these keys is through the use of a KPS, which has been described previously. In general the more keys a robot has the higher the connectivity (the ability of robots to communicate with each other) and the lower the resilience.

Other ways of providing group identity are available and applicable to swarms. Harney and Muckenhirn (1997) and Wong, Gouda and Lam (1998) suggest other ways in which group identification can be achieved. In Harney and Muckenhirn's scheme (1997), they combine techniques developed for the creation of pairwise keys along with techniques used to distribute symmetric keys from a key distribution centre to a network. Wong et al. (1998) use key graphs to specify secure groups.

### Individual Identification

With PKC robots could be uniquely identified by their public key. Members of the swarm could verify any communications to or from the individual; only the member with the associated decryption key could decrypt a message encrypted with the public key, and members could provide a cryptographic signature using their private key to any messages sent. If this system is used with a *nonce* (an arbitrarily chosen number to be used only once with each communication), adversaries are unable to conduct a *replay attack*, in which valid messages are maliciously delayed or repeated. This method is then both spoof-resistant and unforgeable, because members of the swarm can ensure that messages go to or come from specific members. However, this method does pose key management problems: as described by Marti and Garcia-Molina (2006), initially transmitting one's public key may be susceptible to man-in-the-middle attacks. Having certificates signed by a trusted certificate authority can stop this problem, but requires a centralised authority, which is undesirable in the swarm setting. Additionally, this method of identification involves high computational costs during encryption and decryption.

Identification may also be achievable using symmetric-key cryptography. Rather than giving every member of the swarm the same key, different robots could be given different sets of keys so that there are common keys shared by pairs in the swarm. This would prove more resilient than every robot having the same key and could provide individual identity if each swarm member is uniquely identifiable from the set of keys it holds.

## Stigmergy

There are a number of ways that stigmergy can be modelled that can enable it to be treated similarly to direct communication.

Firstly, as described by Şahin (2005), an individual robot could opt to become immobile and act as a stigmergic medium to guide the rest of the swarm. This method considers the immobile robot to be part of the environment; other robots of the swarm leave a message with them and this message is then broadcast to other robots that pass within range.

Secondly, instead of robots opting to become a stigmergic medium, specific sensor nodes can be placed in the environment as stigmergic media. This is analogous to a vehicular ad hoc and sensor network (VASNET), as described by (Piran, Murthy, & Babu, 2011), where a network is made up of communication and computing devices embedded into vehicles and roadside sensor nodes are deployed at predetermined distances beside the road to relay messages.

Another method described in (Şahin, 2005) uses embedded intelligent markers in the environment that can store stigmergic information and interact with each other to simulate physical diffusion like signal spreading.

One further way of modelling stigmergy is by using software agents that interact with their environment. Software agents 'read' information from their environment and 'write' by altering the environment they are in. However, the virtual nature of agents and the potentially active environment in which they are situated may limit their use when modelling stigmergy.

If any of these methods were used to model stigmergy, then communications could be secured in standard ways. For example, indirect messages could be encrypted with either a symmetric or a public key and left in the environment. Furthermore, the use of short-range communication, which is typically used in swarm robotics, would ensure that only agents in the vicinity could communicate. Messages can also be made valid for a certain period of time by using a time stamp and a digital signature on this stamp.

## Local Sensing

Recall that local sensing refers to how an agent's behaviour is affected by what the agent can observe in its immediate surroundings, such as the configuration or behaviour of its neighbours. A robot needs to ensure that any behaviours it takes into account are from legitimate agents. This is possible if agents can authenticate each other.

If PKC is used, robots could use digital signatures. With this tool, robots in the swarm could authenticate each other and ensure that only behaviour by authenticated members of the swarm influence their own behaviour. With the use of a KPS and symmetric-key cryptography, the robots could use MACs to authenticate each other.

# FURTHER RESEARCH

In the preceding sections we discussed the requirements for secure communication in swarm robotics.

There are many open problems still to be considered:

1. We have shown that the use of a KPS would be an effective method for defending against external adversaries. It would be interesting to further study the properties of KPS that are useful for swarms with particular requirements. KPSs may have an adverse effect on the efficiency of the swarm because each robot can only communicate with another entity in its neighbourhood that is in possession of a common key. Implementing a KPS can only either keep constant or decrease the number of robots an individual entity can communicate with in its neighbourhood. Because of this reduced connectivity, the processes of the swarm will be slowed down. A KPS with maximal connectivity could be chosen in order to increase efficiency, but this would mean a compromise in resilience and memory. It would also be interesting to quantify this trade-off for particular swarms.
2. An interesting topic of research would be to further consider software agents and to what extent they could be used to model a swarm robotic network. Although software agents are not autonomous like the robots in a swarm, they appear to be able to communicate implicitly and so may be useful in modelling stigmergy. It is also unclear as to whether or not the physical manner of swarms or the nature of their environment would limit their use of modelling stigmergy and the swarm network.
3. The modelling of stigmergy requires further study. A possible tool for authentication using environmental data is keyed robust fuzzy extractors (Dodis, Katz, Reyzin & Smith, 2006). Keyed robust fuzzy extractors allow two agents sharing a secret key to derive a session key using correlated input, for example, environmental data such as GPS

coordinates, temperature variations, or time, which are not too different from each other if the agents are neighbours.

Simplistically speaking, *A* and *B* can agree on a new key if they share the same secret (which is true if they belong to the same swarm) and if they can get closely correlated samples (near to each other). For example, the value *P* can determine what colour lights *A* emits and if *B* can see this than *B* can generate *R*. This allows "sensing". It is also possible that this technique may be used to model pheromone-type communication, where traces left by an agent fade over time and are only acceptable if they are read quickly.

Note that we still need the KPS to predistribute secret keys. After that environmental variables can be used to modify the keys. The secret keys are needed since environmental variables are public and can also be read by adversaries. It will be interesting to see if such a scheme is workable.

4. It would be interesting to study what can be achieved and what schemes can be applied to the swarm network if some of our swarm requirements were relaxed. For example, if a broadcast channel is allowed and available, a scheme allowing flexibility in connectivity and efficient revocation is possible (Kendall, Martin, Ng, Paterson & Stinson, 2014). With many KPSs, there is a fixed probability that two agents will be able to authenticate each other and this probability is decided upon before deployment. If we have a broadcast channel, we are able to change this parameter during deployment, according to any changes in the swarm or in the environment. In addition, if we know which agents have been compromised they can be revoked. Hence we see that if the swarm is allowed some form of command and control structure, much more may be achieved with potentially low costs.
5. Some possible methods of defence against an internal adversary have been studied here, but none of them appeared to be sufficiently effective. Threshold schemes appear to be the most applicable to swarms, but they still have a number of limitations. Further suggestions for security solutions should be investigated and, due to the individual requirements of swarms, some schemes could be adapted to better suit the environment. It also appears to be the case that individual identity may be important: one must be able to distinguish between individuals in a swarm if it is one of the swarm that is an adversary. Methods of providing individual identity to the robots in a swarm network would be an interesting problem to further explore.

## CONCLUSION

In this chapter, we have discussed a number of ways to secure communications in swarm robotics. After defining several aspects of security and briefly discussing how these can be achieved, we considered a number of applications of swarms and identified the security services that may be vital in a swarm setting. We considered the challenges posed by the nature of the swarm network; some challenges were common and had been considered in similar technologies, whereas other challenges, such as stigmergy, appeared to be unique to swarms.

After considering the challenges, a number of technologies that have already have well researched security solutions were discussed. These technologies had a number of characteristics in common with swarms and hence their solutions may be applicable and adaptable to swarms.

It is important that security is not forgotten in the rush to deploy robotic swarms. We have shown that established techniques will go a long way towards making swarm robotic environments robust to the main threats to their security.

## REFERENCES


Akbani, R., Korkmaz, T., & Raju, G. (2012). Mobile ad-hoc networks security. Recent advances in computer science and information engineering, 127, 659-666. Springer.

Axelsson, S. (2000). *Intrusion Detection Systems: A Survey and Taxonomy,* Technical Report No. 99-15, Dept. of Computer Eng., Chalmers Univ. of Technology, Sweden.

Bellare, M., Canetti, R., & Krawczyk, H. (1996, January). Keying hash functions for message authentication. Proceedings of Advances in Cryptology CRYPTO '96 (pp. 1-15). Springer Berlin Heidelberg. doi:10.1007/3-540-68697-5_1

Beni, G. (2005). From swarm intelligence to swarm robotics. Swarm robotics, 3342, 1-9. Springer.

Berghman, J., & Deckers, K. (2013). *Classification and modelling of attacks on swarm robotic systems in collective exploration* [Unpublished master's thesis]. Leuven Engineering College, Belgium.

Borselius, N. (2003). *Multi-Agent System Security for Mobile Communications* [PhD thesis]. Royal Holloway, University of London.

Brambilla, M., Ferrante, E., Birattari, M., & Dorigo, M. (2013). Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence*, *7*(1), 1–41. doi:10.1007/s11721-012-0075-2

Camazine, S., Deneubourg, J., Franks, N., Sneyd, J., Theraulaz, G., & Bonabeau, E. (2002). *Self-organization in biological systems*. Princeton: Princeton University Press.

Camtepe, S. A., & Yener, B. (2004). Combinatorial design of key distribution mechanisms for wireless sensor networks. Proceedings of the Computer Security ESORICS 2004, Lecture Notes in Computer Science (Vol. 3193, pp. 293-308). Springer.

Chapin, P., & Skalka, C. (2010). SpartanRPC: Secure WSN middleware for cooperating domains. *Proceedings of the 2010 IEEE 7th International Conference on Mobile Adhoc and Sensor Systems (MASS)* (pp. 61-70). California, US: IEEE. doi:10.1109/MASS.2010.5663965

Chen, L., Li, Q., Martin, K. M., & Ng, S. (2013). A privacy-aware reputation-based announcement scheme for VANETs. *Proceedings of the 2013 IEEE 5th International Symposium on Wireless Vehicular Communications WiVeC.* (pp. 1-5). IEEE.

Chen, L., Ng, S., & Wang, G. (2011). Threshold anonymous announcement in VANETs. *IEEE Journal on Selected Areas in Communications*, *29*(3), 605–615.

Choffnes, D., & Bustamante, F. E. (2007). Exploiting emergent behavior for inter-vehicle communication. Proceedings of HotAC II: Hot Topics in Autonomic Computing on Hot Topics in Autonomic Computing. Berkeley, CA, USA: USENIX Association.

Daza, V., Domingo-Ferrer, J., Sebé, F., & Viejo, A. (2009). Trustworthy privacy-preserving car-generated announcements in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, *58*(4), 1876–1886.

de Morais Cordeiro, C., & Agrawal, D. P. (2002). Mobile ad hoc networking. Center for Distributed and Mobile Computing, ECECS, University of Cincinnati.

Diffie, W., & Hellman, M. E. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, *22*(6), 644–654.

Dodis, Y., Katz, J., Reyzin, L., & Smith, A. (2006). Robust fuzzy extractors and authenticated key agreement from close secrets. Proceedings of Advances in cryptology-CRYPTO 06 (pp. 232–250). Springer Berlin Heidelberg. doi:10.1007/11818175_14

Dolev, S., Lahiani, L., & Yung, M. (2007). Secret swarm unit reactive k− secret sharing. Proceedings of *Progress in Cryptology–INDOCRYPT '07* (pp. 123–137).

Duggirala, R. (Dec 2000). *A novel route maintenance technique for ad hoc routing protocols* [Master's thesis]. University of Cincinnati.

Eisenbarth, T., Kumar, S., Paar, C., Poschmann, A., & Uhsadel, L. (2007). A survey of lightweight-cryptography implementations. *IEEE Design & Test of Computers*, *24*(6), 522–533. doi:10.1109/MDT.2007.178

Ellison, C., & Schneier, B. (2000). Ten risks of PKI: What you're not being told about public key infrastructure. *Computer Security Journal*, *16*(1), 1–7.

Eschenauer, L., & Gligor, V. D. (2002, November). A key-management scheme for distributed sensor networks. *Proceedings of the 9th ACM conference on Computer and communications security* (pp. 41-47). ACM. doi:10.1145/586110.586117

European Union: 6th framework programme. (2007). Guardians project. Retrieved from http://www.guardians-project.eu/

European Union: 6th framework programme. (2007). IWARD: Intelligent robot swarm for attendance, recognition. Retrieved from www.iward.eu

Flocchini, P., Prencipe, G., Santoro, N., & Widmayer, P. (2005). Gathering of asynchronous robots with limited visibility. *Theoretical Computer Science*, *337*(1), 147–168. doi:10.1016/j.tcs.2005.01.001

Harney, H. & Muckenhirn C. (1997) Group key management protocol (GKMP) architecture *RFC 2094*.

Higgins, F., Tomlinson, A., & Martin, K. M. (2009). Survey on security challenges for swarm robotics. *Autonomic and Autonomous Systems, 2009. ICAS'09. Fifth International Conference on,* 307-312. Valencia, Spain. IEEE.

Higgins, F., Tomlinson, A., & Martin, K. M. (2009). Threats to the swarm: Security considerations for swarm robotics. *International Journal on Advances in Security*, *2*(2&3), 288–297.

International Organization for Standardization (ISO). (1989). Information processing systems -- open systems interconnection -- basic reference model -- part 2: Security architecture std. 7498-2, rev. 1. Retrieved from http://www.iso.org/iso/catalogue_detail.htm?csnumber=14256

Kapoor, V., Abraham, V. S., & Singh, R. (2008). Elliptic curve cryptography. *ACM Ubiquity*, *9*(20), 20–26.

Kendall, M., Martin, K. M., Ng, S. L., Paterson, M. B., & Stinson, D. R. (2014). Broadcast-enhanced key predistribution schemes. *ACM Transactions on Sensor Networks*, *11*(1), 6. doi:10.1145/2629661

Kounga, G., Walter, T., & Lachmund, S. (2009). Proving reliability of anonymous information in VANETs. *Vehicular Technology. IEEE Transactions on*, *58*(6), 2977–2989.

Lee, J., & Stinson, D. R. (2008). On the construction of practical key predistribution schemes for distributed sensor networks using combinatorial designs. *ACM Transactions on Information and System Security*, *11*(2), 1–35. doi:10.1145/1330332.1330333

Li, W., & Shen, W. (2011). Swarm behavior control of mobile multi-robots with wireless sensor networks. *Journal of Network and Computer Applications*, *34*(4), 1398–1407. doi:10.1016/j.jnca.2011.03.023

Lima, P. U., & Custodio, L. M. (2005). Multi-robot systems. In *Innovations in robot mobility and control* (pp. 1–64). Springer Berlin Heidelberg.

Lopez, J. (2006). Unleashing public-key cryptography in wireless sensor networks. *Journal of Computer Security*, *14*(5), 469–482.

Marti, S., & Garcia-Molina, H. (2006). Taxonomy of trust: Categorizing P2P reputation systems. *Computer Networks*, *50*(4), 472–484. doi:10.1016/j.comnet.2005.07.011

Martin, K. M. (2009). *On the applicability of combinatorial designs to key predistribution for wireless sensor networks. Proceedings of Coding and Cryptology (IWCC2009)*, Lecture Notes in Computer Science (Vol. 5557, pp. 124–145). Springer.

Martin, K. M., & Paterson, M. (2008). An application-oriented framework for wireless sensor network key establishment. *Electronic Notes in Theoretical Computer Science*, *192*(2), 31–41. doi:10.1016/j.entcs.2008.05.004

Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A. (1996). *Handbook of applied cryptography*. CRC press. doi:10.1201/9781439821916

Navarro, I., & Matía, F. (2012). An introduction to swarm robotics. *ISRN Robotics, 2013.*

Parker, L. E. (1998). ALLIANCE: An architecture for fault tolerant multirobot cooperation. *IEEE Transactions on Robotics and Automation*, *14*(2), 220–240.

Piran, M. J., Murthy, G. R., & Babu, G. P. (2011). Vehicular ad hoc and sensor networks; principles and challenges. *ArXiv Preprint arXiv:1108.2776*

Porras, P. A., & Neumann, P. G. (1997, October). EMERALD: Event monitoring enabling response to anomalous live disturbances. *Proceedings of the 20th national information systems security conference,* Baltimore, MD (pp. 353-365).

Raya, M., Aziz, A., & Hubaux, J. P. (2006). Efficient secure aggregation in VANETs. *Proceedings of the 3rd international workshop on Vehicular ad hoc networks* (pp. 67-75). ACM. doi:10.1145/1161064.1161076

Raya, M., & Hubaux, J. (2007). Securing vehicular ad hoc networks. *Journal of Computer Security*, *15*(1), 39–68.

Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, *21*(2), 120–126. doi:10.1145/359340.359342

Russell, R. A. (2004). Visual recognition of conspecifics by swarm robots. *Proceedings of the 2004 Australasian Conference on Robotics & Automation*.

Saez-Pons, J., Alboul, L., Penders, J., & Nomdedeu, L. (2010). Multi-robot team formation control in the GUARDIANS project. *Industrial Robot: An International Journal*, *37*(4), 372–383. doi:10.1108/01439911011044831

Şahin, E. (2005). Swarm robotics: From sources of inspiration to domains of application. *Swarm robotics* (pp. 10–20). Springer Berlin Heidelberg. doi:10.1007/978-3-540-30552-1_2

Schmidt, R. K., Leinmüller, T., Schoch, E., Held, A., & Schäfer, G. (2008). Vehicle behavior analysis to enhance security in vanets. Proceedings of the 4th IEEE Vehicle-to-Vehicle Communications Workshop (V2VCOM2008)

Sun, B., Osborne, L., Xiao, Y., & Guizani, S. (2007). Intrusion detection techniques in mobile ad hoc and wireless sensor networks. *IEEE Wireless Communications*, *14*(5), 56–63. doi:10.1109/MWC.2007.4396943

Thiel, S., Habe, D., & Block, M. (2009, November). Co-operative robot teams in a hospital environment. *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems ICIS 09* (Vol. 2, pp. 843-847). IEEE. doi:10.1109/ICICISYS.2009.5358271

Trappe, W., Washington, L., Anshel, M., & Boklan, K. D. (2007). Introduction to cryptography with coding theory. *The Mathematical Intelligencer*, *29*(3), 66–69. doi:10.1007/BF02985694

Walters, J. P., Liang, Z., Shi, W., & Chaudhary, V. (2007). Wireless sensor network security: A survey. *Security in Distributed, Grid, Mobile, and Pervasive Computing, 1*, 367.

Winfield, A. F., & Nembrini, J. (2006). Safety in numbers: Fault-tolerance in robot swarms. *International Journal of Modelling. Identification and Control*, *1*(1), 30–37. doi:10.1504/IJMIC.2006.008645

Wong, C. K., Gouda, M., & Lam, S. S. (1998). Secure group communications using key graphs. *Computer Communication Review*, *28*(4), 68–79. doi:10.1145/285243.285260

Wood, A., & Stankovic, J. A. (2002). Denial of service in sensor networks. *Computer*, *35*(10), 54–62. doi:10.1109/MC.2002.1039518

Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.

Wu, Q., Domingo-Ferrer, J., & González-Nicolás, U. (2010). Balanced trustworthiness, safety, and privacy in vehicle-to-vehicle communications. *IEEE Transactions on Vehicular Technology*, *59*(2), 559–573.

## KEY TERMS AND DEFINITIONS

**Availability:** Availability ensures that accessibility and usability are available upon demand by an authorised entity. The loss of availability is commonly referred to as a denial-of-service attack.

**Confidentiality:** The assurance that no one other than the intended recipient(s) can read the data; in other words, the data is kept secret between the sender and recipient.

**Cryptography:** Cryptography is the art (and science) of designing cryptosystems to transform messages in such a way that two entities may communicate securely over an insecure channel.

**Data Origin Authentication:** A security service that identifies a specific entity as the source or origin of a given piece of data.

**Entity Authentication:** Sometimes called *identification*, entity authentication is a security service that identifies specific entities in isolation from any other activity the entity may want to perform.

**External Adversary:** An external adversary is an agent who, unlike an internal adversary, is assumed to originate from outside the swarm and thus is not in possession of any secret material established prior to deployment.

**Identification:** Sometimes called *entity authentication*, identification is a security service that identifies specific entities in isolation from any other activity the entity may want to perform.

**Integrity:** The assurance that the data has not been altered, either maliciously or accidentally, in an unauthorised way.

**Internal Adversary:** An internal adversary is an attacker from within the swarm that has access to any secret material established prior to deployment.

**Resilience:** A swarm is resilient if the loss of individual agents has little impact on the success of the task of the swarm.