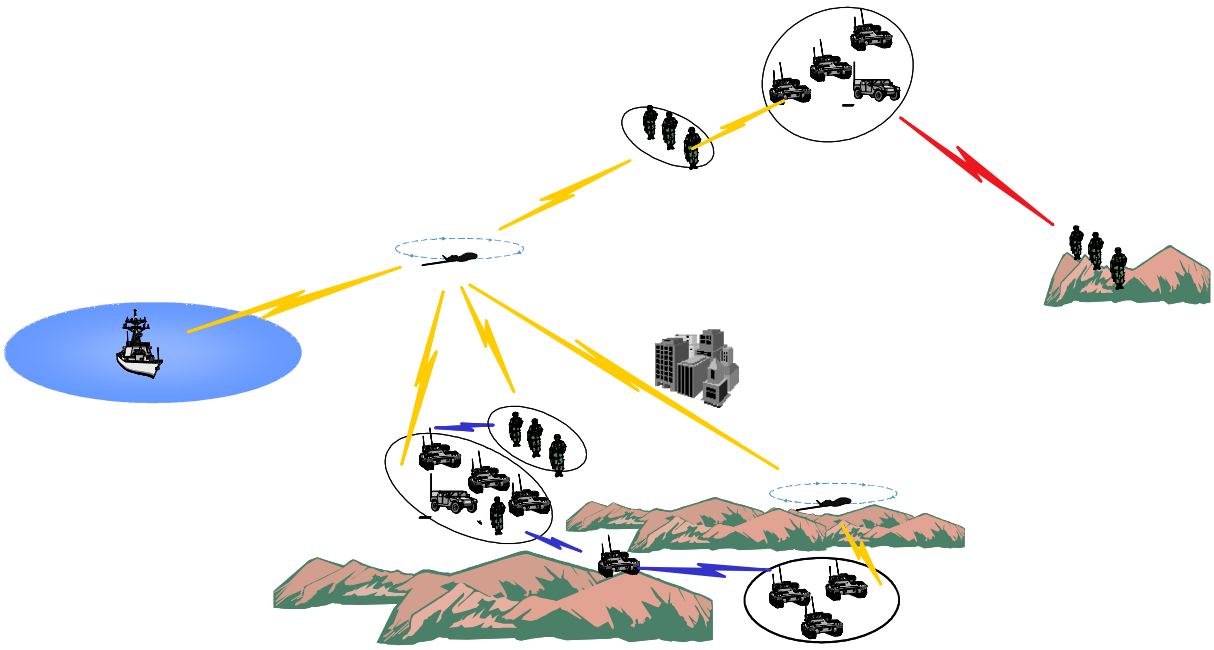
Security solutions for mobile ad hoc networks

Elisabeth Hansson, Alf Bengtsson,
Arne Vidström



FOI-R--1694--SE
ISSN 1650-1942

Command and Control Systems
User report

August 2005

FOI is an assignment-based authority under the Ministry of Defence. The core activities are research, method and technology development, as well as studies for the use of defence and security. The organization employs around 1350 people of whom around 950 are researchers. This makes FOI the largest research institute in Sweden. FOI provides its customers with leading expertise in a large number of fields such as security-policy studies and analyses in defence and security, assessment of different types of threats, systems for control and management of crises, protection against and management of hazardous substances, IT-security and the potential of new sensors.

FOI
Defence Research Agency
Command and Control Systems
P.O. Box 1165
SE-581 11 Linköping

Tel: 013-378086
Fax:

www.foi.se

Security solutions for mobile ad hoc networks

FOI-R--1694--SE
ISSN 1650-1942

Command and Control Systems
User report

August 2005

Issuing organization	Report number, ISRN	Report type
FOI – Swedish Defence Research Agency Command and Control Systems P.O. Box 1165 SE-581 11 Linköping	FOI-R--1694--SE	User report
	Research area code 4. C4ISTAR	
	Month year August 2005	**Project no.** E7506
	Sub area code 41 C4I	
	Sub area code 2	

Author/s (editor/s)	Project manager
Elisabeth Hansson Alf Bengtsson Arne Vidström	Elisabeth Hansson
	Approved by Martin Rantzer
	Sponsoring agency Försvarets Materielverk
	Scientifically and technically responsible Jonas Hallberg

Report title

Security solutions for mobile ad hoc networks

Abstract (not more than 200 words)

This study is a 350-hour project funded by the program "***Gemensamt Taktiskt Radio System, Material System 463"***. The purpose of this report is to first review the state of the art in mobile ad hoc networks security and then identify the security solutions that are relevant for further discussion. The work results in a conceptual security architecture. However, this document does not define the security solutions to be used in tactical mobile ad hoc networks. This will be dealt with in collaboration with MUST/TSA and FMV.

Keywords

Security, mobile ad hoc networks, authentication, key management, secure routing, distributed firewall, intrusion detection and response, authorization

Further bibliographic information	Language English
ISSN 1650-1942	**Pages** 72 p.
	Price acc. to pricelist

Utgivare	Rapportnummer, ISRN	Klassificering
FOI - Totalförsvarets forskningsinstitut Ledningssystem Box 1165 581 11 Linköping	FOI-R--1694--SE	Användarrapport
	Forskningsområde 4. Ledning, informationsteknik och sensorer	
	Månad, år Augusti 2005	**Projektnummer** E7506
	Delområde 41 Ledning med samband och telekom och IT-system	
	Delområde 2	

Författare/redaktör	Projektledare
Elisabeth Hansson Alf Bengtsson Arne Vidström	Elisabeth Hansson
	Godkänd av Martin Rantzer
	Uppdragsgivare/kundbeteckning Försvarets Materielverk
	Tekniskt och/eller vetenskapligt ansvarig Jonas Hallberg

Rapportens titel (i översättning)

Säkerhetslösningar för mobila ad hoc-nät

Sammanfattning (högst 200 ord)

Den här studien är ett 350-timmars uppdrag sponsrat av programmet ”Gemensamt Taktiskt Radio System, Material system 463”. Syftet med rapporten är först granska standarder och forskningsförslag inom området säkerhet för mobila ad hoc-nät. Därefter identifieras säkerhetslösningar som är relevanta for vidare diskussion. Arbetet resulterar i en konceptsäkerhetsarkitektur. Dock bestämmer inte detta dokument vilka säketskrav och säkerhetslösningar, som skall användas. Detta bestämms i sammarbete med MUST/TSA och FMV.

Nyckelord

Säkerhet, mobila ad hoc-nät, authenticering, nyckelhantering, säker routing, distribuerade brandväggar, intrångsdetektering med respons, access control.

Övriga bibliografiska uppgifter	Språk Engelska
ISSN 1650-1942	**Antal sidor:** 72 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Table of Contents

Distributed:
Date:

1 Introduction

Securing mobile ad hoc networks is a challenge. A mobile ad hoc network consists of wireless nodes that form a radio network without any pre-existing infrastructure or centralized servers. One main challenge to these networks results from their vulnerability to security attacks in a hostile environment. Mobile ad hoc networks are particularly vulnerable due to their features of open architecture, cooperative distributed algorithms, limited physical protection, and self-organizing network. These features also result in that many existing security solutions for wired networks and traditional tactical radio networks are not applicable to mobile ad hoc networks. Furthermore, the unique characteristics of mobile ad hoc networks, such as resource constraints and dynamic network topology, result in a number of challenges to security design. Security problems in mobile ad hoc networks are further described in [1].

The purpose of this report is to first review the state of the art in mobile ad hoc networks security and then identify the security solutions that are relevant for further discussion. The work results in a conceptual security architecture which is presented in section 3.

A result of this report is an identification of security solutions that improve the security compared with the current solutions for tactical radio networks. The current solutions are mainly based on encryption at the data link layer and spread spectrum methods at the physical layer. Encryption and spread spectrum methods can not protect against all different types of possible attacks against mobile ad hoc networks [1].

Another result of this report is that it is possible to put strong and reasonable requirements on a future supplier of a tactical mobile ad hoc network, since we know more about what is possible (and not) regarding security solutions for mobile ad hoc networks.

However, this document does not define the security solutions to be used in tactical mobile ad hoc networks. Thus, the purpose of this report is *not* to derive security requirements or recommend a specific security solution. Security requirements should be derived from an *security analysis*, where important parameters are how the system is used (and can be misused), the environment, and the system (functional requirements, applications, protocols etc.). It is important to understand that *security design should be considerable as an indispensable part of the systems development process, not as mechanisms that can be added afterwards*. Thus, an important part of the analysis is integration and discussion with systems that intended to use the system. For example, if SLB (Swedish Army C^3I System) want to apply role-based access control than this affects the mobile ad hoc node. A possible continuation of this report is to perform the security analysis (briefly described above) and thus derive the security requirements.

1.1 Mobile ad hoc networks

In tactical operations, there may be situations where radio network units, or nodes, move in terrain where line-of-sight communication rarely is possible between all nodes and where pre-deployed infrastructure cannot be guaranteed. Mobile ad hoc networks have attractive properties for scenarios of this kind.

One method to connect network nodes is to relay messages through one or several intermediate nodes. This requires a *routing method* which for each session or message determines the route the traffic will follow from source to destination. Networks with this property are called *multi-hop networks*.

A *mobile ad hoc network* is a system of wireless mobile nodes with routing capabilities, any group of them capable of forming an autonomous network that requires no infrastructure and is capable of organizing itself into arbitrary changeable topologies. A mobile ad hoc network has three important properties: *the multi-hop, self-organizing and autonomous property*. By autonomous network, we mean a network that is not dependent on any fixed infrastructure, such as centralized base-stations. The autonomous property can be achieved either by using distributed algorithms in the ad hoc network or by allowing any group of nodes to automatically agree on one of them to take care of some centralized functionality when needed. For large networks, allowing the nodes to merge dynamically into hierarchies might improve traffic capacity. The ad hoc structure is well suited for building self-forming, and self-maintaining networks that are fast deployable in many environments.

1.2 Security issues in focus

Based on the experience of the authors, the areas listed below have been identified as central to the security of mobile ad hoc networks. Consequently, this report review standards, drafts, and research proposals within these areas.

- Authentication and integrity (communication security)
- Key management
- Secure routing
- Distributed firewalls
- Location/identity separation
- Intrusion detection and response
- Access control

The report also discusses the following areas briefly:

- Code signing
- User authentication
- Storage encryption

- Confidentiality of data
- Management

1.3 Assumptions and limitations

This section outlines the assumptions and limitations that were made in this study.

- This study is focusing on technical problems whereas administrative security problems, such as manual rekeying, are not discussed.
- Security issues related to jamming are not discussed. The study is focusing on security proposals that protect mobile ad hoc networks from different types of IT-based attacks.
- Security solutions for applications are not discussed.
- The study is focusing on reviewing security proposals developed for mobile ad hoc networks.
- The adversary is assumed to have high competence and possess relatively large resources regarding e.g. computer capacity. That is, the adversary is an organization.

Security problems for mobile ad hoc networks are discussed in [1]. In order to delimit the security problems the study is focusing on security issues related to the following areas, identified in [1]:

(1) Data link layer: passive and active attacks

(2) Unauthorized node is added to the network

(3) Network layer: routing threats and packet forwarding threats

(4) Need to identify a host uniquely

(5) Distributed algorithms

(6) Open architecture

(7) Bad physical protection

(8) Internal attacks

(9) Data (e.g. logs of events) transmitted over network.

The security issues related to the following areas, identified in [1], are out of the scope for this study:

(1) Network initializing

(2) Ad hoc network connects to other network

(3) Viruses, worms and Trojans

(4) Security problems related to the Operating System (OS), e.g. to configure the OS in order to minimize security holes.

(5) Several confidentiality levels in one network

(6) Heterogeneous networks

(7) Quality of Service (QoS) and IT security

1.4 Report Layout

The structure of the report is as follows. Section 2 and 3, respectively, provides the black-box and internal white-box views of security solutions for mobile ad hoc networks. Section 4 reviews security proposals within the area of authentication and integrity. Section 5 reviews key management proposals for mobile ad hoc networks. Section 6 reviews proposals for secure routing. Section 7 review proposals within the area of distributed firewalls. Section 8 review proposals within the area of identity and location separation. Section 9 review proposals regarding intrusion detection and responses. Section 10 describes proposals in the area of access control. Finally, section 11 discusses user authentication, code signing, confidentiality of communication, storage confidentiality, and management.

2 Context (black-box)

This section describes the services produced by the security system and dependencies to other systems.

2.1 Produced services

The services produced by the *security system for a tactical mobile ad hoc network* aim to protect the system from different types of IT attacks. Different types of attacks are listed below along with the corresponding security services. The aim of this document is to discuss solutions for each listed security service.

Type of threats	**Security service**	**example of solutions**
jamming	spread spectrum method	This is not discussed, see assumptions and limitation in section 1.3.
eavesdropping of data	encryption at data link layer, network layer, transport layer or application layer.	See discussion in 3.1.
eavesdropping of header data	encryption at data link layer	See discussion in 3.1.
An intermediate node can eavesdrop.	encryption at application layer, transport layer or network layer	See discussion in 3.1.
Traffic analysis	Encryption of header information partly protects against traffic analysis but as far as we know no solution protects against all types of traffic analysis.	No solutions available
Active attack against data link layer (1): One example is an evil node sending many packets in order to consume resources (computer capacity, memory, battery, bandwidth) at the target node	fast lightweight authentication which is design to quickly exclude external packets.	No solutions available
Active attack against data link layer (2): One example is an evil node sending false MAC information in order to disrupt part of network (see further description in section 5.1 in [1])	intrusion detection system.	No solutions available
Unauthorized node added to the network e.g. evil node sends packets that is accepted and forwarded by other node	Spread spectrum method (if good enough) or encryption at data link layer or (authentication at data link layer) or (end-to-end security+firewall)	See discussion in section 3.1 and conclusion in section 4.
An intermediate node can modify a message/packet. A node impersonate another node.	Authentication and integrity protection at application layer, transport layer or network layer	See section 4.
Routing attacks	Routing security	See discussion in section 3.1 and section 6.
A mobile node has *bad physical protection* . A mobile node can be stolen or hijacked. The possibility of compromised malicious nodes performing internal attacks is an important threat. Someone can also manage to infiltrate the system by exploring software or design errors. Different types of ways to be able to perform an internal attack is described in section 3.1, see paragraph 11.	(1) Intrusion detection systems or (2) Firewalls partly protects against some internal attacks, but it depends on how the firewall is configured and which type of firewall it is. (3) authentication and integrity protects against some internal attacks.	Authentication and integrity, see section 4. Intrusion detection systems, see section 9. Distributed firewall, see section 7.
modify system security information in node (firewall rules, security policy, logs of events and incidents)	Storage confidentiality	See section 11.4.
A non-authorized person has access to the node. For example, anyone should not be able to insert new (potentially evil) software in the node.	Access control and user authentication	See section 10 and 11.1.
A piece of code that is put on a node should come from a trusted party.	Code signing	See section 11.2.

Figure 1: IT attacks and corresponding security services.

Security services may also be needed due to security needs, see explanation in figure below.

Security needs	Security service	example of solutions
Cryptography is dependent on key management	Key management	See solutions in section 5.
Incident management, supervision of network, security policy management (authorization information)	Security management	We have not investigated different solutions due to lack of time, see discussion in section 11.5.
A node may need to have a secure identity. One example when a secure identity is needed is when a node is going to send logs of event and incidents to another node. If this is needed depends on how the system is going to be used.	Secure identity of node.	See section 4.
Remote management via wireless or wired interface requires authentication and integrity protection in order to protect against attacks such as impersonation, modification and injection of malicious code.	Authentication, integrity and confidentiality on network layer, transport layer or application layer.	See section 4.
A mobile node may have several interfaces e.g. wireless interface, wired interface and direct interface. All interfaes needs approriate security service, since security is never better than the weakest link.	The needed security service really depend on how the system is used and the risk one is willing to take. Firewall and (authentication and integrity protection) on wired interface.	Solutions for wired interfaces exists. This is not discussed, since this report aims at discussion security solutions developed for wireless and mobile ad hoc networks.

Figure 2: Security needs and corresponding security services.

2.2 Dependencies

The set of security services needed is not exclusively affected by possible attacks and identified security needs but also by dependencies with other components, nodes, and systems. Dependencies may be divided into external and internal dependencies.

1. External dependencies: The security of mobile ad hoc nodes is dependent on systems/applications that use the ad hoc nodes, e.g. Swedish army C^3I systems (SLB). The security of these systems may affect the ad hoc nodes. These systems may also have security requirements on the mobile ad hoc node.

2. Internal dependencies: Functions, protocols and applications in the ad hoc node will affect the appropriate security requirements of the node. For example, if the nodes are supervised via wireless interface, then the corresponding security requirements need to be addressed in order to secure this function.

However, it is outside the scope of this text to identify the security requirements resulting from these dependencies.

3 Internals (white-box)

One single security solutions can not protect against all kinds of different attacks against a mobile ad hoc network. Thus, a goal for ad hoc networks is to apply a multi-defense security solution that offers multiple lines of defense against many different attacks. The solution relies on multiple defenses, spanning different devices and different layers in the protocol stack. For mobile ad hoc networks, this means multi-layer security, cross-layer security (e.g. firewall and IDS), and non-layer security (e.g. access control and storage confidentiality). Note that the security in depth principle is also recommended by FMLS2010, se section 3.2.6 in [30].

Important general security principles are;

1. Security in depth

2. Security design should be considerable as an indispensable part of the systems development process, not as mechanisms that can be added afterwards.

3. Security can be considered as a chain and it is only as good as the weakest link.

   This means that the security architecture for the product mobile ad hoc networks should be derived from a careful analysis that is based on facts, such as the vulnerabilities of protocols, applications, and how the system is used. The analysis lists all the important threats, consequences of realized threats, and the likelihood that someone actually will use a particular vulnerability. Based on these results, risk management resulting in an adequate *level of security* can be performed. That is, a threat with a serious consequence that has a potential large probability that someone actually is using it must be alleviated with a security solution.

4. Balance between security strength and network performance

   Security strength and network performance are both important. When more security features are included in the system, the security is enhanced at the cost of resource consumption (bandwidth, computer computation and memory) and decreased flexibility.

3.1 Logical structure view

In this report, we have derived a security architecture from the threat analysis performed in [1]. The security architecture is depicted in figure 3. This security architecture can be regarded as the first contribution in the process to design a security architecture for tactical mobile ad hoc networks. However, the proposed architecture has only considered general threats against a mobile ad hoc network. The final security architecture should also consider more details about the specified system such as functional requirements, how the system is used, as well as the security of applications. It is also necessary to discuss the acceptable level of risk with TSA/MUST/FMV. That is, it is not possible to protect against all types of threats. Thus, it is important to identify what threats we want to protect the system from and what risks we are willing to take.

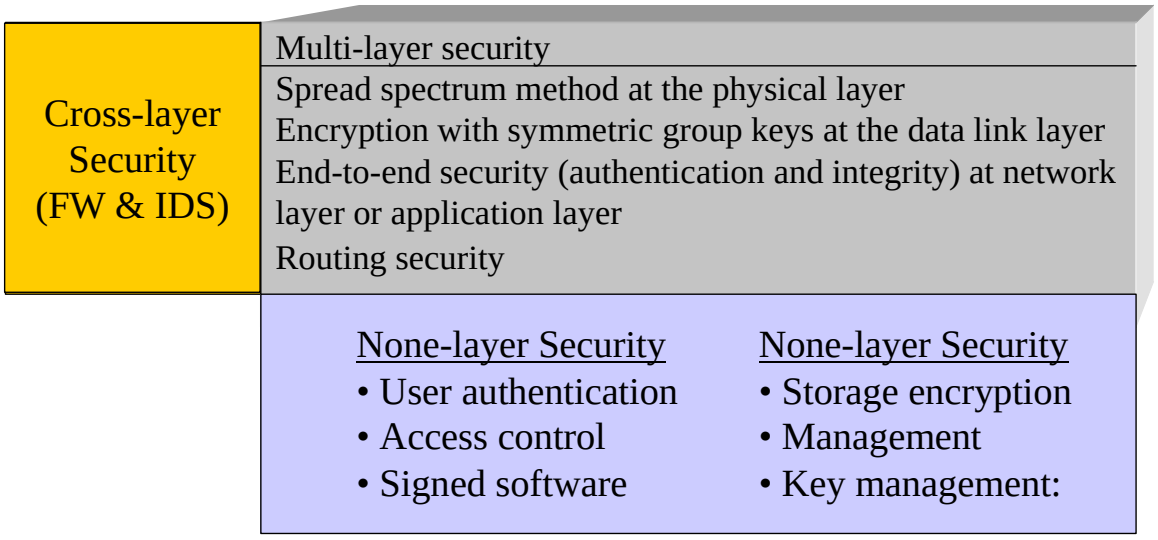


Figure 3: Concept architecture

We can subdivide the security solutions for mobile ad hoc network into;

1. **Physical security: spread spectrum methods**

   The spread spectrum method provides protection against jamming. The spread spectrum method can also provide group authentication, but this depends on the chosen method. If frequency hopping is used without encryption at the data link layer, it is theoretically possible to insert (modified) malicious packets. Thus, frequency hopping by itself does not provide group authentication. Neither does some types of direct sequencing provide group authentication.

2. **Data link layer security: Encryption at data link layer with group keys**

   Encryption at data link layer provides *group authentication* and *protection against eavesdropping of data* and *protection against eavesdropping of headers* (traffic analysis). The main drawback with encryption of data link layer is that each node encrypts and decrypts the message. Thus, an intermediate node can modify and eavesdrop on traffic. This also consumes resources. Another drawback is that automatic distribution of symmetric keys is still a topic that is not solved for ad hoc networks, since the only available solution is manual distribution of group keys. The main advantage of encryption of data link layer is that it provides a high level of security, since all traffic over all communication links is secured. Symmetric-key ciphers can also be designed to have high rates of data throughput. It may not be necessary to implement encryption at data link layer if the following three conditions are met: (1) the spread spectrum method provides group authentication, (2) it is acceptable that an adversary perform traffic analysis, and (3) higher level protocols are protected with encryption.

3. Confidentiality, authentication and data integrity at network layer, transport layer or application layer

Authentication provides protection against masquerading, i.e., a node/user impersonates another node/user. Integrity provides protection against modification by intermediate nodes. Confidentiality provides protection against eavesdropping. *Confidentiality, authentication and integrity protection is needed in a mobile ad hoc network*, but it can be discussed at which layer it should be implemented.

Network layer security provides end-host-to-end-host security while application and transport layer security provides end-user-to-end-user authentication. Data link layer security provides link-to-link security. Therefore, to achieve a high level of security, security mechanisms can be put in the data link layer, network layer, and application or transport layer. Normally, it is too resource demanding to put security in all three layers.

The main advantage of application layer security is that the message/connection is protected the entire way up to the application, i.e. true end-user-to-end-user security. However, application layer security only protects the applications, but it cannot protect the network (routing attacks, resource consumption and other attacks against network layer protocols). A rule is that security mechanisms applied in a specific layer of the OSI stack cannot protect against successful attacks to or vulnerabilities in lower layers, i.e., application layer security cannot protect against vulnerabilities or successful penetrations of the network layer.

The advantage of network layer security is that it provides host-to-host security. Thus, it protects against some network layer attacks but also gives a certain protection to the application. For example, an intermediate node can not modify a message without being revealed. The drawback of network layer security is that it does not provide user-to-user authentication.

Whether to apply confidentiality, authentication and integrity in the network layer, transport layer, or application layer depends on the applications and how the system is used as well as what risks we are willing to take. For example, if the following conditions are met then network layer security may not be needed: (1) group authentication is provided, (2) we do not want to protect the network from internal attacks, and (3) applications are responsible for their own security and are aware of the fact that the network is not trusted. On the other hand, network layer security may be needed if the following conditions are met: (1) we want to protect against internal attacks against the network, (2) we do not know how the system is going to be used, or (3) we do not know the security needs of the applications.

Note that application layer security may be needed for both applications outside the node (which uses the ad hoc node) and applications inside the node (e.g. management). For example, if an ad hoc-node is going to send logs of event to a central node than the node need a secure identity as well as application level security between the node and the server. Another example is management. Remote management via wireless or wired interface of the ad hoc node requires confidentiality, authentication, and integrity protection between the node and the management application.

4. Key management

All key oriented functions such as authentication and encryption require key management of some kind. Any cryptographic mechanism is insecure if the key management is weak.

Key management for mobile ad hoc networks is not an easy task. Currently, manually pre-created distributed keys are the only method that can be applied to mobile ad hoc networks without further investigation. TAK2 can be used for both symmetric and asymmetric keys. However, manually distributed keys have a large number of drawbacks. Relevant problems are lack of real-time revocation and distribution of keys, weak connection to the authorization system, traceability, and misuse of keys [29]. Fast (real-time) revocation of keys is especially important in mobile ad hoc networks, since a realistic threat against mobile ad hoc networks is compromised nodes. Manual key management also means that the keys are used for a relatively long time, which gives an adversary long time to crack the key. In the long term, pre-created distributed keys can be used for initialization, but one should search for a key management method that provides a real-time service. Section 5 discusses methods that are relevant for further discussion, but no solution can be used without a lot of further development.

It would be possible to use asymmetric keys saved on active cards (TAK2) to distribute symmetric session keys. For example, the TLS standard and IPsec key management protocol (ISAKMP/OAKLE) can be configured to create session (and security association) keys. This standard may not be possible to use, but the method they rely on can be used. However, these methods rely on public key encryption which has the highest classification level of restricted. Another solution is to use unique symmetric keys saved on the active card. One node can transmit the new session key in a secure way by encrypting the new key with the old symmetric one. However, if symmetric unique keys are used a total number of $n*(n-1)/2$ keys must be maintained in a network with n nodes. This may only be usable if the number of nodes in the network is small.

In order to be able to enforce the solution in paragraph 2 and paragraph 3 the TAK2 card has to include a group symmetric key and an asymmetric key pair.

5. Cross-layer security: distributed firewalls and intrusion detection system

Cross-layer security can be divided in distributed firewalls and intrusion detection systems. Currently, there are no intrusion detection tools for mobile ad hoc networks, see conclusion in section 9.A distributed firewall can for example provide the following services:

1. Packet filtering (filter out certain MAC addresses and IP addresses).
2. Rejection of connection requests for inappropriate services (filtering of ports and application proxy).
3. Protection against some DoS attacks and protect internal LAN.
4. Restriction of outgoing and ingoing traffic

A distributed firewall is a step in the right direction to enforce network based filtering. If the node is equipped with both encryption and authentication, the main advantage of the firewall

is to make it more difficult to perform internal attacks. The distributed firewall is also a multi-defense. It is possible to implement rules on both the wired and wireless interface.

A drawback of distributed firewalls is that these are centrally managed. A possible solution is to update the firewall rules when accessible. However, this means that it may take a long time to enforce a new important firewall rule. Another issue is that products depend on the operating system.

6. Cross-layer security: intrusion detection systems and response

An Intrusion detection system (IDS) is a system that automates supervision of events by analyzing collected data to make conclusions about ongoing IT attacks against the network or the host. A necessary condition for efficient detection is that intrusion activities have distinct behaviors that are observable. In other words, an IDS detects intrusions by analyzing data collected from the network or the host and also try to prevent such activities that may compromise the system security.

The conclusion of the review of the current intrusion detection tools is that there are no products available for mobile ad hoc networks. The conclusion of the review of the research in the area of intrusion detection for mobile ad hoc networks is that the methods really need to be refined before they can be used in military systems.

Thus, an area that needs to be further discussed is supervision of the network and incident management.

7. User authentication and access control

Authentication is a validation of a user´s identity against previously stored information. This identity can be used by the access control system to ensure that the objects of the system are used by the right person. Note that even if you have user authentication you still need access control. User authentication methods developed for wired networks can be used in mobile nodes.

The purpose of access control is to control which subjects (processes, persons, machines) have access to which objects in the system. In other words which system resources they can use (read, delete, modify). There are methods available for wired networks, but there are some questions regarding the support of access control by the operating system, revocation of certificates and management, see more information in section 11.

8. Security management

Management is needed to enforce security policy management, e.g. distribution of authorization information and user account information, configuration of nodes, supervision of nodes and incident management (e.g. discovers intrusions and manages security weaknesses and responds to intrusions or threats). Security management for mobile ad hoc networks is an area that needs further investigation. In this report, no methods have been investigated due to lack of time.

9. Code signing

The purpose of code signing is to make sure that a piece of software (e.g. a new version of a routing protocol) comes from a trusted party and that it has not been changed since it was

signed. Several solutions are available. However, the solutions are often developed for a certain operating system, which means that it might become necessary to port an existing solution.

10. Storage confidentiality

Mobile nodes need to save system security information in the node, e.g. user account information, firewall rules, keys, and authorization information. The node will also include other sensitive information such as logs of events and incidents. This information need to be protected from unauthorized disclosure.

Methods for wired networks are useful. The solutions are often developed for a certain operating system, which means that it might become necessary to port an existing solution.

11. Protection against internal attacks

The possibility of compromised malicious nodes performing internal attacks is a severe threat against a mobile ad hoc network. Internal attacks exist in wired networks even though most companies do not admit that they have problems with internal attacks. In fact, attacks with serious consequences are often internal attacks. However, the mobile ad hoc network is more vulnerable to internal attacks compared to a wired network, since many protocols developed for mobile ad hoc network are based on *distributed algorithms*. These distributed algorithms open the way for internal attacks, since the algorithms are based on the cooperative participation of nodes. If one node is malicious, it can affect the entire network. It may be possible to perform an internal attack if the adversary succeeds in one of the following attacks:

- Steal or hijack a node.

  A mobile node has bad physical protection. In some military scenarios, it may be very difficult to steal a radio. However, in an international scenario where the military units are under deployment or have spread deployment, it may be easier to steal a mobile node.

- Infiltrate the system by exploring software or design errors.

  It is difficult to design and implement software systems without introducing design and programming errors that an adversary can exploit. If an adversary has adequate resources and tries hard enough, there is a risk that the adversary succeeds in infiltrating the system. History has taught us that no matter how many intrusion prevention mechanisms (e.g. encryption, authentication and firewalls) are inserted in the network, there is always some weak link that an adversary can exploit. For example, even though buffer overflow has been a known security problem for many years, there is still recently released software with buffer overflow security holes. If the buffer overflow security hole is exploited, it may lead to an unauthorized root shell. In order words, someone can infiltrate the system. The risk is reduced by multi-layer security.

- Insert malicious data in a node by sending malicious software to a node which is first connected to the Internet via security mechanisms (a firewall) and later connected to one radio node via the Ethernet network.

As laptops are used today, it is not unlikely that a laptop is used in several networks. Thus, malicious software could spread from one network to another.

- Bribe (or threat) an authorized user to insert the malicious software.

The risk of a successful internal attack is reduced by multi-layer security; firewall, intrusion detection system and end-to-end security (authentication, confidentiality and integrity). Firewalls partly protects against internal attacks, since it is possible to restrict both ingoing and outgoing traffic. End-to-end security partly protects against internal attacks since it protects against masquerading and modification of messages. The aim of intrusion detection system is to detect the remaining internal attacks, but today there are no products for mobile ad hoc networks.

In the following sections (section 4 - 11), different types of security solutions are discussed.

4 Authentication and integrity (communication security)

In this section, we discuss authentication methods and protocols for providing authentication between two nodes. We describe advantages and drawbacks for each protocol.

What is authentication and integrity protection?

Authentication assures the recipient that the message is received from the claimed source. In other words, authentication is a validation of a node's identity against previously stored information. Thus, a secure identity is needed. The *integrity* component of the cipher security protects against message modifications.

There are three cryptographic mechanisms to authenticate the content of messages exchanged among nodes: symmetric solutions, hybrid solutions, and asymmetric solutions. The major authentication mechanisms used in commercial wired systems are passwords, Kerberos and asymmetric solutions.

The cryptographic mechanisms can be put in one or several layers in the protocol stack. Today, the most common way in tactical radio networks to protect the communication is data link layer encryption, but it is also possible to protect the communication with network layer security. Advantages and disadvantages of data link layer security are discussed in section 4.1. Advantages and disadvantages of network layer security are discussed in section 4.2.2.

Secure identity

Authentication is a validation of a user's or node's identity against previously stored information. Thus, a secure identity is needed to be able to perform authentication. The following is a list of commonly used identities [28]:

- The identity of a device or user can be the private key in an asymmetric key-pair
    - The keys can be stored in an active card (also called smart card)
    - The keys can be stored on a soft certificate (e.g. PKCS#12, PEM)
- The identity can come from a key server
- The identity can come from a Kerberos server
- The identity can also be a password (weak authentication)

SwaF has developed three main types of active cards [28]:

1. NBK is a card for symmetric keys. An online key server is used for providing accurate information to the users.
2. TEID is a card used mainly with COTS.
3. TAK2 is a card that can be used for both symmetric keys and asymmetric keys (up to 2048 bits).

It is preferable that the solution can be based on an existing active card. TAK2 is the most relevant card for mobile ad hoc networks.

Why do we need authentication and integrity protection?

One of the most important threats to a mobile ad hoc network is that an unauthorized node is added to the network. An unauthorized node can launch several serious attacks such as eavesdropping, block traffic and disrupt the entire network with routing attacks [1]. Another category of attacks is masquerade, where one entity pretends to be a different entity. A user can masquerade as another user. A node can also masquerade as another node. Another group of serious attacks against a mobile ad hoc network is Denial of Service (DoS) attacks, which prevents the normal use of communication facility and thus affecting the availability. One example of a DoS attack is to send many packets in order to deny legitimate users from using the wireless channel. Thus, efficient methods are needed to exclude external packets. These threats imply that *authentication is needed between the mobile ad hoc nodes.*

Authentication may also be needed for secure transfer of information (e.g. detected attacks and event lists) from the ad hoc node to a centralized server when accessible. Furthermore, it may also be needed to transfer security information (e.g. firewall rules, IDS rules, authorization information, revocation lists) from a centralized server to ad hoc nodes in a secure way.

4.1 Solutions in current tactical radio networks

Today, it is common to apply confidentiality at the data link layer. Usually symmetric group keys are used. Encryption at the data link layer with use of group keys provides some kind of group authentication. When the sender and recipient share a secret key, then only the genuine sender would be able to encrypt a message that the recipient can decrypt and read. The authentication is done by the recipient, by verifying that the message is readable. However, the node cannot automatically dismiss a non-readable message at the data link layer. Neither can encryption by itself protect against active attacks at the data link layer such as congestion, masquerade, deceptive messages replay of packets, or triggering extensive power consumption. Therefore, cipher security should include authentication, integrity and replay protection, as well as confidentiality.

Advantages and disadvantages of data link layer security (assuming group keys):

- All traffic over all communication links is secured, which provides a high level of security. For example, it will exclude external packets transmitted from an adversary and also protects against eavesdropping

- The entire packet including header and data is usually encrypted. This partly protects against traffic analysis. However, an adversary can still notify nodes sending more packets than others do.

- The message must be encrypted and verified each time it enters a node. Thus, the message is vulnerable at each node. Each node can modify and eavesdrops on all traffic. Encryption and decryption at each node also consumes resources.

- Impersonation of another node is possible if group keys are used.

4.2 Standards and popular solutions developed for wired and wireless networks

4.2.1 RAIDUS and DIAMETER

The Remote Authentication dial in User Service (RAIDUS) is a popular protocol for authentication and authorization between a Server and a client [38]. There are several security weaknesses in RAIDUS such as one-way authentication and no replay protection. Therefore, the Diameter Base Protocol, also known as RAIDUS2, was developed to solve the security problems of RAIDUS [39]. Both solutions are based on centralized architectures.

4.2.2 IPsec

The IP security protocol (IPsec) provides security services at the network layer. IPsec is an integral part of IPv6, but can also be applied on IPv4. There are few differences between IPsec with IPv4 and IPv6 [53]. IPsec can be applied in the network in three different ways: host-to-host, network-to-network, and host-to-network protection. The most common configuration is perhaps network-to-network protection, which provides protection between two networks. However, the only configuration applicable to mobile ad hoc networks is host-to-host protection.

IPsec is described in a number of RFCs. The most important RFCs are;

- RFC 2401: An overview of a security architecture
- RFC 2402: Description of a packet authentication extension to IPv4 and IPv6
- RFC 2406: Description of a packet encryption extension to IPv4 and IPv6
- RFC 2408: Specification of key management capabilities

There are currently two protocols that can be used to provide security; Authentication Header (AH) and Encapsulation Security Payload (ESP). The services are as follows; connectionless integrity, data origin authentication, rejection of replayed packets, confidentiality and limited traffic flow confidentiality.

	Authentication Header	Encapsulation Security payload
Connectionless integrity	**x**	
Data origin authentication	**x**	
Rejection of replayed packets	**x**	
Confidentiality		**x**

Figure 4: IPsec services

A key concept in IPsec is Security Association (SA). SA is a one-way relationship between a sender and a receiver. A recipient and a sender must agree on security services to the traffic carried between them such as keys, authentication algorithms and (ESP or AH) protocol.

Advantages and disadvantages

IPsec provides confidentiality, integrity, authentication, and protection against replay attacks at the network layer. It is possible to include military-grade encryption algorithms.

One important drawback is that IPsec cannot be readily used for traffic other than unicast. For multicast, IPsec can protect the traffic at the cost of efficiency, which may cause many duplicates.

IPsec provides network layer security. Advantages and disadvantages of network layer security are;

- The encryption and decryption is carried out at the two end systems. That is, the source node encrypts/signs the data and the destination node decrypts/verifies the data. Thus, intermediate nodes can not eavesdrop on the traffic or modify traffic without disclose.

- The traffic pattern is not protected since the header must be open to be able to route the packet.

- The idea of end-to-end encryption/authentication is that intermediate nodes do not decrypt/verify the packets. In other words, intermediate nodes will not examine and dismiss an evil packet from an adversary that is intended for another node. The end-node (addressed node) will dismiss an evil packet from an adversary node. *Thus, end-to-end encryption/authentication can not provide node authentication.* That is, it can not protect against an adversary node sending bogus packet, which consumes resources. For example, AH in transport mode is viewed as an end-to-end payload, which is not examined or processed by intermediate nodes. However, it is possible to configure a firewall to only allow authorized IPsec traffic and filter external packets from an ad hoc network.

There are some known security problems in IPsec. One problem is that IPsec is very complex, which may result in incompatibilities, bugs, misinterpretations and faulty configurations. IPsec also allow many different configurations and some of them include known security holes. For example, IKE aggressive mode (key exchange protocol) is vulnerable to man in the middle attacks on the public key exchange. Therefore, when using IPsec, the consequences must be carefully weighted when making a decision on how to configure and implement IPsec. Furthermore, IPsec itself is also vulnerable to denial of service attacks. Moreover, although some anti-replay protection is included, IPsec does not preclude replay attacks, IPsec just makes it harder to execute them successfully. Security problems in IPsec are described in more detail in [53].

4.2.3 802.11 security (WEP, WPA and 802.11i)

The first attempt to provide security in the IEEE 802.11 standard was based on the Wired Equivalent privacy (*WEP)* standard. The WEP standard has however proved inadequate for securing wireless networks. Many security experts have identified holes in the underlying WEP specification. WEP has been criticized for weak encryption, no key management, too short encryption key, too short initialization vector and static encryption keys. For a large busy network, a hacker with the proper equipment and tools can collect and analyse enough data to recover the shared encrypted key within 15 minutes.

In order to address 802.11 security issues, IEEE has together with the industry, the so-called Wi-Fi Alliance, developed the Wi-Fi Protected Access (*WPA*) protocol [31].

WPA implements three improvements compared to WEP;

1. 802.1x EAP.

802.1x EAP provides mutual authentication and a method for distributing encryption keys dynamically [32].

2. Stronger encryption.
   WPA applies Temporal Key Integrity Protocol (TKIP) to address WEP´s known vulnerability in the area of data encryption [33]. TKIP uses a 128 bit temporal key and a 48-bit initialization vector. TKIP combines the temporal 128 bit key with the node's MAC address and then adds the 48-bit initialization vector. TKIP also provides a dynamic distributing method, which changes temporal key every 10000 packets. Unfortunately, TKIP is based on the same encryption algorithm as WEP. Note that the problem is not the RC4 algorithm, but the way the RC4 algorithm is used.

3. Stronger Integrity Validation
   WPA uses the Michael Message Integrity Check to enforce stronger Integrity Validation [7].

WPA improves the level of security compared with WEP, but there are still potential encryption weaknesses in WPA mainly due to the usage of RC4. WPA can be considered as an interim solution, which is not appropriate in military contexts.

IEEE has also formed a task group to develop the *802.11i standard*. The draft standard of 802.11i was released on 24 June 2004. The architecture contains the 802.1X for authentication, integrity and origin authentication, RSN for keeping track of association and CCMP based on Advanced Encryption Standard (AES) protocol to provide confidentiality. An important part of the authentication process is a four-way handshake between the access point and the client.

This new protocol has enhanced the security features for wireless LANs fundamentally. For example, IEEE 802.11i uses the AES protocol, which is an improvement compared to WEP/WPA based on RC4. However, the 802.11i is based on centralized servers and can thus not be used in a mobile ad hoc network.

4.3 Solution proposed by LedsystT

In the LedsystT document *Security architecture overview*, it is described that *"all access to infrastructure, services and systems shall be authorized"*, see section 3.3.6 in [30]. For communication security, they discuss *in general* the use of authentication and integrity protection with digital signatures, message authentication codes and keyed hash values as well as different encryption methods. No specific method for node authentication in mobile or autonomous networks is recommended. However, they propose to use the P2P data distribution concept as a solution to distribute security information to mobile nodes. The idea is to handle the information on a distributed overlay network. This means that some type of P2P authentication method is needed, but they do not describe or recommend any P2P authentication method. In peer-to-peer authentication, authentication is no longer the duty of a server, but the responsibility of both end nodes.

4.4 Authentication for mobile ad hoc networks

Authentication and establishment of keys among nodes are important security issues in mobile ad hoc networks. The authentication phase of a node can be divided into two phases. The first phase, named initialization, is executed to exchange the data that is required in later authentication between the nodes. In symmetric cryptography, the initialization phase refers to an authentic and confidential channel for exchanging symmetric key data. In asymmetric cryptography, the authentic and confidential channel is used for exchanging public keys of other nodes and for downloading the node's own private key. In military networks, the initialization phase can be done with manual key distribution. The second phase, the authentication phase, is the actual authentication over an insecure channel using the authentic data that was exchanged in the initialization phase. This section discusses the second authentication phase.

Besides, symmetric and asymmetric methods, there are two types of authentication that we may need in our mobile ad hoc network: node authentication and broadcast authentication. In broadcast communication, each packet is sent from a source to a number of receivers whereas node authentication provides online authentication between two devices. In node authentication, after the nodes have successfully authenticating each other, the nodes may establish a session key that is used to encrypt all further communication.

4.4.1 Symmetric solutions

The main principle of symmetric encryption is that only a single key is used for encryption and decryption. One common symmetric authentication approach is to generate an authentication tag with a Message Authentication Code (MAC) algorithm which is appended to each message for authentication. A MAC is an authentication tag derived by applying an authentication scheme, together with a secret key, to a message. Unlike digital signatures, MACs are computed and verified with the same key. There are several types of MACs. The one hash function-based MAC (HMAC) is described below.

HMAC

HMAC use a secret key in conjunction with a hash function to produce the authentication tag. A cryptographic one-way hash function is appended to a block of data of any size. The output is a fixed-length output.

This technique is efficient and affordable for low-end devices. However, there is an issue with scalability. If pair-wise shared keys are used, a total number of $n*(n-1)/2$ keys must be maintained in a network with n nodes. Another problem is that HMAC only can be verified by the intended receiver, making it unsuitable for broadcast message authentication.

Asymmetric MAC Broadcast authentication

MAC provides two-way entity authentication whereas digital signatures also provide broadcast authentication. However, there is a proposal by Canetti et al that provides broadcast authentication with symmetric MAC [54]. In their proposal, each entity gets a subset of keys at initialization time. The main set and the subset are chosen so that the probability that each two subset have at least one key in common is high. Their proposal provides broadcast authentication but also generate a lot of message overhead for key-agreement.

Kerberos assistant authentication for Mobile ad hoc networks

In Kerberos, the server is usually a single point of failure and really not intended for a dynamic and mobile network. Therefore, Pirzada and McDonald have introduced certain changes to the original protocol to make it more suitable for ad hoc networks [55]. In their proposal, they have multiple Kerberos servers for distributed authentication and load distribution. The servers periodically replicate their databases with each other. When a node N1 want to communicate with another node N2 it sends a request to one of the Kerberos servers. The server creates a ticket, which it sends back to node N1. Node N1 sends the ticket to node N2, which acknowledges the ticket. Then a secure session is established between node N1 and N2. Thus, an obvious issue is the communication overhead created. This is probably not acceptable for short messages that are common in tactical communication. Another issue, which is not discussed in the proposal, is the availability of servers and the number of servers.

4.4.2 Hybrid solution

Many solutions combine symmetric and asymmetric cryptography to provide efficient authentication. A common approach in wired networks is to first use asymmetric solutions for authentication and key exchange of the session keys and then symmetric keys for encryption.

Mixed authentication solutions

In order to get the best of both asymmetric and symmetric solutions mixed authentication solutions can be used. For example, an asymmetric key agreed scheme such as Diffie-Hellman can be used followed by a symmetric authentication method such as MAC for each authentication. This solution is both scalable and effective, but there are no standardized protocols, implementations, or detailed specifications that are based on this solution. Also note that this solution assumes an available Certificate Authority (CA) for revocation of keys (key management problem).

One-way HMAC key chain

Another hybrid method is the one-way HMAC key chain. A chain of outputs can be obtained by repeatedly apply a hash function h() on an initial secret input x. These outputs can be used in the reverse order of generation to authenticate messages. Thus, the outputs from the hash function (x, h(x), h(h(x)), etc) are used as a one-way key. Lamport's one-time password-scheme is based on hash-chains [17]. A node randomly chooses a secret k. A hash function h is used to define a sequence of passwords k, h(k), h(h(k) etc. These passwords can be used for keys. Note that the Lamport method does not provide entity authentication as there is no proof of communication.

TESLA

TESLA is a broadcast authentication protocol based on a hash-chain based protocol [25, 27]. To use TESLA for authentication, each node chose an initial key k and computes a one-way key chain: k_1=h(k), k_2=h(h(k)), k_3=h(h(h(k))) etc. Each node pre-determines a schedule at which it publishes each key of its one-way chain. For example, key k_i is published at time T_i= T_0+t*x. TESLA is efficient, since it adds only a single message authentication code (MAC) to a message for broadcast authentication. The authentication is also lightweight. However, TESLA requires tight clock synchronization. Another drawback is that the receiving node need to buffer a message for verification until the key is revealed. Thus, the verification of the messages is

delayed. There is also a version called microTESLA, which uses a pre-distribution key for the initialization authentication of the final hash-chain key k_0.

LHAP: A lightweight Hop-by-Hop Authentication protocol

Zhu et al have proposed a lightweight protocol, called LHAP, which is optimized with respect to performance [23]. Their method is based on two techniques; (1) hop-by-hop authentication for verifying the authenticity of all packets and (2) one-way key chain and TESLA for packet authentication. The latter is also used for reducing the overhead and establishing trust among nodes. This means that intermediate nodes authenticates all the packets they receive before forwarding them. When a node first joins the network, it computes its one-way key chain and TESLA key chain. These keys are signed and encrypted and then broadcasted to its neighbors. After this phase, named trust management phase, the node can start to communicate with the other nodes in the network. It then uses a lightweight protocol for traffic authentication that is similar to TESLA. One advantage, compared with TESLA is that their authentication technique does not use periodic and delayed key disclose. However, the drawback is that this scheme does not achieve the same level of security as TESLA [23]. This method is interesting, since it provides node and broadcast authentication with a lightweight technique. However, the method assumes tight clock synchronization. Another issue is that the authors have not described how a node knows about its neighbors or any details about the key exchange.

4.4.3 Asymmetric solutions

In asymmetric solutions, each node has a certificate issued by a certificate authority (CA) and an assigned public/private key pair. The private key is always linked mathematically to the public key. Asymmetric solutions can be used for both message authentication and entity (node) authentication.

Digital signature

A digital signature is similar to HMAC. As with HMAC, the digital signature uses a hash function that accepts a variable-size message as input and produces a fixed-size message digest as output. Unlike the HMAC, the hash function does not take a secret key as input. The message is instead encrypted with the private key to generate a digital signature. There are several types of algorithms that provide a signature. The most common ones are probably RSA and elliptic curve digital signature algorithms.

The disadvantage of public-key encryption is the computational overhead of current algorithms. A slow algorithm may cause congestion if an adversary sends many authentication requests. Each node also needs to keep a certificate revocation list of revoked certificates. However, public-key encryptions only needs a total number of n public/private key pairs. Thus, the method is scalable.

Hierarchical authentication architecture

Venkratraman and Agrawal have presented an end-to-end authentication scheme that relies on mutual trust between nodes and is based on a hierarchical architecture [22]. Their method assumed that a node that joins the network is equipped with a system (group) public key and system (group) private key. Their proposal also assumes a hierarchical architecture, where the cluster head node is responsible for distribution of cluster keys to all the cluster members. The cluster key is encrypted with the system public key and broadcasted by the head. When a node

joins a network for the first time strong mutual authentication is performed with use of the system key pair. When a node leaves a cluster and joins another cluster, the new cluster head authenticates it with the system key pair and gives it a new cluster key. When a node in one cluster wants to communicate with a node in another cluster, a session key is created for that communication. However, this approach is usable only if the nodes move in groups and it is possible to identify a cluster head in advanced. There are also several questions that need further investigation, such as redundancies and replacement of cluster heads. Another issue is that the communication between clusters seems to be a complex process that is based on the TCP transport layer protocol.

Location-based security

International Series Research, Inc of Boulder, Colorado has developed a technology called cyberLocation, which provides location-based authentication based on GPS. A location signature is computed from raw observations of all the GPS satellites in view. This provides a unique signature to a particular place and time, since the signals are unique and constantly changing.

This technique may be valuable for military missions, since it complements other authentication mechanisms. However, a problem in mobile ad hoc networks is that the adversaries are also in the vicinity. Thus, the method is not useful in this context. The method is ideal for protecting fixed sites. For example, it could be used to restrict access or sensitive transactions to clients located at those sites.

4.5 Conclusion and discussion

Authentication is the first mechanisms for defense against malicious activity. It is the validation of a node's identity against previously stored information. *The question is; validation against what? Should every node store information about other nodes?*

In mobile ad hoc networks, every node must be a router. One of the natural questions regarding node authentication in mobile ad hoc networks therefore is: *Should every node also be a server for authentication?* Although it is possible to do so, there are some obvious problems. It is not trivial to add a new node to the operating network or to remove a misbehaving node. On every such occasion, the updated information must be propagated to all nodes in the network in a secure and real-time way. Another problem is the size and resources of the mobile nodes, which have to save all information about all other nodes.

A three-stage (or two-stage) protocol may be used;

(1) Initialization

The authentication of a node can be divided into two phases. The first phase, named *initialization*, is executed to exchange the data that is required in later authentication between the nodes. The second phase is the actual authentication over an insecure channel using the authentic data that was exchanged in the initialization phase. The initialization phase can be handled by manual delivery of a shared secret (asymmetric and/or symmetric) on active cards (e.g. TAK2).

(2) Authentication solutions that are relevant for further discussion

a. Encryption at the data link layer + individual authentication

Encryption at the data link layer with use of group keys provides group authentication. The authentication is done by the recipient, by verifying that the message is readable. However, the node cannot automatically dismiss a non-readable message at the data link layer. Neither can encryption by itself protect against active attacks at the data link layer such as congestion, masquerade, deceptive messages replay of packets, or triggering extensive power consumption. Therefore, if this solution is used in a hostile environment, it should be complemented with some type of authentication and integrity protection. Examples of authentication solutions are described below see 2b, 2c and 2d.

b. Individual authentication

Examples of individual authentication solutions are MAC, HMAC, TESLA, LHAP, asymmetric MAC Broadcast authentication and digital signature. The digital signature assumes distribution and revocation of asymmetric keys. The MAC, HMAC, LHAP and asymmetric MAC Broadcast authentication method are dependent on a key management method to distribute symmetric keys. However, distribution of symmetric keys in an ad hoc network is still a topic that is not solved by the secure routing protocols. One solution which solves part of the problem is to use asymmetric keys saved on active cards (TAK2) to distribute unique symmetric keys. However, asymmetric cryptography is based on certificates that need to be constructed, revoked and maintained by a trusted third party. Currently it is not possible to perform this in real-time in a mobile ad hoc network, since this requires an always accessible on-line server. However, it is much better to perform authentication with manual distributed keys than no authentication at all.

c. Hybrid authentication

Many solutions combine symmetric and asymmetric cryptography to provide efficient authentication. For example, an asymmetric key agreed scheme such as Diffie-Hellman can be used followed by a symmetric authentication method such as MAC for each authentication. This solution is both scalable and effective, but there are no standardized protocols, implementations or detailed specifications that are based on this solution. Also note that this solution assumes an available CA for revocation of keys (key management problem).

d. Fast lightweight authentication at the data link layer

Even though authentication is an important security service in mobile ad hoc networks, there are not many papers dealing with authentication for mobile ad hoc networks. One relevant issue is how to handle DoS attacks such as when an attacker feeds a victim node with a large number of packets to exhaust the victim’s computation resources. Some type of fast lightweight authentication at the data link layer is thus required. As far as we know, there are no papers that try to optimize the authentication method in order to be resistant against DoS attacks. This subject can be considered as an own research area. (Note also that these problems may also be handled by intrusion detection systems.)

(3) Session (or security association) key establishment

After the successful authentication, the nodes can start to establish a *session key* in the third protocol stage. That is, when to nodes which to communicate they establish a logical connection. For the duration of that logical connection, all user data are authenticated and/or encrypted with a one-time session (or security association) key. The permanent key (on the active card) is used to send the session key to the other nodes. If only one message is sent between the two nodes and these nodes never are going to communicate again this stage (stage 3) becomes redundant. It could also be a design decision only to use the permanent keys on the active card for encryption and authentication. One advantage of only using the permanent keys is that these keys never leave the active card. On the other hand, a drawback is that the permanent keys will probably be used during a relative long time and maybe for encryption of much data.

For example, the TLS standard and IPsec key management protocol (ISAKMP/OAKLE) can be configured to create session (and security association) keys. However, the TLS standard is performed at the TCP level and IPsec with ISAKMP/OAKLE assumes IPv4 or IPv6. Thus, these protocols is not be usable in a tactical mobile ad hoc network if the protocol is not based on IP, but the method is still possible to implement in another protocol.

To conclude, *authentication and integrity protection is needed in a mobile ad hoc network*, but it can be discussed at which layer it should be implemented. Advantages and disadvantages of data link, network, transport and application layer security is discussed in section 3.1.

The general trend in embedded processor technology is that processors become more and more powerful and has more memory, which in the future enables constrained mobile devices to perform complex computations. Thus, asymmetric schemes that require heavy computations may be used for all authentications (not only initial authentication). However, the limited battery power of mobile nodes may still be an issue. The trend of more and more powerful processors with more memory may also make it possible to use symmetric unique keys without any problems if the number of nodes is limited.

5 Key management

All key oriented functions such as authentication and encryption require a key management of some kind. Any cryptographic mechanism is insecure if the key management is weak. Key management is therefore a central aspect of security. A key management service should provide solutions for the following requirements:

1. **Cryptosystem**: The key management system should provide some kind of cryptosystem; the most common ones are asymmetric or symmetric.

2. **Key creation**: The key management system should be able to create keys.

3. **Key storage**: The key management system should provide storage of keys.

4. **Key distribution**: The key management should ensure that keys are securely distributed. This also includes (periodic) updates of keys.

5. **Revocation of keys**: Key revocation means to stop the key from being used any further due to the lifetime has expired or the key is compromised.

The symmetric key management approach has computation efficiency, but it suffers from attacks against key distribution and scalability. Scalability is a problem, since the number of keys is n*(n-1)/2 keys in a network with n nodes. Key distribution is a problem, since there is a need to transfer the secret key securely and in a tamper free fashion from sender to receiver. Asymmetric cryptography addresses this problem by distributing open public keys or distributes a new session key by encrypting it with the public key.

The asymmetric key management approach has been widely deployed for wired networks due to its simplicity for key distribution, scalability and at the same time providing authentication, integrity and non-repudiation. However, it is still unclear if asymmetric approaches can be extended to ad hoc networks. The asymmetric approach is dependent on effective management of digital certificates, which is achieved with a Public Key Infrastructure (PKI). The most important part of the PKI is the Certificate Authority (CA) that provides certificate issuing, renewal, revocation of certificates and certification directory service. However, providing such infrastructure in mobile ad hoc networks is a challenge due to the infrastructure-less nature of mobile ad hoc networks. In other words, key management is a complex area in ad hoc networking since a *centralised* always available *on-line* key server is not appropriate.

The challenge of designing key management for a mobile ad hoc network is to establish a secure communication infrastructure before any routing fabric has been established and in the absence of any infrastructure or centralized online server. Thus, we will mainly focus on the problem of secure key distribution.

In this section we first describe current standards and solutions developed mainly for wired networks. Then research in the area of key management for mobile ad hoc networks is described.

5.1 Standards and solutions developed for wired and wireless networks

Key distribution can be achieved in a number of ways:

- Manual delivery: the keys are physically deliver (e.g. on an active card) to the node.
- One node delivers the key: if two nodes have previously got a common key, one party can transmit the new key in a secure way by encrypting the new key with the old one.
- Trusted third party deliver: If the nodes have an encrypted connection to a trusted third party (TTP), the TTP can deliver new keys.

5.1.1 Current key management solutions for radio networks

Today, military Swedish key management is mainly based on *manual key distribution* [29]. There are a large number of drawbacks with manual key managements. One important problem is that it takes long time to revoke a compromise key. It is important to bring down lead times in order to deliver and revoke keys in real-time [29]. Revocation of keys in real-time is especially important in mobile ad hoc networks, since nodes in these networks compared with a wired networks have an increased risk of being compromised. Deliver of keys in real-time is also important to perform frequent key changes. Other problems of manual key distribution are weak connection to the authorization system, traceability and misuse of keys [29].

In current tactical radio networks, it is common to use *group key management*. Thus, all nodes share a secret key that is used for encryption and decryption. This is a feasible solution for static groups, but not appropriate for dynamic groups [29]. For example, in mobile ad hoc networks where nodes frequently join and leave the group, a node that joins the group must be given a new key. Otherwise, the node is able to decrypt (earlier) traffic that the node is not authorized to eavesdrop on. Similarly, when a node leaves the group all members in the group must be given a new key. In other words, group key management for dynamic groups involves a problem with forward and backward security. Another problem with group key management is that the key never can be used for authentication [29]. Thus, it is not possible to know who sent a message, since a node can impersonate another node. A third problem is that compromised nodes must be excluded manually, since it is not possible to send new keys to some nodes.

5.1.2 Standards for key management

The following are some examples of key management standards [29];

- X.509 Certificate format standard.
- Kerberos symmetric key distribution.
- ISAKMP, IKE, TLS and MIKEY Internet Standards.
- ETSI TS 102176 Electronic Signatures and Infrastructures (ESI) Algorithms and parameters for Electronic Signatures
- FIPS 186-2 Digital Signature Standard

- FIPS 140-2 Cryptographic Modules.
- ISO/IEC 11770 Key management.
- ISO/IEC 9796 and 14188 Digital Signatures.
- ISO/IEC 18031 Random Number Generation.
- ISO/IEC 18032 Prime Number Generation.
- ISO/IEC 18033 Asymmetric encryption.

The above standards can be divided into symmetric and asymmetric approaches and are mainly developed for wired or wireless (not autonomous) networks. Both approaches (asymmetric and symmetric) rely on an online centralized server for revocation of keys and also issuing new keys.

For example, Kerberos relies on an online trusted third party (TTP) to distribute session keys [40]. That is, the TTP determines which nodes are allowed to communicate with each other. When permission is granted for two nodes to establish a connection, the TTP distribute a one-time session key for that connection. This requires that the TTP is always accessible and is thus not suitable in a mobile ad hoc network. The TTP is also a single point of failure.

5.2 Solution proposed by LedsystT

Nilsson et al (LedsystT document) have proposed to manage autonomous and mobile situations with pre-created and distributed keys [30]. They propose the use of TAK2 (symmetric and asymmetric) that includes smartcards and smartcard readers. The solution of pre-created keys is practically possible. However, as described in another LedsystT document [29] manually keys have "*a large number of drawbacks*". A serious problem of manual key distribution is that it takes a long time to revoke a key. Fast (real-time) revocation of keys is especially important in mobile ad hoc networks, since a realistic threat against mobile ad hoc networks is compromised nodes. Other problems of manual key distribution are the lack of real-time distribution of new keys, weak connection to the authorization system, traceability, and misuse of keys [29].

FM has developed a symmetric session key system, which is inspired by Kerberos. The main difference is that the users must hold a symmetric key in order to identify themselves against the key server [28].

The LedsystT document "target architecture Key Management Overview" describes three different key management systems but do not discuss any key management solutions for mobile ad hoc networks [29];

1. EKMS Key distribution system handles symmetric keys used for individual entities. This system consists of one or more centralized Key Distribution Centres (KDC).
2. EKMS Group and Sensor Key management mainly deals with symmetric group keys.
3. EKMS Certificate Management system is responsible for the management of asymmetric keys and certificates.

5.3 Research in key management for ad hoc networks

Secure key management has been one of the most critical issues in the research of secure mobile ad hoc networks. Since ad hoc networks are dynamic in topology and functions should be performed by the nodes in a self-organized manner, a secure communication infrastructure needs to be established for the key management protocol. Important considerations of key management systems for mobile ad hoc networks are vulnerability and availability. Resistance against *vulnerability* is important, since a mobile node can be compromised. Thus an important feature is how many compromised nodes the system can withstand. *Availability* of the key management system is important since the nodes need to be able to contact the key management system for issuing new keys and revocation of keys.

5.3.1 Key distribution and key pre-distribution

In a key distribution protocol, an authority creates or otherwise obtains secret values and securely distributes it to other nodes. A traditional type of key management protocol using key distribution is Kerberos, which relies on a trusted third party (TTP) called a Key Distribution Centre (KDC). The KDC has to be online always.

Another type of key management protocol is based on key pre-distribution scheme, which require the existence of an offline trusted party. The offline TTP pre-initialized each node in a set with some secret information likely a set of long-lived keys. From this set of keys, any subset of nodes can non-interactively compute a common secret session key.

5.3.2 Asymmetric solutions

The main idea of asymmetric solution is to bind a trusted party (or leader) with a pair of keys. The trusted party, often called the CA, creates certificates by binding a public key to a certificate. The public key is used for entity authentication and for session key establishment. One of the biggest advantages of asymmetric cryptography is that the private key is never out in the public. Note that in symmetric solutions the secret key is known and shared by the two communicating nodes. In fact, the distribution of the shared secret key is one of the most vulnerable phases in symmetric solutions. The disadvantage of public-key encryption is the computational overhead of current algorithms. A slow algorithm may cause congestion when an attacker feeds a victim node with a large number of packets to exhaust the victim’s computation resources. Thus, asymmetric solutions are less resistant to DoS attacks.

In the following section, we describe different key management protocols based on asymmetric encryption schemes.

Distributed CA Model

The simplest approach to providing CA functionality in a mobile ad hoc network would be to assign one node to be the CA. However, this approach is not fault tolerant, since the method is not resistant to single point of failure. If an adversary succeeds to compromise the CA node (which has bad physical protection), the entire security service is compromised. Neither is the approach appropriate with respect to the expected mobility of ad hoc networks.

To provide better fault tolerance, researchers have proposed ways of distributing the CA functionality to several nodes. Many proposed solutions are based of a cryptographic technique called Secret Sharing, which was first proposed by Shamir [49]. In Shamir´s secret sharing

scheme (k, n), a secret is slit into n pieces according to a random polynomial. The secret can be recovered by combining k pieces based on Lagrange interpolation. The Shamir approach has been applied to cryptography keys and is also known as Threshold Cryptography. Threshold cryptography is thus a technique in which a group of nodes forms a CA.

One of the earliest attempts to use threshold cryptography in a peer group is the work of Zhou and Haas [21], which propose to distribute the responsibility of the CA to t+1 nodes. Thus, the private key of the CA is divided into t+1 shares. When a node requests a public key, each of the server nodes signs the requested public key with its share of the system's secret key. Thus, an obvious problem is the workload by the server nodes, since they have to respond to all requests. Another issue is that an adversary can get access of shares of the private keys. As long as not more than t nodes are compromised the certificate service can operate (n>=3t+1, n number of nodes). Furthermore, this scheme does not describe how a node can contact t servers, if the servers are scattered in the whole area. Thus, the solution assumes a small group of servers with rich connectivity.

One important problem with the Zhou and Haas approach is the *availability* of server nodes. In mobile ad hoc networks, there is no guarantee of connectivity between any two nodes. Even though the CA is secured and on-line, it does not mean that nodes will be able to reach the CA. Kong et al [16] improve the Zhou and Haas approach by addressing the availability problem. They propose a localized key management scheme. In this scheme, the distributed CA consists of any local multiple (say k) nodes that collaboratively serve as a certificate authority server. That is, any local k nodes collaboratively provide authentication service. However, one problem is that in case the threshold k is to large, nodes will have to keep moving to get certificates updated. If k is set to a relatively low number, compromising k nodes around a victim node is a relatively easy task.

Luo et al [18] have improved the proposed protocol by Kong et al by letting *k* depend on network conditions. Shares can also be updated in case compromised nodes are detected. Nodes are notified about compromised nodes by flooding a list of revoked certificates. To obtain a certificate a node has to identify itself to k nodes. Thus, the method assumes that each node has an initial certificate before it can join the network. This assumption is realistic for tactical networks. However, there are several issues that need further investigation. The method assume that all nodes are trusted or a reliable intrusion detection system to identify misbehaving nodes among its one-hop neighborhood. Currently, there are no reliable intrusion detection systems for mobile ad hoc networks, see section 9.

More recently, Yi et al have proposed a different distributed CA method [50]. In their approach, certificate service is distributed to Mobile Certificate Authority (MOCA) nodes, which are chosen based on an observation of heterogeneity within the network. The MOCA nodes are more secure and computationally powerful nodes. A node can locate MOCA nodes either randomly or through the shortest part. An issue in this approach is how nodes can locate these paths securely, since most secure routing protocols are based on the establishment of a key service.

Identity-based scheme

Identity-based models are based on the idea that a unique identity (e.g. e-mail address) is used as the public key. Thus, no public key certificates are needed. However, this method requires a CA at the initial stage of the network in order to distribute the secret keys of all users. Thus, an obvious drawback is that the CA knows all keys. Another problem is that the method assumes a centralized CA. Examples of identity-based models are [10, 14 and 18].

Self-organized Model

Hubuax et al [9, 14] have proposed a self-organized public-key management system for ad hoc networks, which is similar to PGP. In this system, every node acts as its own CA. Thus, each node maintains a local certificate repository. Each node also issue and distribute its own certificates. When a node wants to verify a public key of another node it tries to find a certificate chain to the other node. In other words, to obtain the certificate of another entity the requester builds a certificate chain from his repository list and implicitly trusted entity list until a path to an identity that has the desire certificate in its repository is found. Thus, a problem is that it is possible that a trust chain does not exist to the other node. Another problem is the lack of any trusted secure party (director) and the lack of certificate repositories.

5.3.3 Symmetric solutions

Traditional key distribution protocols rely on an infrastructure with one important component, the trusted third party (TTP). One example of a traditional key distribution protocol is Kerberos, which relies on an online trusted third party (TTP) to distribute session keys [40]. This solution requires that the TTP is always accessible and is thus not suitable in a mobile ad hoc network.

Hierarchical approaches

A number of approaches attempt to increase the availability of the TTP by replicating the online key server to a subset of nodes arranged hierarchical or arbitrary [41, 42]. This is usable only if the nodes really move in groups without loosing contact with each other. There is also an issue regarding the scalability of these schemes.

Pre-distributed scheme: Symmetric key derived from small amount of secret data

Blom et al and Matsuoto et al have also proposed a method to derive keys to all nodes in the network from a small amount of data [45, 47]. The nodes are pre-initialized with some secret information. From this secret information any subset of nodes can compute a common session key. Thus, two nodes can compute a common key without interaction. A relevant question is how strong this algorithm is.

Pre-distribution scheme for sensor networks

Eschenauer et al have proposed a key management scheme for sensor networks, which is based on symmetric key pre-distribution [51]. An offline trusted party pre-initializes each node in a set S with some secret information with which any subset of nodes later can compute a common session key. One problem of the solution is that the nodes may not know each other before deployment. In other words, they do not have any prior knowledge to which it is going to meet. This may not be a problem in a tactical mobile ad hoc network. Another more important issue is how strong these algorithms are. Further investigation is needed to determine if these algorithms are strong enough for military systems.

5.4 Group key management protocols

Secure group communication requires scalable and efficient group membership management. One important issue in group key management is to ensure backwards and forward secrecy. That is, when one node leaves the group the other nodes should receive a new key in order to make sure that the leaving node can not eavesdrop on the traffic later.

One example of group key management intended for military scenarios is made by Rhee Et al [52]. They have proposed a group key management architecture that uses a two-layer key management protocol where a group of keys is divided into cell groups (ground nodes) and control groups. Each cell is managed by a single mobile backbone node, which managed its group by generating, updating and distributing the group key shared among all the cell members. Thus, an obvious problem is the single point of failure, i.e., the system is not fault tolerant. It could also be a problem if you want to change the mobile backbone node.

5.5 Conclusion

Research in the area of key management for mobile ad hoc networks is still in its early age. Currently, manually pre-created distributed keys are the only method that can be applied to mobile ad hoc networks without further investigation. TAK2 can be used for both symmetric and asymmetric keys. However, as mentioned in section 5.2, manually distributed keys have a large number of drawbacks. Relevant problems are lack of real-time revocation and distribution of keys, weak connection to the authorization system, traceability, and misuse of keys [29]. Fast (real-time) revocation of keys is especially important in mobile ad hoc networks, since a realistic threat against mobile ad hoc networks is compromised nodes. This also means that the keys are used for relatively long time, which gives an adversary long time to crack the key. In the long term, pre-created distributed keys can be used for initialization, but one should search for a key management method that provides a real-time service.

Methods that are relevant for further investigation are;

- Pre-distribution schemes: In pre-distribution schemes (such as [51]) an offline trusted party pre-initializes each node in a set S with some secret information with which any subset of nodes later can compute a common session key. Further investigation is needed to determine if these algorithms are strong enough for military systems.

- Hierarchical approaches: A number of approaches are based on hierarchical techniques. For example, [41, 42] attempt to increase the availability of the TTP by replicating the online key server to a subset of nodes arranged hierarchical or arbitrary. These methods are only usable if the nodes really are moving in groups without loosing contact with each other.

- Secret sharing techniques: Zhou et al focus on the share updating procedure [21]. Kong et al [15] improve the Zhou and Haas approach by addressing the availability problem. Yi et al emphasize more efficient communication with the use of Mobile Certificate Authority (MOCA) nodes [50]. None of these methods can be used without further development. Relevant problems that need to be considered are vulnerability and availability. Resistance against *vulnerability* is important, since a mobile node can be compromised. Thus an important feature is how many compromised nodes the system can withstand. *Availability* of the key management system is important since the nodes need to be able to contact the key management system for issuing new keys and revocation of keys.

- Self-organized Model: Hubuax et al [9, 14] have proposed a self-organized public-key management system for ad hoc networks. In this system, every node acts as its own CA. Each node also issue and distribute its own certificates. When a node wants to verify a public key of another node, it tries to find a certificate chain to the other node. Thus, a problem is that it is possible that a trust chain does not exist to the other node. Another

issue is the lack of any trusted secure party (director) and the lack of certificate repositories.

6 Secure routing

In a mobile ad hoc network, nodes within radio range communicate directly with each other, while nodes that are too far apart use other nodes as relays (i.e. use routing protocols). The existing routing protocols are designed without consideration of security and are thus subjected to a variety of attacks. Routing attacks can be performed by internal attackers as well as external attackers. Below is a list of different types of routing attacks.

- Routing disruption attacks.

  In routing disruption attacks, packets are routed improperly. Two types of routing disruption attacks are falsification and interference. Interference means that a malicious node tries to disrupt the exchange of messages between two legitimate routers. In falsification, a malicious node sends falsified information. If the malicious node acts as origin, it can claim routes for which it is not authorised, or routes that do not exist at all. The node could also inject falsified information. If the malicious node acts as forwarding node, the node could delete or modify information that should be forwarded.

- Resource consumption attacks

  Resource consumption aims at using up resources such as bandwidth, memory and computer capacity.

- Impersonation

  In impersonation, malicious nodes assume the identity of legitimate nodes in routing packets.

- Eavesdropping

  Malicious nodes capture all traffic (including routing traffic) and thus obtain routing information.

- Traffic analysis

  Traffic analysis means that a malicious node analyses all captured/received traffic in order to extract information, such as, which nodes are communicating frequently or exchange huge amounts of data.

There are basically two approaches to protect the routing protocols. One approach, named proactive, attempts to prevent the adversary from performing attacks by using cryptography techniques. Different types of cryptography techniques are symmetric, asymmetric, and one-way HMAC key chain solutions. The second approach, intrusion detection and response, attempts to detect routing attacks and then react against the detected attacks.

6.1 Symmetric solutions

As mentioned above, the main principle of symmetric encryption is that only a single key is used for authentication, encryption and decryption. Symmetric solutions assume that the nodes share a secret and use this shared secret to compute the encrypted information.

Ariadne for Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR) and Secure Routing Protocol (SRP) take the symmetric approach for protecting against routing attacks.

Advantages:

- This technique is efficient and affordable for low-end devices.
- Symmetric-key ciphers can be designed to have high rates of data throughput

Disadvantages:

- Scalability: A total number of *n*(n-1)/2* keys must be maintained in a network with n nodes. Note that group keys can not be used to protect against several types of routing attacks such as impersonation, interference and falsification.
- Distribution of symmetric unique keys in an ad hoc network is still a topic that is not solved by the secure routing protocols. One possible solution which solves part of the problems is to use asymmetric keys saved on active cards (TAK2) to distribute symmetric keys, see discussion in section 3.1.

	Encryption	Authentication/Integrity
Routing disruption (origin node)		
Routing disruption (forward node)	YES	YES
Impersonation		YES
Resource consumption		
Traffic analysis		
Eavesdropping	YES	

Figure 5: Protection against routing attacks with symmetric solutions.

Ariadne

Aridne [60] is a secure On-Demand Routing Protocol for Ad Hoc Networks based on Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR) [77]. The protocol can authenticate routing messages using one of three schemes: shared secrets between each pair of nodes, digital signatures, and shared secrets between communicating nodes combined with broadcast authentication. The last scheme is described in this report.

Ariadne assumes a shared unique secret between the communicating nodes. This secret is used for message authentication. First, the source node initiates the route request message and computes the Message Authentication Code (MAC) of the shared secret known only by source and destination. Then, the destination authenticates the origin and value of the request by re-

computing the MAC and comparing it with the received MAC. For source routing protocols, the main challenge is to ensure that each intermediate node cannot remove existing nodes from or add extra nodes to the path [63]. This is achieved in Ariadne by using hash chains. That is, intermediate nodes compute their own MAC using an entry from the hash chain and using additional fields that identify the address of the intermediate node. The additional field and the MAC are added to the request.

Secure Routing Protocol (SRP)

One approach to secure the routing discovery procedure is the Secure Routing Protocol (SRP), which is proposed by Papadimitratos et al [56]. SRP can be applied as an extension of the Dynamic Source Routing (DSR) protocol and the Zone Routing Protocol (ZRP).

The SRP guarantees that a node initiating a route discovery will be able to identify if an intermediate node on the path to the destination node modify or inject falsified information. This goal is achieved with the existence of a security association between the pair of end nodes. That is, the source and destination address and the unique (with respect to the pair of end nodes) query identifiers are the input for the calculation of the Message Authentication Code.

6.2 One-way HMAC key chain

A chain of outputs can be obtained by repeatedly apply a hash function h() on an initial secret input x. These outputs can be used in the reverse order of generation to authenticate messages. Thus, the outputs from the hash function (x, h(x), h(h(x)), etc) are used as a one-way key. A broadcast authentication protocol based on a hash-chain based protocol is TESLA [25, 27], which can be used to secure routing updates.

SEAD for DSDV uses one-way HMAC key chain for protecting against routing attacks.

Advantages:

- Authentication is lightweight, since the validation does not require much processing power. Hash chains can also be constructed before network set-up.

Disadvantages:

- One authenticated initial hash value needs to be exchanged.
- TESLA requires tight clock synchronization.
- The receiving node need to buffer a message for verification until the key is revealed. Thus, the verification of the messages is delayed.

	Hash chain
Routing disruption (origin node)	
Routing disruption (forward node)	YES
Impersonation	YES
Resource consumption	
Traffic analysis	
Eavesdropping	

Figure 6: Protection against routing attacks with hash chains.

Secure Efficient Distance Vector Routing (SEAD)

SEAD is a secure ad hoc network routing protocol using distance vector routing [59]. The design of SEAD is based on the DSDV-SQ version [78] of the protocol. For distance vector routing protocols, the main challenge is that each intermediate node has to increment the routing metric correctly [63]. SEAD uses the elements from one-way hash chains to provide authentication for both the sequence number and the metric in each entry. In contrast to Ariadne, which is based on end-to-end security, SEAD operates on a hop-by-hop basis due to the basic operation of distance vector routing.

6.3 Asymmetric solutions

As mentioned above, in asymmetric solutions each node has a certificate issued by a certificate authority (CA) and an assigned public/private key pair. The public key can be used to encrypt messages, since the encrypted message only can be decrypted with the private key. The private key is used to sign a message. A signed message can be validated using the corresponding public key. SAODV and ARAN take the asymmetric approach for protecting against routing attacks**.**

Advantages:

- Asymmetric solutions are scalable, since only a total number of *2n* keys (private and public keys) must be maintained in a network with *n* nodes.

- One of the biggest advantages of asymmetric cryptography is that the private key is never out in the public. Note that in symmetric solutions the secret key is known and shared by the two communicating nodes. Asymmetric cryptography addresses this problem by distributing open public keys and symmetric session keys encrypted with the private key.

Disadvantages

- The disadvantage of public-key encryption is the computational overhead of current algorithms.

- Generation and administration of public/private keys require a key management system. Today, in autonomous systems, this is often handled with a static trusted third party and manually key exchange.

	Encryption	Authentication/Integrity
Routing disruption (origin node)		
Routing disruption (forward node)	YES	YES
Impersonation		YES
Resource consumption		
Traffic analysis		
Eavesdropping	YES	

Figure 7: Protection against routing attacks with asymmetric solutions.

ARAN

ARAN is an on-demand routing protocol that provides authentication and non-repudiation using pre-determined certificates [58]. The certificates contain the IP address of the node. A node initiating a route discovery, constructs the message, which includes the destination IP address, its certificate, and a nonce. The sent message is signed with the private key. Intermediate nodes receiving the route discovery will verify the message by using the initiating node's public key. Thus, the message is per hop validated. The destination node verifies the last intermediate node's signature and the source signature with the public keys. If both checks are successful, a reply is constructed and digitally signed.

Secure-AODV

Secure AODV [57] is an extension to the Ad Hoc On-demand Distance Vector (AODV) routing protocol [4]. The Secure-AODV protocol assumes that each node has public keys of all other nodes, so that intermediate nodes can validate received routing messages. The basic idea is that the originator of a routing message appends a digital signature and the last element of a hash chain. Intermediate nodes, that receive a routing message, validate the signature and the hash value. Then the intermediate node forwards the packet along with the generated k-th element of the hash chain. K is the number of traversed hops. Note that the hash chain in this case does not need time synchronization, which is different from the one-way HMAC key-chain for authentication.

6.4 Routing security: multipath routing

A different approach to secure ad hoc networks is multipath routing. First, the protocol determines a set of diverse paths connecting the source and destination. Then, it disperses a message into N pieces, so that successful reception of any M-out-of-N pieces allows

reconstruction of the original message at the destination. The individual parts are sent via multiple paths from the source to the destination. Upon reception of the messages, the destination node informs the source node about intact routes.

Advantages:

- The multipath routing provides protection against man-in-the-middle attacks, modification and replay attacks by sending the individual parts over multiple channels.
- Due to redundancy, the system is more resistant against jamming against one link.

Disadvantages:

- The method relies on multiple paths which may not exist. All current routing algorithms do not support multipath routing.
- The method uses routing to secure data in ad hoc networks, but does not secure routing.
- Malicious nodes could still execute routing attacks.

Bouam [61] take the multi-path approach. Papadimitratos and Haas [62] have extended this approach in the Secure Message Transmission protocol. Each piece of the N pieces is equipped with a cryptographic header that provides integrity, replay protection and origin authentication.

6.5 Conclusion and discussion

The secure ad hoc routing protocols take the proactive or reactive approach to enhance ad hoc routing protocols such as DSR and AODV with security extensions. The reactive approach is reviewed in section 9.2 (research for intrusion detection for ad hoc networks). The conclusion of the reactive approach (in section 9.2) is that the methods really need to be refined before they are practical useable.

The proactive approach may provide protection against impersonation and routing disruption by forwarding node, but can not protect against routing disruption by origin node, resource consumption and traffic analysis, see figure below. Different types of intrusion detection solutions try to target the remaining unsolved routing attacks, such as routing disruption by origin node, but currently there is no technique that can be used without a lot of further improvements and development. Thus, the available (proactive) solutions will enhance the security, but will not protect against all different types of routing attacks.

	Encryption	Authentication/Integrity
Routing disruption (origin node)		
Routing disruption (forward node)	YES	YES
Impersonation		YES
Resource consumption		
Traffic analysis		
Eavesdropping	YES	

Figure 8. Protection against routing attacks with proactive approach.

There are basically three different approaches that provide a proactive solution; symmetric, asymmetric and one-way HMAC key chain. If it is possible to handle the distribution of symmetric unique keys, the symmetric approach along with hash chain is recommended. One solution is to use asymmetric keys saved on active cards (TAK2) to distribute symmetric keys.

7 Distributed Firewalls

The conventional firewall is something that protects traffic flowing through a single point between the inside and outside network. They rely on the assumption that everyone on one side of the firewall is trusted and that anyone on the other side may be an enemy. In contrast, the distributed firewall is implemented on every node. A distributed firewall is a mechanism to enforce a network domain security policy through the use of a policy language. Certificates can be used to enable the identification of any member of the network policy domain. The policy is set by a central management node.

Bellovin [64] lists three components to implement a distributed firewall;

- **Policy language**: A policy language states what sort of connections are permitted and prohibited. For example, the language can consist of packet filtering rules in a packet filtering firewall.

- **System management and safe distribution:** A management tool changes and enforces the security policy. Note that a security mechanism is needed to distribute the security policy in a safe way.

- **A mechanism that applies the security policy to incoming (and outgoing) requests**

A distributed firewall can be used for simple packet filtering, i.e. filter out certain MAC addresses. A distributed firewall is also excellent at rejecting connection requests for inappropriate services. That is, the firewall includes filtering of ports or an application proxy that investigate the application layer packets. This function is relevant if the application is inside the node, but the solution is also relevant if the application is located outside the mobile ad hoc node. First, even if applications such as situation awareness are located outside the node, the node will probably always include some type of applications (e.g. management application). Second, if the node is connected to an internal LAN that includes the application, the firewall can be configured to protect the application and the internal LAN.

Furthermore, firewalls also have other features such as protection against some DoS-attacks and port scanning as well as attacks based on fragmentation. The firewall can also be used in combination with an intrusion detection system. For example, if the intrusion detection system detects a malicious node, it could reconfigure the firewall to drop all packets from the malicious node. The firewall should be configured to restrict both ingoing and outgoing traffic. The restriction of outgoing traffic will make it more difficult for an internal attacker to perform attacks.

Advantages:

- Distributed firewalls can be used in mobile ad hoc networks. It is a step in the right direction to enforce network based filtering.

- Packet filtering is based on the identity in the certificates. Thus, the risk of spoofed identities is minimized.

- If the mobile ad hoc node is equipped with both encryption and authentication, the main advantage of the firewall is to make it more difficult for an internal attacker to perform

malicious attacks. The firewall also gives a multi-layer protection. That is, if an adversary succeeds in cracking the key or encryption algorithm, the firewall will restrict the adversary from performing some types of attacks against the node. For example, it is more difficult to infiltrate the node.

Disadvantages:

- Distributed firewalls must be centrally managed. A possible solution is to update the firewall rules when the server is accessible. However, this means that it may take a long time to enforce a new important firewall rule.

- A difficult issue in dealing with certificates is revocation, especially when no online connection is possible.

- Currently there is no support for user authentication and access control.

- There is no support for extending policy to higher layers. For specific applications, the policy needs to extend to these applications to enforce rules of the applications developed for a tactical network.

There are several commercial products for software firewalls such as Symantec, Zone Alarm, f-secure and IPtables. The principle in these firewalls can be used. Note that these software firewalls are dependent on the operating system which may limit their field of application. Another important issue is management. Even a small set of firewalls, which have several dozens of rules, can quickly become a maintenance nightmare. To make a firewall distributed a *good* management tool is needed. It must be easy to use and secure. At the same time, be applicable to a mobile ad hoc network where a centralized always on-line server is not usable. A possible solution may be to update the firewall rules when the server is accessible.

Conclusion

The concept of distributed firewalls is a valid one. It would indeed increase the security of a mobile ad hoc network. The principle of available software firewall products is usable. However, there are issues for further development, such as management and addressing authentication (and access control) of the operating system.

8 Location/identity separation

Currently, the Internet has two global namespace, Internet Protocol (IP) addresses and Domain Name Service (DNS) names. The IP address is often used for two purposes. The first one is to describe the location of a network interface attached to a network. The second one is to identify the node hosting the network interface. Thus, the IP address is intended for both *location* and *identification* of a host. This means that node mobility becomes more difficult. It also has an impact on security, since an IP address of a mobile host can not be used to identify a host uniquely. An IP address may be seen as a temporary identifier of a mobile host.

Examples of research in the area of location and identifier split are;

- **Host Identity Protocol (HIP)**
- **Forwarding directive, Association, and Rendezvous Architecture (FARA)**
- **Split Naming/Forwarding Network Architecture (SNF)**
- **Internet Indirection Infrastructure**

In this report, we evaluate the HIP protocol, since this protocol is discussed in LedsystT.

Host Identity Protocol (HIP)
The Host Identity Protocol (HIP) [36, 37] might solve the problem of location and identifier split, although it currently is only a draft. HIP provides a new namespace intended to fill a gap between the IP and DNS namespaces. HIP also provides a new namespace with the aim to provide a secure unique identity of the host. That is, the Host Identity namespace consists of Host Identifiers (HI), which is the public key of an asymmetric key pair. The Host Identity, and the corresponding Host Identifier, can either be public (e.g. published in the DNS), or unpublished. The Host Identity is an abstract concept assigned to a computing platform whereas the Host Identifier is a public key used as a name for a Host Identity. In other words, the Host Identity refers to the abstract entity that is identified whereas the Host Identifier refers to the concrete bit pattern that is used in the identification process.

HIP is placed between IP (layer 3) and TCP (layer 4). In order to start the HIP protocol, the source node has to know how to reach the other mobile node. This is usually accomplished through a DNS like server. This server is used for translating the Host Identity into the current IP address.

HIP includes a simple key exchange protocol, and results in a key that can be used in Encapsulated Security payload (ESP) in IPsec. Thus, IPsec is used to carry the actual data traffic. The HIP base exchange uses the Host Identifies to set up a pair of keys to enable Encapsulating Security Payload (ESP)[1] in an end-to-end manner.

HIP is design to reduce certain types of denial-of-service (DoS) attacks, but will absolutely not defend against all types of DoS attacks. The protocol will also enhance the security in IP networks, since it provides a security identity based on asymmetric encryption. However, there

[1] ESP is a protocol in IPsec.

are also security issues when using HIP in mobile ad hoc networks. The protocol is based on asymmetric encryption, which assumes an always-accessible server for revocation of keys.

Advantages

- The identity of the computer will not reveal its location.
- The protocol provides a secure unique identity based on asymmetric encryption.

Disadvantages

- HIP assumes an always on-line DNS like server, which makes it unsuitable for a distributed mobile ad hoc network.
- HIP is based on asymmetric cryptography. One disadvantage of public-key encryption is the computational overhead of current algorithms. Generation and administration of public/private keys also require a key management system of some type. A difficult issue in dealing with certificates is revocation, especially when no online connection is possible.

9 Intrusion detection systems and response

An *intrusion* can be defined as “any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource” [65]. *Intrusion detection systems* (IDS) detect intrusions by analyzing data collected from the network or the host and try to prevent such activities that may compromise system security. An IDS is a system that automates supervision of events by analyzing collected data to make conclusions about ongoing IT attacks. A necessary condition for efficient detection is that intrusion activities have distinct behaviors that are observable.

IDSs can be classified according to the audit source location. *Host Based Intrusion Detection Systems* (HIDS) analyzes data obtained from the host such as system log files and applications. *Network Based Intrusion Detection Systems* (NIDS) analyze data from the network such as the HTTP protocol.

The methodology of detection can also be used for classifying IDSs. Three approaches are used for the detection of intrusions: anomaly detection, misuse detection (signature matching), and specification-based detection [66]. All of these alternatives can be combined in the design of an intrusion detection system.

The most used method is *misuse detection*, which use the characteristic patterns of already known attacks. Commonly, the precision of this method is often good, with few false alarms, but it depends entirely on the knowledge of how the intrusion is achieved. With *anomaly detection*, we instead monitor activities and estimate patterns for normal behaviors. Patterns that differ from the norm generate alarms. In this way, previously unknown attacks can be detected, but one challenge with this technique is to avoid alarms when infrequent, but legal, activities take place. The *specification-based detection* method tries to avoid this problem. The main challenge in specification-based detection is how to define the set of constraints that describe the correct operation of the protocols to detect known and unknown attacks efficiently.

IDSs with response are needed in mobile ad hoc networks due to;

1. Mobile nodes in a hostile environment often have *poor physical protection* compared to what is normal for fixed networks [67]. A mobile node can be stolen or hijacked or an intruder can penetrate the security mechanisms. Thus, we must not only consider attacks from external nodes, but also take into account attacks from internal compromised nodes

2. Compared to centralized networks, nodes in a mobile ad hoc network cannot assume a *clear line of defense*. By definition, a mobile ad hoc network does not have a central security mechanism (e.g. router, gateway, firewall, central authentication) that can fend off certain attacks that are targeted at the whole network. The wireless channel is accessible to both legitimate users and adversaries. Furthermore, to achieve an autonomous ad hoc network, many protocols developed for mobile ad hoc network are based on *distributed algorithms*. These distributed algorithms open the way for new attacks, since the algorithms are based on the cooperative participation of nodes. If one node is malicious, it can affect the entire network. To summarize, no clear line of defense, distributed algorithms and bad physical protection means that *internal attacks* can not be neglected in mobile ad hoc networks.

3. It is difficult to design and implement software systems without introducing *design and programming errors* that an adversary can exploit. If an adversary has adequate resources and tries hard enough, there is a risk that the adversary succeeds in infiltrating the system. For example, even though buffer overflow has been a known security problem for many years, there is still recently released software with buffer overflow security holes. If the buffer overflow security hole is exploited, it may lead to an unauthorized root shell. In order words, someone can infiltrate the system.

4. History has taught us that no matter how many intrusion prevention mechanisms (e.g. encryption, authentication and firewalls) are inserted in networks, there are always weak links that adversaries can exploit. Intrusion detection systems are part of typical defense in depth strategies.

To summarize, the possibility of compromised malicious nodes performing internal attacks is a severe threat to a mobile ad hoc network. Cryptography can reduce the amount of successful intrusions, but cannot fully eliminate them. Encryption and authentication provide protection against some types of attacks from external nodes, but will not protect against attacks from inside nodes, which already have the required keys.

Hence, to obtain an acceptable level of security in military contexts, traditional security solutions should be coupled with *intrusion detection systems* (IDS) that continuously monitor the network and determine whether the system (the network or any node of the network) is under attack. An intrusion detection system collects and analyses audit data to detect intrusions. Once an intrusion is detected, e.g. in the early stage of a denial of service attack, a response can be put into place to minimize damage.

The goal is not perfect protection, but to make it more difficult to perform successful attacks. Cryptographic mechanisms are the first line of defense and intrusion detection systems the second.

9.1 Evaluation of wireless intrusion detection tools

An evaluation of wireless intrusion detection tools is presented in [68]. The evaluation is summarized in this section. In [68], sixteen wireless 802.11 intrusion detection tools were theoretically evaluated by documentation. From these sixteen tools, ten tools were chosen that were best suited for mobile ad hoc networks. These ten tools were evaluated according to the requirements identified in section 5 in [68]. The evaluated tools were AirDefence [69], Airespace [70], AirMagnet Distributed [71, 72], AirMagnet Laptop [71, 72], AirXone [73, 74, 45], Aruba [44], Red-Detect [43], RFprotect [35], and Snort [34].

The following is a summary of the evaluation of tools according to requirements in section 5 in [68]:

- The IDS should be able to detect attacks based on anomaly detection and specification-based detection.

  All tools can detect attacks based on signature matching. Most tools also claim to detect attacks based on anomalies, but the provided methods only cover a few attacks on the

application level. For example, AirXone can detect inappropriate and intrusive behavior by authorized users, e.g., file-sharing, file transfer and instant messaging.

- The IDS should be able to perform automatic response

Even though a few tools have automatic responce to some detected attacks, no tool can respond to detected attacks with such power that the system administrator is not needed. A qualified criticism is that the problem of the tools is actually that the accuracy of the detected attacks it to bad to be able to perform a (automatic) response without further investigation of a system administrator. (It could also be discussed if an automatic response is desirable. However, a true mobile ad hoc network should have features such as self-organizing and autonomous, which means that it should be able to run it without a system administrator.)

- The IDS solution should be scalable.

All tools are scalable except for the distributed solution of Snort and AirMagnet laptop. Snort and AirMagnet laptop requires a system administrator to analyze the logs and take appropriate action against the detected attack at every physical location of the nodes.

- The IDS should be applicable to tactical mobile ad hoc networks

The requirement ”applicable to tactical mobile ad hoc network” is not fulfilled by any tool. A typical architecture is a hardware or software sensor and a centralized management server with a console for the system administrator. First, the sensor gathers and processes intrusion data and sends it to the centralized server. Next, the system administrator analyzes the data to take appropriate action against the detected attack. The problem with this solution is that it assumes continuous contact with the management facility.

The sensors are updated through a link to a centralized authentication database or manually with information (e.g. new MAC addresses and new signatures). Manual administration is neither scalable nor feasible in a self-maintaining mobile ad hoc network. Communication with a centralized database is not always possible because of the autonomous nature of mobile ad hoc networks. However, the signatures could be updated from centralized servers when they are accessible.

- The IDS should be able to detect attacks against each layer of the TCP/IP stack

The requirement “Detect attacks against each layer of the TCP/IP stack“ is very general. To clarify what attacks the tools can detect, the requirement is divided into three important parts; detect relevant wireless attacks, detect attacks against protocols developed for ad hoc networks, and detect attacks in the application layer and transport layer.

  - Some tools can detect attacks against all layers. For example, snort-wireless is compatible with standard Snort, which includes over 1500 prewritten rules. These rules mainly cover attacks against the transport layer and application layer. Example of examined protocols in Snort are; HTTP, DNS, POP2, POP3, IMAP, FTP, TFTP, ICMP, rsh, rlogin, SMTP, SQL, and SNMP. If the provided rules (signatures) by Snort are useful or not depends on the applications/protocols used in the mobile ad hoc network.

- No tool can detect any attack against a protocol developed for mobile ad hoc networks.

- The evaluated tools are intended for wireless networks, but can detect few relevant attacks against a tactical mobile ad hoc network. That is, many of the attacks detected are not relevant in a tactical mobile ad hoc network. For example, most tools can detect unauthorized access points and unauthorized hosts, which is a weak substitute for strong encryption. We want to use the intrusion detection system as a complement to strong encryption, i.e., not use it instead of strong encryption. However, the evaluated tools can detect some relevant attacks. Many tools can detect scanning from known products such as Netstumbler and Wellenreiter. The scanning is normally a warning of an incoming attack. The tools can also detect a few relevant DoS attacks. For example, red-detect can detect authentication floods and probe frame attacks. Some tools can also detect the presence of an unknown ad hoc network.

Conclusion of evaluation

The vast difference between corporate wireless networks and tactical mobile ad hoc networks makes it difficult to apply intrusion detection techniques developed for a corporate wireless network to a tactical mobile ad hoc network.

The most important difference is perhaps that the entire network is mobile in a tactical mobile ad hoc network, whereas only the nodes are mobile in corporate wireless networks. Thus, intrusion detection tools for tactical mobile ad hoc networks must be independent of centralized functionality and fixed infrastructure. Even though many of the evaluated intrusion detection tools implement IDS sensors in a distributed manner, without dependency of fixed wired infrastructure, they are still dependent on centralized management. That is, the distributed IDS sensors generate alerts that are sent to centralized servers for storage and analysis by the system administrator. Thus, detected attacks by the sensors can be unknown to the management facility for a long time, since nodes in tactical mobile ad hoc networks can lose contact with centralized servers. However, some intrusion detection tools [34, 71] can be deployed without centralized servers, but these solutions require a system administrator to manually analysis the logs at every node. Thus, this solution is not scalable. To summarize, research is required to develop a tool with an architecture that can be applied to a tactical mobile ad hoc network.

Furthermore, the ability to detect previously unknown attacks is desirable in military networks, since military organizations have resources to develop attacks that are unknown and not used in civilian contexts. Thus, the intrusion detection system should be able to detect previously unknown attacks based on anomaly detection (behavior-based) or specification-based detection. Unfortunately, all tools evaluated have their strength in signature matching. Even though many intrusion detection tools can detect attacks based on anomalies, the provided methods can only cover a few attacks. Thus, research is needed to develop methods that can detect previously unknown attacks, which also cover many attacks.

Moreover, the tools evaluated can only detect a few relevant wireless attacks on a tactical mobile ad hoc network. For example, it is not relevant to detect rogue access points because a mobile ad hoc network does not have access points. Another issue is that no tool can detect any attack against protocols developed for mobile ad hoc networks such as auto configuration protocols and routing protocols. However, some tools can detect a few DoS attacks, which are relevant also in a tactical mobile ad hoc network. Several tools can also detect a lot of application layer and

transport layer attacks in wired networks, which is relevant if these applications are used in the mobile ad hoc network. However, a tactical mobile ad hoc network will probably also include many specific military applications. To summarize, research and development is required to detect more attacks relevant in the context of tactical mobile ad hoc networks.

9.2 Research in intrusion detection for ad hoc networks

This section presents examples of relevant research in the area of intrusion detection and response for wireless mobile ad hoc networks.

Distributed intrusion detection for ad hoc networks

Zhang and Lee propose that intrusion detection and response should be both distributed and cooperative to suite the needs of wireless network [24]. In their proposed architecture, every node participates in intrusion detection and response. Each node is responsible for detecting intrusions locally. Neighbor nodes can also collaborate in global intrusion detection actions when an anomaly is detected in local data or if there is inconclusive evidence.

The internal of an IDS agent is structured into six pieces. The data collection module is responsible for gathering local audit traces. Next, the local detection engine uses the gathered local audit traces to detect local anomaly. The collaboration engine is used if the detection methods need broader data set or require collaboration of nodes. Intrusion responses are provided by both the local response and global response modules. Finally, a secure communication module provides a high-confidence communication channel among IDS agents.

The main contribution of this work is that it presents a distributed and cooperative intrusion detection architecture based on anomaly detection techniques. One problem with the architecture is that it relies on cooperative participation between nodes for detection and response, even though the purpose really is to determine if any of these nodes are malicious. The paper does not describe how the detection is performed.

Anomaly detection for Mobile ad hoc networks

Furthermore, Zhang, Lee and Huang have proposed a method to detect attacks against protocols based on anomalies [75]. They describe a procedure for anomaly detection, which can be used to build an anomaly model for an ad hoc network. First, they select audit data and perform appropriate data transformation. Two local data sources are used for anomaly detection, local routing information (cache entries and traffic statistics) and GPS information (Physical movement is measured by velocity and distance). Second, they compute classifiers using training data and apply the classifiers to test data. Thus, classification algorithms are used to build anomaly detection models. A classifier is trained using normal data. For each training run, a corresponding model is built. The classifier predicts the next event, given the previous n events. When the actual event is not what the classifier has predicted there is an anomaly.

The results of their experiments, for two different classifiers (Ripper and SVM-Light), are presented below. The results show that anomalies may be used for detecting routing attacks, but the methods really need to be refined. Their method resulted in many false alarms.

	RIPPER	RIPPER	SVM_LIGHT	SVM_LIGHT
Routing protocols	Detection rate	False Alarm Rate	Detection rate	False Alarm Rate
DSR	85-91%	9-15%	99%	0.03-0.07%
DSDV	85-91%	5-24%	84-86%	6-26%
AODV	88-92%	1-20%	94-97%	1-4%

Figure 9: Detection rate and false alarms rates of anomaly basic events.

Watchdog and Pathrater

Sergio Marti et al. have proposed a watchdog technique that detects malicious neighbor nodes by listening on their incoming and outgoing traffic [76]. Each node has a watchdog. For example, the watchdog verifies that the next node in the path also forwards the packet by listening to that node's transmissions. Their method resulted in a lot of problems that are still unsolved. For example, a misbehaving node may not be detected, since a node can have difficulties to listen on forwarding node's signals due to collisions. Moreover, a neighbor node cannot know if the forwarded packet actually reaches the receiver (forwarder might lower transmission power).

Attack analysis and Detection for Ad Hoc Routing Protocols

In a recent paper, Huang and Lee [19] have proposed to detected attacks based on both specification-based and statistical based approaches. First, normal events of the protocol are modeled with extended finite state automation (EFSA) according to protocol specification. The EFSA can detect anomalies of events that are direct violations of the specifications. Second, they detect statistical anomalies by constructing statistical features from the specification. That is, statistics on the states and transitions of the EFSA is used to train a detection module to detect those anomalies that are statistical in nature.

The research is validated with the AODV routing protocol. That is, an EFSA is build from the RFC of the AODV protocol. The proposed EFSA is evaluated in experiments on a wireless emulation platform. Huang and Lee claim that their experiments show that their specification-based and statistical-based models detect most of the basic anomalies. Some results of their experiments are shown in the figure below. The figure presents the detection rate and false alarms for different routing attacks. For example, their method can detect the "flooding of data packets" routing attack with a detection rate of 92% and false alarms of 5%. Imagine that a node receive thousands of packets. Then 5% of false alarms means a lot of false alarms. Thus, their method really needs to be improved before it can be used in an intrusion detection tool. However, their method of detecting attacks based on specification-based approaches gives no false alarms (at least according to Huang and Lee), but the detection rate was still to low. The average detection rate was about 90% (varying from 59% to 100% depending on the attack).

Routing attacks	Detection rate	False alarms
Flooding of data packets	92%	5%
Flooding of routing messages	91%	9%
Modification of routing messages	79%	32%
Rushing of routing messages	88%	14%

Figure 10: Detection rates and false alarms rates of anomaly basic events.

9.3 Conclusion

Even though intrusion detection tools have been available for wired systems for long time, there are no products available for mobile ad hoc networks today. Section 9.1 presents an evaluation of ten wireless intrusion detection tools with regard to their suitability for tactical mobile ad hoc networks. These tools were evaluated according to proposed and identified requirements for intrusion detection systems in mobile ad hoc networks. Since the tools are mainly intended for corporate wireless networks, they do not meet the specific tactical requirements on intrusion detection systems in mobile ad hoc networks well. Note, that this does not mean that they are not useful for wireless 802.11 networks; since the evaluation did not aim at investigating their efficiency in wireless networks in general. However, to develop a tool for mobile ad hoc networks, more research is required:

1. An IDS architecture with better support for the autonomous and self-organizing properties of a tactical mobile ad hoc network is needed.

2. Methods that detect attacks relevant in the context of tactical mobile ad hoc networks are needed.

The research presented in section 9.2 shows that anomalies may be used for detecting routing attacks, but the methods really need to be refined. However, the specification-based approach performs better, since it is possible to get very few false alarms. Also note that the provided methods only were able to detect attacks against one routing protocol. A useful intrusion detection system must be able to detect attacks against many protocols and applications.

10 Authorization/access control

The function of *access control* is to control which *subjects* (processes, persons, machines, etc.) have access to which *objects* in the system – which files they can read, which data shall be shared and with whom, which system resources they can use, and so on. Thus, access control is the center of IT security! Obviously, there are many aspects of access control.

Access control is needed in any system, whether large or small, whether military or civilian. Without access control, you cannot achieve the goals of information security, that is to ensure confidentiality, integrity and availability. It is quite obvious that access control of information objects is necessary for confidentiality and integrity. However, access control of system resources is also needed to maintain availability of the system.

The rules for access control are described in a *policy*. A crucial part of any system for access control is the ability to formulate the policy in a way that both addresses the security needs and is possible to enforce by functions within the operating system, or within a system dedicated for access control. In a large, distributed system the policy tends to be complex. Complexity by itself is an enemy of security.

Access control works at a number of levels. In [15], four levels – application, middleware, operating system and hardware – are discussed. The following text summarizes the reasoning in [15].

> The users see the total system at the application level. The policy tends to be very rich and complex. The users might be organized in a number of groups and users might act in different roles, changing dynamically in time and location. Each role could initiate a number of possible access requests. Some of these might require a valid authorization by a third party, while others might be time critical and require response in real time.
>
> The applications may be written on top of middleware, such as a database management system, which enforces a number of protection properties. A database system typically has access control for transactions to and from the database.
>
> The middleware will use facilities provided by the underlying operating system. This typically controls access to files, communication ports etc.
>
> The operating system, finally, relies upon features provided by, for instance, memory management hardware, hardware for access to the radio channel, etc.

The functions of access control, and the policies, are not the same at all levels, but they must stick together, since a higher level relies upon functions in a lower level.

A rather general way to describe the flow in access control is depicted in figure 11. The *subject* is the active part, which requests some kind of access. Often, the request emanates from a person, so this person's identity is important in the access control. In addition, other identities are important, e.g. which processor is used, which program is requesting etc. The *activity* should describe the context of the request. One important activity is the role in which the subject acts – is it as a particular type of operator, is it as an administrator, is it as a data source etc? The *environment* should describe relevant external information, like time of day, geographical location etc.

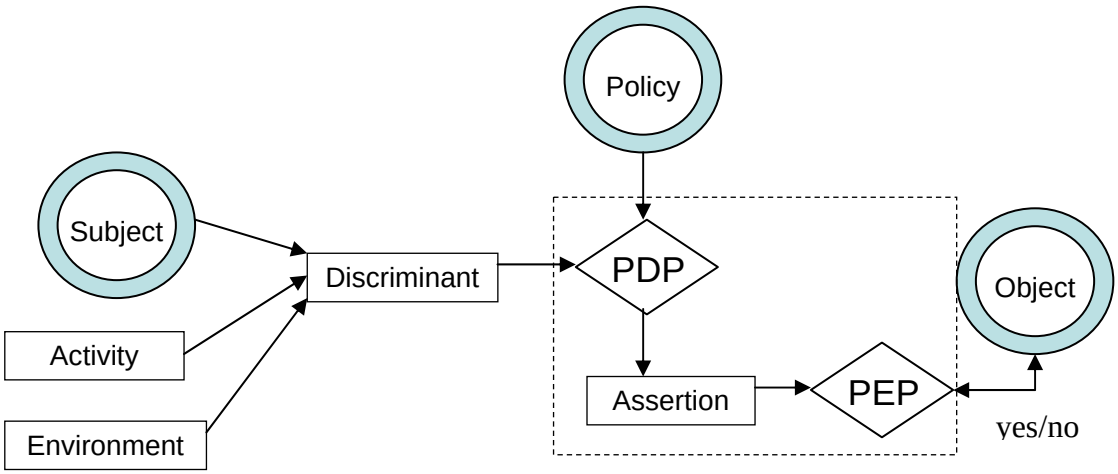


Figure 11: Flow in access

These inputs are put together into a *discriminant*, which can be thought of as a data structure suitable as input to the next step, the PDP, Policy Decision Point. The other input to the PDP is the rules in the *policy*. The policy is created and maintained by dedicated users, the security managers. It is obvious that the data in the discriminant and the rules in the policy must be expressed by the same vocabulary, for the PDP to be able to make a decision. The PDP performs a number of actions. It verifies the correctness of the input, e.g. verifies that the identities have been properly authenticated. It matches the rules of the policy with data in the discriminant. It might give a yes/no output, for request granted/denied. More generally, though, it creates an *assertion* as input to the next step, the PEP, Policy Enforcing Point. At the PEP the final yes/no result is compiled from the assertion and from data tied to the object, e.g. an access control list.

The dashed box around the PDP and the PEP means that they are closely related. In a standard operating system for a stand-alone computer, they might be thought of as the same point and not as two parts. However, in a distributed system they might be implemented in different nodes. The PDP might be a trusted authority, for example an authorization server. The assertion is then sent over the network to the PEP, perhaps in the form of a privilege certificate. The PEP then has to verify the correctness of the assertion.

There is a lot more to be said about access control. For example, what should happen when an access is granted? If it is a request to read some information object, should the object be sent back to the subject? In a distributed system, the object then must be protected, probably by encryption, which means that the system for access control must interact with the cryptographic system. As a general observation, access control has much in common with cryptography. Another observation is that the parts of the system for access control must be controlled themselves. The policy must be maintained by proper users only. The activity is a crucial part. How can for instance a subject change between different roles?

10.1.1 Basic access control

Figure 11 is a generic description of access control. The implementation of access control, at one of the four levels mentioned above, will more or less deviate from the description. At the operating system level, controlling a single computer (like an ad hoc radio node) a basic system for access control is described in any basic book on UNIX or Windows. The subjects are the different users, which should be able to run a process on the system. The policy consists of some data files maintained by a systems manager. In these files, every user is given an identity and each user is tied to some parameters that will be matched with a discriminant. Examples of parameters are password for authentication, allowed times of day etc. The policy also describes groups of users. The manager gives each group an identity and lists the members (users) of each group.

Each object that should be access controlled has tied to it an ACL, Access Control List. The ACL is created when the object is created by the object owner. The ACL can be modified later on, but this is a kind of access request that must be carefully controlled. The ACL lists which requests are permitted for different types of subjects. In basic UNIX there are only three entries – permissions for the object owner, permissions for a particular group of subjects (typically the group of users the object owner belonged to at creation time) and permissions for all other subjects, respectively. There are also basically only three permissions – grant/deny for read, write and execute, respectively. In different versions of Windows, emanating from Windows NT, there are more entries in the ACL and more types of permissions.

The discriminant in figure 11 is attached to the requesting process and in UNIX it contains identities for the process owner (typically authenticated at login time) and for a group which the process owner now belongs to. This group could be changed, by executing a command like *chgrp*, thereby emulating a change of activity. This is another example of a type of request that should be carefully controlled. In Windows, you find a similar discriminant, called the *access token*.

In both UNIX and Windows, the PDP and PEP are combined in some kind of security monitor. When a subject requests an access to an object, the discriminator is compared to the object's ACL. If the request is granted, the monitor gives the subject a handle to the object for the permitted operation.

This type of basic access control is well tested. Its main advantage is that it is rather simple, in its basic shape, but this also is a drawback. With only three entries in the ACL and only three types of permissions, you end up permitting too many subjects doing too much, a violation of the principle of least privilege. In extended UNIXes and Windows, the granularity is finer, but this adds complexity.

It is also hard to scale up to large systems. Creating ACLs for each object describing permissions for each group of users is not scalable. Another obvious drawback is that it is dependant on platform. Although similar, there are differences, which make it hard to mix different operating systems.

10.1.2 Role Based Access Control, RBAC.

The disadvantages mentioned lead to models for more scalable access control. In the basic access control described above the policy is centralized around users and groups of users. The permissions are scattered around in the ACLs for the objects. To find out what a particular group of users is allowed to do, you essentially have to analyze each object's ACL. This is certainly not scalable.

The idea, for getting better scalability, is to focus on permissions in the policy. Instead of groups and subgroups of subjects, the policy is described as sets of permissions, and relations between such sets. Subjects (typically users) are assigned to the sets. The combination of permissions, relations and subjects is called a role. The roles are meant to be managed, fairly statically, by a security manager. The dynamic aspects of the system (compare activity in figure) are modeled by what is called sessions. At runtime, a subject can establish sessions. Each session activates a subset of the roles that the subject is assigned to. The activations are controlled by rules and relations in the policy. In this way, it is possible to model different kinds of separation of duty relations. Examples are relations saying only one subject may act as operator_x at any time, no subject may simultaneously act as operator_y and as operator_z, etc. RBAC is a more natural model to describe large systems at a high level.

It is particularly NIST in USA that since the early 1990s has supported the development of RBAC. In [12] a framework for RBAC is described and [11] is a proposed standard for RBAC. The proposed standard describes what elements and functions should be in a system for role based access control. There are many products, both extended operating systems (like Trusted Solaris [6] or Windows Server 2003 [5]) and special identity management products, that claim to support RBAC. No product, however, does support RBAC fully. The proposed standard is therefore structured in four parts – core RBAC and three kinds of extensions to the core. The core RBAC describes the core – roles, assignment of subjects and permissions to roles, and establishment of sessions. The proposed standard argues that it is possible to get core RBAC in an operating system with basic group-based access control. One extension is hierarchical RBAC, where it is possible to define roles in the common object oriented way with inheritance from roles to sub roles. The next extension is SSD, static separation of duty. This allows constraints that are evaluated at role management time. The last extension is DSD, dynamic separation of duty, with constraints evaluated at run time.

10.1.3 Access Control in Mobile Ad-hoc Networks

So far, general aspects of access control have been described, applying to many types of distributed systems. Elsewhere, special characteristics of mobile ad hoc networks are stated. Two characteristics, that affect access control, are:

- The operating system in the mobile nodes shall mainly support mobility and communication. It is therefore likely focused on real time functions, rather than elaborated access control.

- An ad hoc network must not depend on a focal point, like a server, being accessible all the time. Ideally, there should be no focal points at all.

The first characteristic, the operating system, mainly affects the enforcing point, PEP, in the figure. It is most likely, that the operating system would support basic, group-based access control. To support RBAC, more properties from the discriminant must be available to the PEP.

This implies either an operating system at the level of Trusted Solaris or Windows 2003 Server, or that the properties can be transferred as assertions that can be evaluated at the PEP.

The other characteristic, the independence of focal points, is a fundamentally hard problem. No focal points at all means that, in the figure, both the policy itself, management of the policy, and the PDP, all are spread out in the nodes of the network. In [3] there is a discussion on the hard problem of how to establish trust among an ad hoc group of nodes and how to make distributed decisions. They advocate swarm intelligence as a practicable way, but much research remains to be done. One part of the problem is how the PEP can verify that the PDP consists of a special group of nodes. This can be solved by threshold cryptography, see for instance [2].

If you accept that the policy is managed centrally and that the PDP can be chosen within a group of particularly capable nodes, the problem is quite reduced. The assertions are then essentially certificates, as in a PKI. This means that the hardest problem is the withdrawal of assertions. This is like classical CRLs, Certificate Revocation Lists, but with much shorter time intervals. They also have to be distributed by some peer-to-peer protocol, not just stored at servers.

11 Additional security services

In this section, some additional important security services for mobile ad hoc networks are discussed. The following security services are discussed;

- User authentication, see section 15.1.
- Signed software, see section 15.2
- Storage encryption, see section 15.3
- Confidentiality, see section 15.4.
- Management, see section 15.5.

There is no need to discuss user authentication, signed software and storage encryption in detail in this report focusing on problems typical for mobile and ad hoc nodes, since these services are similar for wired networks. Neither is there a need to discuss confidentiality of data, since this service has been available in tactical radio networks for long time. Management is not discussed in detail due to lack of time. This can be consider as future work and is a distinct area within security.

11.1 User authentication

Authentication is a validation of a user's identity against previously stored information. Thus, a secure identity is needed to be able to perform authentication. This identity is often used by the access control system to ensure that the objects of the system are used by the right persons. The following is a list of commonly used identities [28]:

- The identity of a device or user can be the private key in an asymmetric key-pair
    - The keys can be stored in an active card (also called smart card)
    - The keys can be stored in soft certificates (e.g. PKCS#12, PEM)
- The identity can come from a key server
- The identity can also be a password (weak authentication)

11.2 Code signing

The purpose of *code signing* is to make sure that a piece of code comes from a trusted party and has not been changed since it was signed.

First, the trusted party uses a *cryptographic hash function* to calculate a *hash value* from the code. In the case of code signing, the most important requirement for a cryptographic hash function is *pre-image resistance*. This means that it should not be computationally feasible to find another piece of code with the same hash value as the original one.

The hash value is *asymmetrically encrypted* using a *private key* owned by the trusted party. The resulting cipher text is called the *signature*. The private key should be stored and used only in a tightly controlled off-line system to ensure that it does not get compromised.

The same code and signature can be sent together to all nodes that require the code. When such a node is to use the code, it calculates a hash value from the received code. It also decrypts the signature using the *public key* of the trusted party. Finally it compares the hash values to make sure that the code indeed comes from the trusted party and that it has not changed since it was signed.

The public key of the trusted party must be available to a node when it is about to check a signature. A *public key certificate* contains, among other things, the public key and the identity of the key owner, as well as a signature of these calculated by a trusted third party. In this way, the node can for example be sent the public key together with the signed code. To be able to verify the correctness of a certificate, the node must possess at least one public key (that of the trusted third party) from the very beginning.

A certificate should also contain an expiration date, and it should be possible to *revoke* a certificate if needed. Revocation information should be made available to the node in some way without the need of a central server, for example by spreading it throughout the network at a regular interval.

Cryptographic algorithms

The most popular cryptographic hash functions are MD5 and SHA-1. At present, neither MD5 nor SHA-1 can probably be considered the strongest available algorithm. The algorithms that currently seem to be most reliable and probably will be the most commonly used in the foreseeable future are SHA-224, SHA-256, SHA-384 and SHA-512. These are improved versions of the original SHA-1 algorithm. Their capability of pre-image resistance has however not been mathematically proven.

Some popular asymmetric algorithms for creating digital signatures are RSA, DSA (Digital Signature Algorithm), and ElGamal. RSA presumably get its security from the difficulty of factoring large numbers, but this has not been mathematically proven. DSA and ElGamal presumably get their security from the difficulty of calculating discrete logarithms over a finite field, but this has not been mathematically proven either. Furthermore, neither factoring large numbers, nor calculating discrete logarithms over a finite field, are mathematically proven to be as hard problems as they seem to be today.

Cryptographic systems

Many basic cryptographic algorithms, like for example RSA, are not secure when used in the “textbook” way. A whole *cryptographic system* must be designed around an algorithm, where everything from security enhancing padding to handling of insecure special cases is taken into account. It cannot be stressed enough that designing a cryptographic system is no easy task - even when the software implementation has not yet been taken into account. It is completely possible to build a fully functional cryptographic system using the very best cryptographic algorithms and yet end up with a completely insecure final product.

Using a standard cryptographic system that has been evaluated by many experts for a long time is one way to minimize the risks. If the cryptographic system must be designed from scratch, it is

very important that the designers possess a much higher competence level than what is required to design and implement something functional built on standard algorithms.

Modularity and cryptographic system design

No algorithm can be expected to be secure enough indefinitely. Therefore the cryptographic system and its software implementation should be designed in a modular way such that the selected algorithms can be exchanged for new ones should the need arise. When designing such a modular solution it is important to specify the exact requirements for the algorithms involved, seen from the perspective of the cryptosystem as a whole. These requirements should be documented explicitly for future reference.

Software implementation

Implementing a cryptographic system in software adds even more security risks. For example, a slight mistake in the implementation can make the final result, from a cryptographic point of view, an entirely different cryptographic system than the one intended to be built.

Using a standard cryptosystem has benefits in this aspect since interoperability testing between different implementations of standard cryptographic systems sometimes can reveal such implementation mistakes.

What to sign

It is important that all code is signed completely, including any metadata describing the code, and so on. It is not as trivial a problem as it might seem to determine which parts really need to be signed and which do not. Therefore the above recommendation should be followed if at all possible.

Porting aspects

Since it is not clear exactly which operating system will be run on the nodes, it is hard to give specific advice regarding standard solutions for code signing. One example of a standard solution is *authenticode*, but this assumes the operation system to be Windows. If the operating system used is of a less common kind, it might become necessary to port an existing solution. It is important to understand that porting a cryptographic system implementation can be quite a tricky task. Code that seems usable as-is on the new platform might in fact give rise to new flaws in combination with the new underlying platform.

11.3 Confidentiality of communication

Confidentiality is in this report only briefly described since confidentiality is a security service that is implemented in tactical radio networks today. Consequently, there are many encryption solutions developed for military use. SwaF specify both secret symmetric and public-key encryption algorithms for different types of classification levels [30]. The developed symmetric encryption algorithms can be classified up to the level of top-secret, whereas the public-key encryption algorithms can be classified up to the level of restricted [30].

Data confidentiality is the protection of transmitted data from passive attacks such as eavesdropping. Encryption at the data link layer with symmetric group keys provides some type of group authentication, but not data origin authentication (se definition in section 2) or entity

authentication (se definition in section 2). Below is table that describes what types of attacks confidentiality protects against.

Group of attacks	Data link layer encryption with group keys
Traffic analysis	Not totally
Eavesdropping of header data	Yes
Eavesdropping of data	Yes
Masqurade	No
Replay	No
Modification of message	No
Denial of service	No
Internal attacks	No

Figure 12: Protection given by data link layer encryption against certain types of attacks

11.4 Storage confidentiality

Mobile nodes need to save system security information (e.g. user account information, firewall rules, authorization information) local in the node. The node will also include other sensitive data such as logs of events and incidents. This information need to be protected from unauthorized disclosure. This is achieved with encryption of information. SwaF include symmetric encryption algorithms that can be used for classification level up to top-secret as well as asymmetric algorithms that can be classified up to the level of restricted [30]. Also, note that there are many other important security considerations apart from the chosen encryption algorithms. For example, the same key should only be used for a certain amount of data.

The different types of mechanisms used to implement storage confidentiality are;

- Physical hardware drive encryption
- Hardware encryption with software
- Partition encryption with software
- File encryption

11.5 Security Management

Security management includes methods to enforce:

- secure configuration of nodes,
- security policy management,
    - e.g. distribute authorization information,
- encryption key management,
- incident management,
    - discover intrusions, managing security weaknesses, and respond to intrusions or threats, and

- supervision of nodes and network

Management is not discussed in detail due to lack of time. This can be consider as future work and is a distinct area within security.

12 References

1 E. Hansson, “Säkerhetsproblem i mobila ad hoc-nät”, FOI-R-1633, Maj 2005

2 Nitesh Saxena et al, “Access Control in Ad Hoc Groups”, Proc of the 2004 International Workshop on Hot Topics in Peer-to-Peer Systems (HOT-P2P’04)

3 Laurent Eschenauer et al, “On Trust Establishment in Mobile Ad-Hoc Networks”, Proc of the Security Protocols Workshop, April 2002

4 C. Perkins, E. Belding-Royer, S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561; July 2003

5 Mohan Rao Cavale, “Role-Based Access Control Using Windows Server 2003 Authorization Manager”, Microsoft Cooperation, Jan 2003, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetserv/html/AzManRoles.asp?frame=true&hidetoc=true

6 Sun Microsystems, White paper “RBAC in the Solaris Operating Environment”, 2001-04-27, http://www.sun.com/software/whitepapers/wp-rbac/wp-rbac.pdf

7 R. Blom, “An optimal class of symmetric key Generation System”, in advanced on Cryptology - Eurocrypt’84, LNCS vol. 209, p. 335-338, 1985.

8 D. Balfanz, D.K. Smetters, P. Stewart, and H. Chi Wong. Talking to Strangers: Authentication in Ad-Hoc Wireless Networks, Proceedings of Network and Distributed System Security, Symposium 2002 (NDSS '02), 2002.

9 S. ·Capkun, J.-P. Hubaux, and L. Buttyan. Self-Organized Public-Key Management for Mobile Ad Hoc Networks, IEEE Transactions on Mobile Computing, January-March 2003.

10 National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), 2003

11 David Ferraiolo et al, “Proposed NIST Standard for Role-Based Access Control”, ACM Transactions on Information and System Security, Vol 4, No 3, Aug 2001, pp 224-274

12 Ravi Sandhu et al, “Role-Based Access Control Models”, IEEE Computer, Vol 29, No 2, Feb 1996, pp 38-47

13 S. Gokhale and P. Dasgupta. Distributed Authentication for Peer-to-Peer Networks, Sympo-sium on Applications and the Internet Workshops 2003 (SAINT'03 Workshops), IEEE Computer Society 2003, ISBN 0-7695-1873-7, 2003, pp. 347-353.

14 J.-P. Hubaux, L. Buttyan, and S. ·Capkun. The Quest for Security in Mobile Ad Hoc Networks, ACM Symposium on Mobile Ad Hoc and Computing, MobiHOC 2001, 2001, pp. 146-155.

15 Ross Anderson, “Security Engineering: A Guide to Building Dependable Distributed Systems”, John Wiley & Sons, 2001.

16 J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang. Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks, International Conference on Network Protocols (ICNP) 2001, 2001.

17 L. Lamport. Password authentication with insecure communication, Communication of the ACM, vol. 24, no. 11, 1981, pp. 770-772.

18 H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang. Self-Securing Ad Hoc Wireless Networks, Seventh IEEE Symposium on Computers and Communications (ISCC '02), 2002.

19 Y. Huang and W. Lee, “Attack Analysis and Detection for Ad Hoc Routing Protocols, “ RAID 2004, 2004.

20 A. Weimerskirch and D. Westho. Zero Common-Knowledge Authentication for Pervasive Networks, Tenth Annual International Workshop on Selected Areas in Cryptography (SAC 2003), 2003.

21 L. Zhou and Z.J. Haas. Securing Ad Hoc Networks, IEEE Network Journal, vol. 13, no. 6, 1999, pp. 24-30.

22 L. Venkatraman and D. P. Agrawal, “A novel Authentication scheme for Ad hoc Networks”, IEEE 2000.

23 S. Zhu et al, “LHAP: A lightweight Hop-by-Hop Authentication Protocol for Ad-Hoc Networks, Proceedings of the 23 rd International Conference on Distributed Computing Systems Workshop (ICDCSW´03).

24 Y. Zhang and W. Lee, “Intrusion Detection in Wireless Ad Hoc Networks,” 6th Int´l Conf. Mobile Comp. and Net., Aug. 200, pp. 275-83.

25 A. Weimerskirch and G. Thonet, A Distributed Light-Weight Authentication Model for Ad-hoc Networks.

26 Adrian Perrig, Ran Canetti, Dawn Song, and J. D. Tygar. Efficient and Secure Source Authentication for Multicast. In Network and Distributed System Security Symposium, NDSS ’01, pages 35–46, February 2001.

27 Adrian Perrig, Ran Canetti, J.D. Tygar, and Dawn Song. Efficient Authentication and Signing of Multicast Streams over Lossy Channels. In IEEE Symposium on Security and Privacy, pages 56–73, May 2000.

28 LedsystT JV, Daniel Arvidsson, “Authentication and privilege control”, SENI 04-0411, rev 0.2 2004-08-30.

29 LedsystT, FMV/JV, “FMLS2010 Key Management Overview”, rev 2.1, 2005-04-04

30 B. Nilsson and S. Burström, “Security architecture overview”, rev 2.0, LT1K P04-0385, 2005-04-07.

31 http://www.wi-fi.com/OpenSection/pdf/Wi-Fi_Protected_Access_Overview .pdf.

32 S. Wong, The evolution of wireless security in 802.11 networks: WEP, WPA and 802.11 standards, May 20, 2003.

33 www.deadhat.com/wlancrypto

34 http://snort-wireless.org/

35 http://www.networkchemistry.com/products/it_security.php

36 Moskowitz, R.; Host Identity Protocol; Internet draft; June 2005 (http://www.ietf.org/internet-drafts/draft-ietf-hip-base-03.txt)

37 Moskowitz, R.; Host Identity Protocol Architecture; Internet draft; January 2004, (http://www.ietf.org/internet-drafts/draft-ietf-hip-arch-02.txt)

38 Rigney et al, Remote Authentication Dial In User Service (RADIUS), www.ietf.org, RFC 2865, June 2000

39 Calhoun et al, Diameter Base Protocol, www.ietf.org, RFC 3588, September 2003

40 J Kohl, and B, Neuman, “The Kerberos Network Authentication service”, RFC 1510, Seotember 1993

41 B. DeCleene et al, “Secure Group Communications for wireless networks”, IEEE Milcom01, Oct. 2001.

42 S.P. Griffin et al “Hierarchical Key Management for mobile Multicast Members”, 2002.

43 http://www.red-m.com

44 http://www.arubanetworks.com/products/airos/ids-fs/

45 http://reviews.infoworld.com/article/04/04/02/14TCwids_1.html

46 R. Blom, “An optimla Class of symmetric key generation System,” Eurocrypt´84, 1985

47 T. Matsuoto, and H. Imai. “ On the key predistributed systems: A practical solution to the key distribution problem”, Crypto’87, 1988.

48 W. Stallings, “Network security essentials”, ISBN 0-13-035128-8, 2003

49 A. Shamir. How to share a secret. Communication of the ACM. 1979

50 S. Yi and R. Kravets, Key Management for Heterogenous Ad Hoc Wireless Networks, July 2002.

51 L. Eschenauer and V.D. Gligor, “ A key management scheme for distributed sensor networks”, in preceedings of ACM CCS´02, Nov 2002, pp. 41-47.

52 K. Rhee et al, “A group Key management Architecture for Mobile Ad-hoc Wireless Networks, 3 International Workshop on Information Security Application (WISA 2002) , August 2002.

53 H.A. Schotanous et al, Security aspects of mobile and ad hoc networks, FOI-R-1161, Feb 2004.

54 R. Canetti, J. Garay, G. Itkis, D. Miccianicio, M. Naor, and B. Pinkas. Multicast security: A taxonomie and some efficient constructions. InProceedings of IEEE INFOCOM '99, New York, USA, March 1999.

55 A.A. Pirzada and C. McDonald, "Kerberos Assistant Authentication in Mobile Ad-hoc networks", 27th Australasian Computer Science Conference, 2004.

56 P. Papadimitratos and Z.J. Haas, "Secure Routing for Mobile Ad Hoc Networks," SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002), San Antonio, TX, January 27-31, 2002

57 M. Zapta and N. Asokan, "securing Ad Hoc Routing Protocols," ACM WiSe, 2002..

58 B. Dahillet al, "A secure protocol for Ad Hoc Networks", IEEE ICNP, 2002.

59 Y. Hu, D. Johnson, and A. Perrig, "Sead: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks," IEEE WMCSA, 2002.

60 Y. Hu, A. Perrig, and D. Johnson, "Ariadne: A secure on-demand Routing Protocol for Ad Hoc Networks, "ACM MOBICOM, 2002.

61 S. Bouam and J. Ben-Othman, " Data Security in Ad hoc Networks Using Multipath Routing", http://www.prism.uvsq.fr/users/sbouam/Bouam_PIMRC_Final_Paper.pdf

62 P. Papadimitratos and Z.J. Haas, "Secure Message Transmission in Mobile Ad Hoc Networks," http://www.cs.huji.ac.il/labs/danss/sensor/adhoc/routing/papadimitratos_2003securemessage.pdf

63 H. Yang, "Security in Mobile Ad Hoc Networks: Challenges and Solutions", IEEE Wireless Communications, February 2004.

64 S. M. Bellovin, "Distributed Firewalls", http://www.cs.columbia.edu/~smb/papers/distfw.pdf

65 R. Heady, G. Luger, A. Maccabe, and M. Servilla. "The architecture of a network level intrusion detection system. Technical report", Computer science Department, University of New Mexico, August 1990.

66 S. Axelsson, "Intrusion Detection Systems: A taxonomy and Survey", Tech. Report no. 99-15, Dept. of Comp. Eng., Chalmers Univ. of Technology, Sweden, Mar. 20, 2003.

67 Y. Zhang, W. Lee. Intrusion Detection in Wireless Ad hoc Networks. Proceedings of MOBICOM. Pages: 275-283. ACM. 2000.

68 E. Hansson and A. Hansson "Evaluation of wireless Intrusion Detection tools for Mobile Ad Hoc Networks", FOI-R-1374, November 2004.

69 http://www.airdefense.net/products/features/security.html

70 http://www.airespace.com/pdf/airespace_wireless_protection_system_ds.pdf

71 http://www.airmagnet.com/scripts/whitepapers.php, see the AirMagnet impact document.

72 http://subscriber.acumeninfo.com/uploads2/2/C/2C61E212894BEFB30971637BAA278D64/1081976778096/SOURCE/4729airmagnet.html, Network Computing, Watching the Waves, AirMagnet Distributed v4 review

73 http://www.vigilantminds.com/site/files/airxone.pdf

74 http://www.wi-fiplanet.com/tutorials/article.php/3099791

75 Y. Zhang, W. Lee and Y. Huang. *Intrusion Detection Techniques for Mobile Wireless Networks*. In *Report on a Working Session on Security in Wireless Ad Hoc Networks*. L. Buttyán, J-P Hubaux (eds). Mobile Computing and Communications Review. Vol. 7, No. 1. Pages: 74-94. ACM. January, 2003.

76 S. Marti et al., "Mitigating Routing Misbehaviour in Mobile Ad Hoc Networks," Proc. 6th Annual Int´l Conf. Mobile Comp. and Net., Boston, MA, pp. 255-65

77 Johnson, D.B., et al.; The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR); Internet draft; February 2002.

78 Josh Broch et al. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*, pages 85–97, October 1998.

13 Revision history

Date	Revision	Changes	Signature	CCB Status
2005-05-26	0.1	First version	FOI/ELHA	
2005-08-25	0.9	Second version	FOI/ELHA	
2005-08-31	1.0	Review version	FOI/ELHA	