Mobile IP Working Group
INTERNET-DRAFT

Vesselin Tzvetkov
Erika Sanchez
Sheffield University
15 September 2000

Mobile Virtual Private Network

<draft-tzvetkov-mvpn-01.txt>

Status of this Memo

Abstract

Mobile networks present several challenges and problems that current security mechanisms did not take in consideration in their specification, even though they continue to be a very good solutions for fixed networks, they can undermine the performance of mobile environments.

Current security mechanisms offer enough protection to fixed networks for the establishment of secure communications between nodes. Security protocols such as IPsec offer a strong and secure solution for the creation of VPNs and its functionality has made it very popular in the networking community, but for Mobile VPN such solutions are not optimal.

This document specifies a protocol that uses the basis of encryption and authentication for the establishment of Virtual Private Networks on mobile environments using Mobile IP Protocol (RFC2002) and providing a secure, fast and optimised mobile communication.

Table of Contents

Status of this Memo 1
Abstract 1
Table of Contents 2
1. Introduction 4
  1.2 Comparison with Build with IPsec Mobile Virtual Private Networks 5
2. Terminology 7
  2.1. General Terms 7
  2.2 Specification Language 8
3. Overview of Mobile Virtual Private Network 9
  3.1 Requirements for Mobile Virtual Private Network. 9
  3.2. Basic Operation 10
  3.3. Structure of MVPN 12
  3.4 Mobile Virtual Private Network 15
    3.4.1. First Contact 15
    3.4.2. Re-registration of Mobile Node 22
  3.5. Changing the Foreign Agent 22
  3.6. Session key generation 23
  3.7 Mobile data flow 29
4 TCP in TCP 30
5. Error messages 31
6. Messages structure 32
  6.1 C notation 32
    6.1.1 Vectors 32
    6.1.2 Numbers 32
    6.1.3 Enumerateds 32
    6.1.4 Constructed types 33
    6.1.5 Variants 33
  6.2 Messages in the 'first contact' phase 34
    6.2.1 Registration request form MN. Extension Structure 34
    6.2.2 Foreign Agent Information Extension. 35
    6.2.2 Home Agent Information Extension 35
    6.2.3 Foreign Reply Extension 37
  6.3 Session key generation 37
    6.3.1 Request Communication Message 37
    6.3.2 Correspondent Node Information Message 38
    6.3.3 FA Retriving Message 39
    6.3.4 Mobile Respond 39
    6.3.5 Communication Reply Message 40
7. Mode of operation and encryption methods 40
  7.1 Nodes authentication procedure 40
    7.1.1 Encryption Algorithm for the PKI structure 40
    7.1.2 Mode of Operation 41
    7.1.3 Packet Authentication 41
  7.2 Diffie-Hellman key exchange 42

7.3 Encryption algorithms and modes of operation , Mac building in “data flow protection” phase 44
8. Conclusion 45
9. References..47

1. Introduction

The IETF RFC 2002 [1] defines the possibility for a node to change its point of attachment without losing its ability to communicate. The introduction of mobile entities provides new concerns especially for the area of security having as principal concern the registration of a Mobile Node. The Mobile Node and other mobile entities MUST be able to provide information about their identity to other parties in order to be able to trust and validate the information exchanged between them which is the goal of this protocol.

The second goal is to achieve a flexible protocol for building MVPN which can be used in the modern and the future Network; friendly to Network Address Transmissions and **Network Monitoring**, **packet filtering** etc. Some of the security protocols like IPsec don't work with techniques typical for enhanced proxies as: optimising the TCP connection (gateway for optimisation ”**TCP session splitting**” for **GEO satellites**), Network Address Transmission (**NAT**), Network Monitoring etc. The new protocol should be friendly and transparent to these techniques.

As mentioned earlier the basis of a Virtual Private Network relies on the use of cryptographic mechanisms such as encryption, message digests, etc. The protocol presented in this document seeks the best performance for mobile environments without affecting the expected levels of security, MVPN provides strong authentication and fast re-registration, and can be applicable to current network systems, mobile or fixed.

MVPN is based on a Public Key Infrastructure, which offers a secure environment for authentication and session key distribution, it also improves the security and performance of the system with the inclusion of a third party called Trust Centre .

Building MVPN (Mobile Virtual Private Networks) don’t adds new TCP/UDP packets transmissions then the one of Mobile IP.

This protocol is optimised for:

- small number of messages
- compatible with Mobile IP protocol
- high security high quality of Mobile Virtual Private Network
- Fast data transmission. After authentication procedure
- Use of Smart cards in the mobile devices.
- Practical use

We consider that the **Smart cards** (esp. High-End Smart Card) are very suitable for mobile devices and we recommend them for building MVPN.

1.2 Comparison with Build with IPsec Mobile Virtual Private Networks

Until now, Virtual Private Networks (VPNs) have proved to be a convenient solution for the establishment of secure connections between private nodes throughout a public network. The creation of such VPNs requires the use of special techniques such as IPsec [2],

The application of such techniques in a mobile environment can provide a secure solution but the performance of the network will be affected. These techniques were created to work over **fixed networks** and since mobile networks differ in certain characteristics, it is necessary to find better solutions. For example: the firewall filtering oft filters of on inner port coming packets with a network address out of the local network. Using Mobile IP this situation is possible and legal, but in the “statical world” this is treated as an attacks.

IPsec adds extra and unnecessary overhead to packets that are short and it is very strict in the use of its services and modes, which makes it difficult to be optimised for Mobile IP.

IPsec is also not optimised **for wireless case** ,where the number of packets should be kept as low as possible. In MVPN with MIP there is **no additional packets** during the authentication phase, which means *saving of resources*.

IPsec also interferes with other mechanisms such as the ones implemented by Quality of Service (**QoS**) by using end to end encapsulation, which includes the TCP header [5].

Building Mobile Virtual Private Networks requires the consideration of new parameters and the use of conventional methods such as IPsec can arise new big problems, for example the main two possibilities for MVPN with IPsec :

- If an **'end to end'** protection with IPsec is used, the foreign and Home Agents will not be able to authenticate the data flow between the mobile and correspondent nodes. In this case the foreign/Home Agent network will be vulnerable since the firewall protecting the Foreign Agent cannot perform packet authentication and packet filtering due to the fact that all the information preceding the IPsec header is encrypted. (ESP+ authentication). The level of security for home/Foreign Agent **is not acceptable**, since filtering is one of the most used protection for local networks from MAN/WAN.

- If a **system of security tunnels** is used, for example tunnels from: 'CN to HA', 'HA to FA' and 'FA to MN', the connection will not be optimised for performance. IPsec will allow the creation of such tunnelling system but the management of security channels, keys and packets will affect the performance of the MVPN. At the end there will be **3 security channel,** for example type ESP + authentication, which requires 6 session keys. In this case for sending one packet **6 times encryption/decryption** and **3 times authentication** are needed.

Method	***IPsec "Gateway to Gateway"***	***IPsec "End user to End user"***	***MVPN on MIP***
Secure tunnels	3 to 4	1	1
Security of HA Network (Firewall filtering)	yes	no	yes
Security of FA Network (Firewall filtering)	yes	no	yes
Packet transmissions for establishing the secure tunnels	9 to 24	3 to 6	4 to 5
Secret Keys	12 to 16	4	2
Key establishment using MIP registration messages	no	no	yes
Friendly to TCP/UDP based techniques (NAT, TCP session splitting ,Filtering ,Monitoring)	no	no	yes
Flexible to be used for other purposes then mobile VPN based on MIP	yes	yes	no

Table 1

In comparison the suggested MVPN with MIP requires for sending one packet through private network **3 authentications** and **2 encryptions / decryption**, which means *saving power*. All this is summarised on Table 1

Schneier and Ferguson performed an analysis of IPsec [6] and described several concerns about the security provided by IPsec. Even though is application to Mobile IP is possible [7] and in some level effective it does not allow Mobile IP to offer its best performance.

In comparison, MVPN requires the use of encryption and other cryptographic mechanisms as well as the inclusion of additional headers in the registration request and registration reply messages of the standard Mobile IP protocol. During the same registration process all entities involved in it will receive session keys for authentication and encryption that will enable them to perform a faster communication and negotiation for future registrations. MVPN uses the same number of messages required for the registration process hence it only adds processing time for the encryption and decryption operations.

We believe that *there are not universal solutions for every case,* it is not depending how powerful is the method . Even though MVPNs do not pretend to establish a universal solution, that can be used in any environment that requires authentication, integrity and confidentiality, by specifying the principles on which the MVPN should work.

MVPN offers flexibility in the selection of the encryption/decryption algorithm and key size, it is possible to make it compatible with existing mechanisms and have the same features as IPsec VPNs. IPsec can be used in any environment and will offer authentication, confidentiality and integrity mechanisms but it will involve a more complex administration and/or pure performance for a mobile environment which makes it less suitable as an appropriate solution.

2. Terminology

2.1. General Terms

IP	Internet Protocol.
MVPN-	Mobile Virtual Private Network.
Node	A device that implements IP.
Public Key	Key used in Public Key Infrastructures available for the public.
Private Key	Key used in Public Key Infrastructures as private.
$Ku_A(\ )$	Public key of node A.
$Kr_A(\ )$	Private key of node A.
$E_{KuA}(\ )$	Encryption using the public key of node A.
$E_{KrA}(\ )$	Encryption using the public key of node A.

DSS	Digital Signature Standard.
HD	Diffie-Hellman Exchange Algorithm
MIP	Mobile IP

Security association (SA) - Simplex connection that affords security services to the traffic carried by it.

$Z = X \| Y$ The bits of X and Y are put spoilt. The size of Z is equal to the sum of X’s size plus Y’s size. The Most significant bit (MSB) of X is the MSB of Z , Lest significant bit of Y (LSB) is the LSB of Z .

Terms specific to Mobile IP can be found in RFC 2002.

2.2 Specification Language

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", ”SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [3].

3. Overview of Mobile Virtual Private Network

3.1 Requirements for Mobile Virtual Private Network.

To ensure that the registration of a Mobile Node with its Home Agent is made under secure conditions (authenticity, integrity and confidentiality) it will be necessary to:

- Authenticate the Mobile Node with its Home Agent for the registration of the Mobile Nodes' care-of address and provision of services paid by the Mobile Node.
- Authenticate the Home Agent with the Mobile Node to ensure that a valid Home Agent registers the care-of address and provides the services required by the Mobile Node.
- Authenticate the Foreign Agent with the Home Agent to ensure that a valid Foreign Agent provides the required services.
- Authenticate the Home Agent with the Foreign Agent for the same reasons mentioned above. This authentication can be use for billing purposes.
- Authenticate the Mobile Node with the Foreign Agent to ensure that the care-of address is assigned to an authorised and valid Mobile Node.
- Authenticate the Foreign Agent with the Mobile Node to ensure that the care-of address assigned to the Mobile Node is valid.

The structure of MVPN is shown on the figure 1.

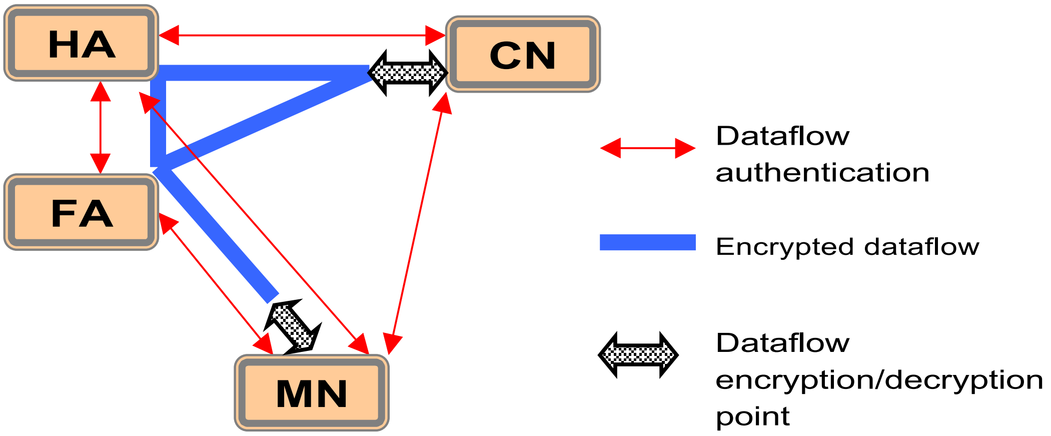


Figure 1

Even when Mobile IP does not require the establishment of a security association between the correspondent and Mobile Nodes it is necessary to perform a mutual authentication to provide confidentiality and integrity in the information exchanged between them, this will not only protect the nodes but also the home and foreign networks. The model presented in this document works under the assumption that a company can have several private networks spread around the world, and they can be visited by every node that is registered with them (locally or externally).

This model requires each correspondent node to provide information that enables the mobile entities (Home Agent, Foreign Agent and Mobile Node) to authenticate it. This requirement is NOT mandatory for all applications MPVN might have but it should be considered as an optional feature at implementation time.

The Dataflow should be encrypted and it MUST be possible to be decrypted only from the End Points in the MVPN and namely MN and CN. (see figure 1.). The data flow should also be authenticated in all nodes

3.2. Basic Operation

Before a Mobile Node can start communicating with any correspondent node, it is necessary for it to obtain a care-of address from an available Foreign Agent, and then register it with its Home Agent.

To avoid any possible intrusions within the networks it is required to authenticate all the messages exchanged during the registration process (registration request and registration reply messages), and then establish session keys. In the model described here the session keys include an authentication and an encryption key, that will be used during subsequent communications that any of the mobile agents may have with the other parties.

The authentication of a node requires that a shared secret, referenced by a security association, is already shared between the authenticating parties. In many occasions this shared secret is exchanged between them by a process called handshake or negotiation.

MVPN suggests the use of a Public Key Infrastructure (PKI), which requires every mobile entity to have a Private (Kr) and Public (Ku) key registered with a trusted third entity. The generation and distribution of these keys will be explained in following sections.

With the use of a PKI every entity will be able to authenticate and exchange session keys (shared secrets) between them in a very secure manner. A third entity known as **Trust Centre** (Certification authority) will be in charge of the registration of the public keys created by the Home Agent and their distribution to other nodes.

If a node wishes to contact another one, it needs to query the Trust Centre for the public key of the node. This implies that every node will know the public key of the Trust Centre and will be able to contact it in a secure way.

MVPN work under the following assumptions:

- The MN and the CN Mobile Node should have a **SMART CARD** that contains its Private Key. It's possible to use MVPN without using smart card than the calculation should be made in other way. The smart cards are very comfortable for the mobile security purposes. They are widespread in GSM mobile communication (Sim cards).They will play further a significant role in the future. We recommend to use smart card and especially **hight-end smart cards.**
- Every Mobile Node MUST know the Public Key of its Home Agent.
- Every node that belongs to a domain MUST know the Public Key of the Trust Centre available for that domain.
- Every node MUST have a unique Identification that SHOULD be created and distributed by the Home Agent.
- Every Public Key MUST be registered with the Trust Centre and MUST only be retrieved by providing a mean for its authentication.
- A node MUST only obtain the Public Key of another node by using the Trust Centre services. The Trust Centre MAY act as a broker for the retrieval of Public Keys of nodes that belong to different domains.
- The home and Foreign Agents MAY keep a list of Public Keys that belong to specific Mobile Nodes. This is to be used in case the Trust Centre is not available or to accelerate the authentication process.
- The keys MUST be refreshed every certain period of time to reinforce the security of the system.
- Every node MUST have the capability of performing cryptographic calculations such as encryption.

To store the public keys, the validity period of the keys and all information connected to the public keys should use the Directory Information certificates **x509v3** recommended form CCITT. These certificates are well known and described in the CCITT recommendation. They are wildly used from Certification Authorities (CA) and important part of SSL/TSL and all modern authentication protocols. The Nodes should store the keys using this certificates. This document is concentrated on the use of the keys not on the distributing of the **x509v3 certificates** from CA to the nodes, which relate to PKI structure.

The authentication information will be carried as extensions of the Mobile IP protocol. The scheme presented here only establishes the basic principles by which a PKI authentication process should be created in a mobile environment hence, it will be possible to create compatible security mechanisms that work under the same basis.

To provide a small level of protection the Foreign Agent advertisement message sent by the Foreign Agent will be linked to the Registration Request. After that, the exchange of registration messages will be the same as the one indicated in the Mobile IP protocol specification.

The protocol will be composed by two sets of messages. The first will be known as **'First Contact'**, and is carried out when the Mobile Node visits a foreign domain for the first time. The second set of messages belongs to the **'re-registration'** phase, and is performed when the lifetime of the previous Registration Request is about to expire, during this phase the mobile agents already know their Public Keys, the Public Key of the Mobile Node and already share session keys (authentication/encryption) with the Mobile Node, these infers that less information will be exchanged for the re-registration of the Mobile Node and will be not necessary to contact the Trust Centre unless it is required (keys expired).

The messages transmitted on a public network, in the Mobile IP case the registration request transmitted from the Foreign Agent to the Home Agent, will require a stronger level of security. *It is our belief that security should be scaled according to the risks presented in different scenarios*; since the information is travelling on a public network it is necessary to provide a stronger security mechanism. After completing the registration process the Mobile Node is now able to receive information from any authorised correspondent node.

Even though the Mobile Node is ready for receiving/sending information the correspondent node will need to complete a similar procedure for its authentication. Before initiating any communication, the correspondent node needs to authenticate itself with the Mobile Node. This will be performed with the help of the home and Foreign Agents and will also serve for the distribution and establishment of a session key to be shared between the correspondent and Mobile Nodes using the Diffie-Hellman algorithm.

The mobile entities and the correspondent node MUST be able to negotiate or indicate the type of cryptographic mechanism to implement for encryption and message digest calculation, this implies that the length of the keys will be variable and should be specified in the extension on which it is included.

3.3. Structure of MVPN

For the authentication procedure is used the possibility in MIP to send extensions in the ‘UDP registration messages’. They are added to the main datagramms. The structure of the extension is specified in RFC2002 and is shown in figure 2.

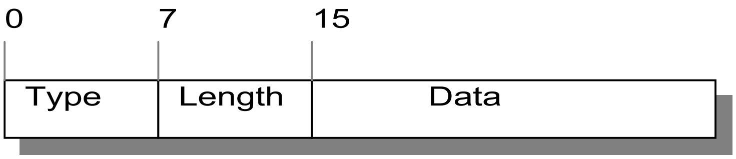


figure 2

In this way all need messages for authentication are sent in the standard registration messages for MIP. MIP needs 4 messages to register the Mobile Node . In these 4 messages with the use of extensions the Mobile Node is authenticated to Foreign Agent and Home Agent. The Foreign Agent is also authenticated to the Mobile Node using the Trust in Home Agent. Well know technique form **AAA** servers. In this registration procedure are also exchanged information for generation of secret key for protection of the all further data transmission between the parities.

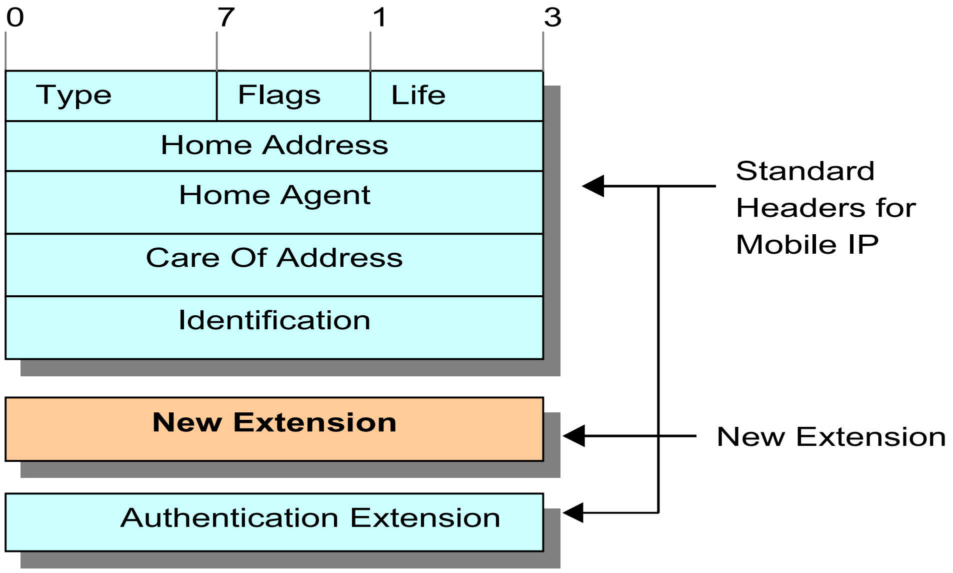


figure 3

The structure of the messages sent to register the Mobile Node (MN) are shown in figure 3. The diagram (figure 3) shows where the extensions need to be included in the Registration Request message and corresponding in all registration messages.

The messages are shown on the following diagram (figure 4) :

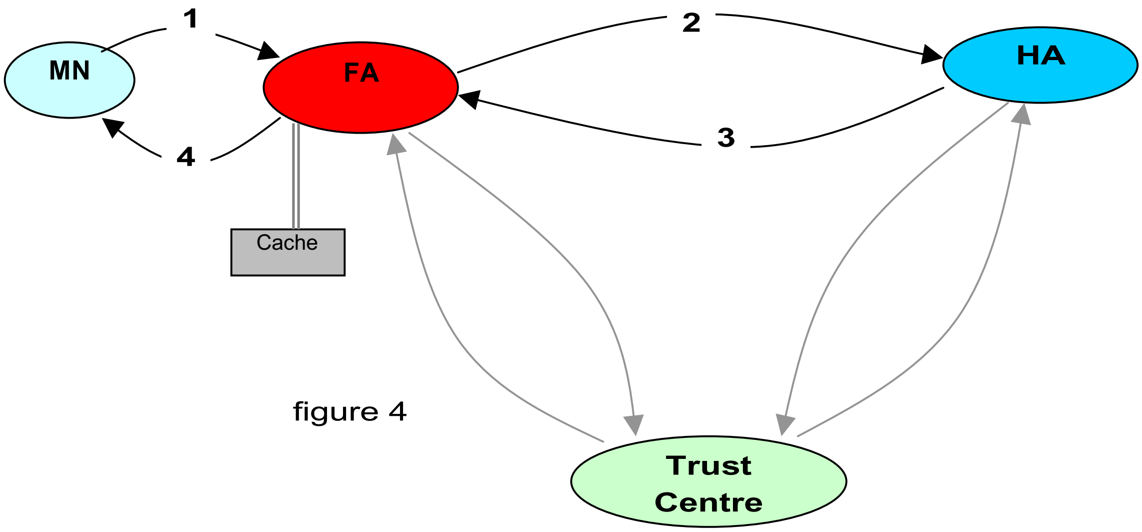


figure 4

The first message is registration request to the Foreign Agent , then as described in MIP the Registration Request is forwarded to Home Agent (HA) this is message 2. After checking the request for validity the HA replies with the message 3 'registration reply', which is checked and at the end the message 4 is transmitted to MN and MN is attached to the Network.

During the registration procedure all nodes are authenticated and two secret keys are shared between FA,HA,MN. These keys are used for the service messages between FA,MN,HA for example re-registration after the Life Time period expires. The first key is for Authentication of the data and the second is used for encryption of the data. They are called authentication and encryption keys. The description of their exact use is to be read in 3.2.2.

The phase of authentication is called "first contact". When the MN is staying in the FA network and there is a need only of re-registration after the Live Time period expired comes the phase "re-registration". When a Corresponding Node (CN) required connection to MN, then first session keys are generated in the phase "session key generation" The session key generation and CN authentication needs exchanging of 5 messages. After CN and MN can communicate using data flow encryption and protection.

It is important to notice that for the data flow encryption and protection are used one encryption key shared between MN and CN and one authentication key shared between FA,HA,MN,CN. In this way FA, MN, HA, CA can authenticate the data, but only MN and CN can decrypt/encrypt it, because the encryption key is shared only between MN and CN.

After changing the FA the MN doesn't need to changes the keys . MN should only inform the new FA, after the 'first contact' phase, with the CN`s IP and authentication key for this session. The same for all session which are available. There is no need of changing the keys. This is also an advantage of MVPN. This procedure is also described in the following chapters.

3.4 Mobile Virtual Private Network

3.4.1. First Contact

In this phase the Mobile Node is visiting the network managed by a Foreign Agent for the first time. None of them knows the Public Key of the other party and they do not share any information that might be used to attach them. The Mobile Node is only aware of the Public Key of its Home Agent.(see figure 4)

The First Contact is performed by the distribution of four extensions that are added to the Registration messages already needed to be exchanged between the mobile entities.

The first message received by the Mobile Node is the ICMP Agent Advertisement as described in [1]. This message MUST not be protected since it represents the first encounter the Foreign Agent has with any Mobile Node and it is certain that a Mobile Node does not have any association with the Foreign Agent, i.e. does not know the Public Key of the Foreign Agent.

The Foreign Agent MUST set the sequence number field of the ICMP Agent Advertisement message. This 16-bit value MUST be used by the Mobile Node as a nonce, this value MUST be returned to the Foreign Agent to help it keep track(also statistic) of the advertisements answered and to insure that the messages are not been replayed.

3.4.1.1 Registration Request

After receiving the ICMP advertisement, the Mobile Node answers with a Registration Request, which must include an authentication extension and the 'mobile information' extension defined by the MVPN protocol.(see figure 5)

Since the Mobile Node is using a **SMART CARD** it is possible for the home and foreign domains to keep a record and inventory of all the Mobile Nodes using their services. That is why the extensions defined by the MVPN include relevant information about the identity of the Mobile Node, consisting of specific information of the SMART CARD.

If the Mobile Node is not using a SMART CARD then the information that corresponds to it can be omitted or the fields can be used to provide other relevant information, this information can also be discarded to reduce the size of the packet.

Other information included in the **'mobile information'** extension will serve the Home Agent to perform the appropriate authentication and the creation and distribution of the session keys.

The 'mobile information' extension has the following logical structure:

1 **M=[MN Nonce || DVC MAC ID || USER ID || MN Auth key]**
Mobile Inf Ext. = $\mathcal{E}_{Kr\ MN}$ [$\mathcal{E}_{Ku\ HA}$ [M]] || Other Info || HA ID || ICMP Nonce

MN Nonce 32-bit number generated by the Mobile Node as nonce

ICMP None the number of the ICMP Agent Advertisement the mobile node is responding.

DVC MAC ID 64-bit identification of the computer used as Mobile Node

USER ID 48-bit identification of the Mobile Node

SC ID 64-bit identification of the SMART CARD the Mobile Node is using.This information is optional.

OTHER INFO 64-bit information for the Foreign Agent. It is used in case the Foreign Agent cannot contact the Home Agent and the Foreign Agent has the Mobile Node's Public Key in its cache memory or obtained it through a Trust Centre (CA). This information is optional.

MN Auth key key used in the calculation of the authentication extension.

The complete structure of the 'mobile information' extension can be found in section 6.2.1.

If the Home Agent is not available it is possible for the Mobile Node to provide its Public Key to the Foreign Agent, this is not recommendable during the 'first contact' since it could be possible for any attacker to intercept the key and use it. The retrieval of the Public Key with the Trust Centre (CA) enables the Foreign Agent to verify its authenticity. This will allow the Foreign Agent the provision of certain services while trying to establish contact with the Home Agent.

The authentication extension will be constructed as defined in [1] and is used to protect the integrity of the packets. The authentication key applied for the generation of the authentication extension will be added to the 'mobile information' extension to enable the Home Agent to perform the appropriate verification of the message and authenticity of the node.

There is the possibility of using an optional extension 'Home Agent addresses', which contains the addresses of other Home Agents trusted by the Mobile Node's Home Agent that can perform the authentication of the Mobile Node, they share the private key of the Home Agent and all the Public Keys belonging to the Mobile Nodes. They cannot provide typical Home Agent services such as binding messages, tunnelling and packet forwarding, but they will enable the Mobile Node to have access to Internet services.

The use of Home Agent negotiators provides the capability of allow a valid user to communicate, more information about this extension will be provided in a future draft. Since this is the first contact the Mobile Node has with the Foreign Agent, the latter will not be able to perform the integrity check and authentication of the packet. The Foreign Agent MUST save the information contained in the packet, create the 'foreign information' extension, add it to the Registration Request message and send it to the Home Agent. The integrity and authentication of the MN Registration Request message will be performed until the Home Agent replies to the Foreign Agent about that request, i.e. when it receives the key used by the Mobile Node for the creation of the authentication extension.

The purpose of the 'foreign information' extension is to provide information to the Home Agent so it can perform the authentication of the Foreign Agent. The Foreign Agent MUST request the public key of the Home Agent to the Trust Centre (CA) by using the security association existing between them, since the Home Agent belongs to a different domain the Trust Centre (CA) will need to contact another Trust Centre (CA) in order to retrieve the appropriate key.

The 'foreign information' extension includes the Foreign Agent IP address, a nonce and the identification of the Foreign Agent. The complete structure of the extension is shown in section 6.2.2.. The logical structure is shown

2 **F=$\mathcal{E}_{Ku\ HA}$[FA ADR || FA NONCE || Supp Alg]**
Foreign Inf Ext. = $\mathcal{E}_{Kr\ FA}$[F] || FA ID

down:

FA Nonce	32-bit nonce number generated by the Foreign Agent
FA ADDR	32-bit IP address of the Foreign Agent
FA ID	64-bit Identification of the Foreign Agent
Supp Alg	Supported encryption and authentication algorithms

The authentication extension provided by the Mobile Node is no longer needed, the Foreign Agent will calculate a new one that includes a **Digital Signature** for integrity protection. More protection will be provided in this message since it will travel outside the foreign domain, i.e. public network. This escalation of security improves the optimisation of the authentication process. If the FA and Ha consider that DSS is not needed because of the computational resources, it's sure possible to be used HMACs or keyed H-functions, as described for the previous messages..

The Home Agent will receive the Registration Request and perform the authentication of the Mobile Node. Also it will verify the integrity of the message by using the digital signature included in the packet, the Public

Key of the Foreign Agent and the group key, generated by the Trust Centre (CA).

3.4.1.2. Registration Reply

After verifying the integrity and authentication of the packet the Home Agent will reply to the Foreign Agent with a Registration Reply message as defined in [1].

The Registration Reply MUST include the 'home information' extension, which serves the same purpose as the other extensions previously described.

It will provide information about the Home Agent as well as the session keys, encryption and authentication, that the mobile entities will use in future communications.

Some of the information will be available only for the Mobile Node, which means that it will be protected with the Mobile Node's Public Key, it is important to maintain confidentiality between the Mobile Node. In this way Foreign Agent is authenticated to MN using the trust in Home Agent. The Home Agent works as a CA between FA and MN. In the extension reply is provided the Foreign Agent information relevant to the services needed to be supplied to the Mobile Node.

The 'home information' extension contains the nonce provided by the Foreign Agent, result of the Registration Request, Home Agent's identification, Public Key of the Foreign Agent (provisioned by the Trusted Centre), encryption and authentication keys (for future communications), authentication key used by the Mobile Node for the creation of the authentication extension sent by the Mobile Node to the Foreign Agent, and the Public Key of the Mobile Node.

3 **Q=FA NONCE || Result INF || KuMN ||**
|| Mn Nonce || Ses Enc Key || Ses Auth Key || MN Auth Key

W= $\mathcal{E}_{\text{Ku MN}}$ **[KuFA || MN Nonce]**

Home Inform. Extension = $\mathcal{E}_{\text{ku FA}}$ **[** $\mathcal{E}_{\text{Kr HA}}$ **[Q || W]]**

MN Nonce	32-bit nonce generated by the Mobile Node
KuFA	n-bit Foreign Agent's public key
Ses Auth Key	n-bit session key used by the mobile entities for dataflow authentication
MN Auth Key	n-bit authentication used by the mobile Node for Registration Packet authentication
Ses Enc Key	n-bit session key used by the mobile entities for dataflow encryption
KuMN	n-bit Mobile Node's public key

Result Inf Information form HA to FA
Sec Alg Which security algorithms are to be used for authentication and encryption, corresponding to Ses Enc Key and Ses Auth Key

The complete structure of the 'home information' extension is shown in section 6.2.3.

The extension MUST be added to the Registration Reply and sent to the Foreign Agent as a 'home information' extension. The Foreign Agent MUST verify the integrity of the message and the authentication provided by the Home Agent, including the verification of the nonce.

With the information provided by the Home Agent, the Foreign Agent is now able to authenticate the Registration Request message provided by the Mobile Node, after that the message can be discarded.

The Ses Auth Key and Ses Enc Key are **shared key between FA, HA, MN**. The keys are used for service messages between the parites. They are used in the re-registration procedure, when the MN is still in FA network but the Life Time of the registration is at the end and MN should send new registration request. In this case the request is sent as usual, but the hole data flow is protected as described in 3.5.1

The Foreign Agent will extract the information relevant to the Mobile Node and create the 'foreign reply' extension that contains an acknowledge value that indicates whether the authentication process was successful or not, the nonce value provided in the registration request by the Mobile Node, registration and authentication keys, and Foreign Agent's public key.

4 **T=** $\mathcal{E}_{\mathbf{Ku\ MN}}$ **[**$\mathcal{E}_{\mathbf{Kr\ FA}}$ **[ACK || MN Nonce || Ses Enc Key || Ses Auth Key]]**
Foreign Reply Ext. = T || W

Ack Connection acknowledgement

The complete structure of the 'foreign reply' extension is shown in section 6.2.4.

The values ICMP nonce and MN nonce help the Foreign Agent to identify the Mobile Node it needs to authenticate and to prevent any **replay attack**. After performing the authentication, the Registration Reply MUST be encrypted with the public key of the Mobile Node and its integrity MUST be protected with an authentication extension created with the application of a **keyed hash function** or **HMAC function** and using the authentication key provided by the Home Agent in the 'home information' extension and generated from MN.

The Registration Reply (message 4) is sent to the Mobile Node, with the 'foreign reply' added. The Mobile Node MUST verify the authenticity of the packet and record the encryption and authentication keys provided by the Home Agent. At the end the Mobile Node is capable to communicate with any correspondent node, providing that the Registration process was successful, and the mobile entities share an encryption and an authentication key, they also know the public keys of the other parties.

The hole procedure is shown graphically once more on the figure 6

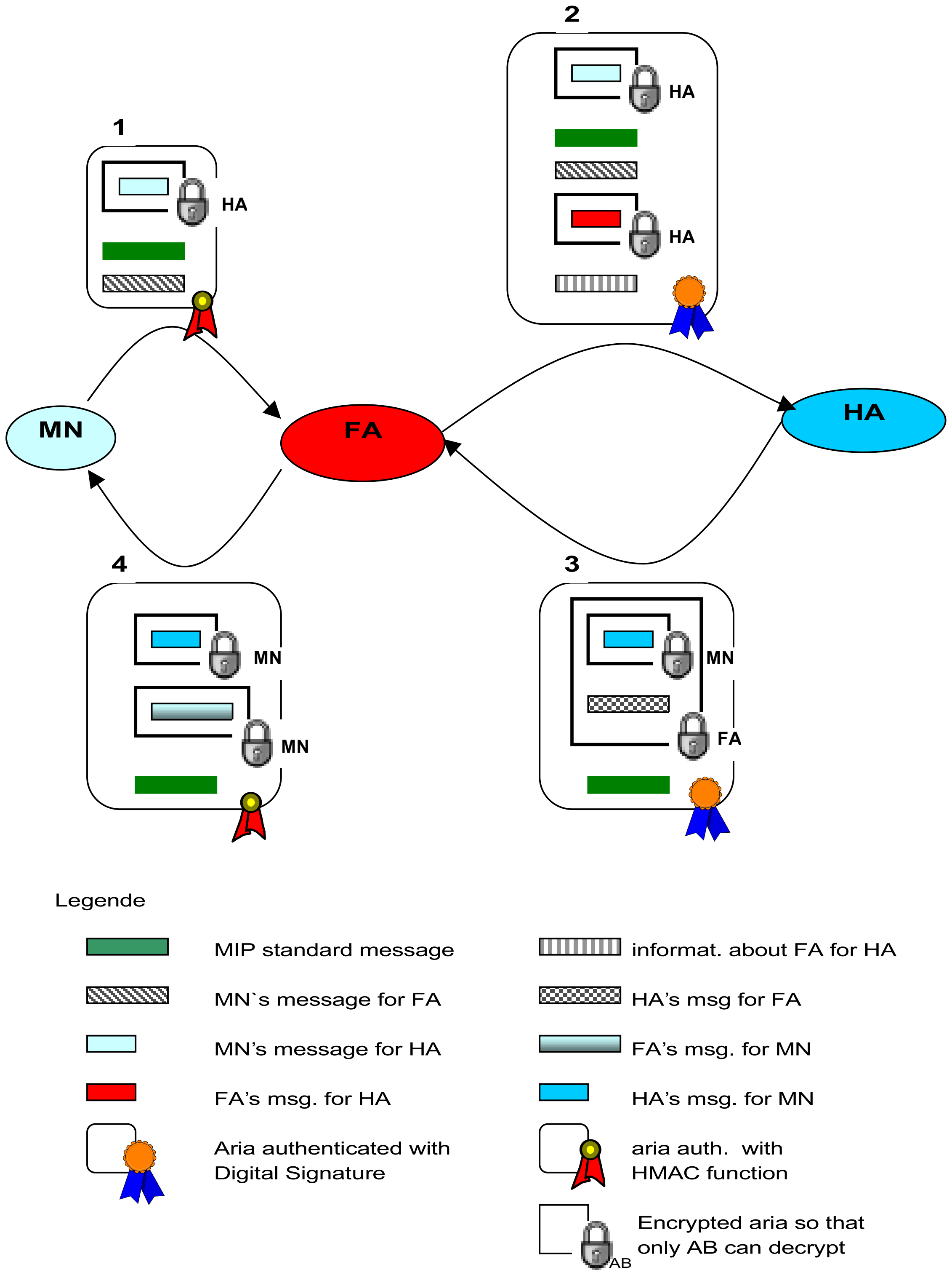

2
HA
HA
1
HA
MN
FA
HA
4
MN
MN
3
MN
FA
Legende
MIP standard message
MN`s message for FA
MN's message for HA
FA's msg. for HA
Aria authenticated with Digital Signature
informat. about FA for HA
HA's msg for FA
FA's msg. for MN
HA's msg. for MN
aria auth. with HMAC function
Encrypted aria so that only AB can decrypt
AB

figure 6

3.4.2. Re-registration of Mobile Node

3.4.2.1. Re-visiting a Foreign Agent

The re-registration phase refers to the registration of a Mobile Node to its current point of attachment, in the case that the lifetime of the previous Registration Request is about to expire and a new registration is needed and the Mobile Node is staying in the current network.

During the previous registration all the necessary information was distributed to the mobile entities, which means that **less calculations** will be required and more time will be saved. The purpose of MVPN is to optimise the authentication process for security and performance. The re-registration specified in this document will be applicable only if the lifetime of the encryption and authentication keys shared between the mobile entities are still valid, if not the Mobile Node will need to complete the 'first contact' phase for the registration. Before the lifetime of the previous registration expires, the Mobile Node MUST send a normal Registration Request message without including any extension, but protecting the data flow by using the encryption and authentication keys that the mobile entities already share. In this way the procedure of re-registration is **quicker and more secure**. The Data flow is protected as described in 3.6.

The Registration Request received by the Foreign Agent will be encrypted and authenticated, since the Foreign Agent has a copy of the keys it MAY decrypt and verify the integrity of the packet.

The Foreign Agent MAY decide to delegate the message to the Home Agent by only changing the IP header of the packet, or it can perform the validation of the Registration Request.

3.5. Changing the Foreign Agent

When the MN changes the Network to a new FA the **first step** is the authentication which is made as described in ‘first contact’ phase. **Second step** : the Mobile Node MUST provide the session authentication key of his current connection with different CN to the new Foreign Agent. This will allow the Mobile Node **to keep communicating without interruptions** and the new FA can start authenticating the packets.

The message MUST be sent as an UDP message and MUST contain the ID of the session held between the Mobile Node and the correspondent node, the correspondent nodes' IP address, the length of the authentication key and the authentication key. The UDP message MUST be protected with data flow protection with the Authentication key and Encryption Key shared between FA, MN, HA .(see 3.6)

Every time a Mobile Node changes Foreign Agents or the lifetime of the previous Registration Request ends, the keys will be considered expired and the Mobile Node will need to register following the procedure indicated in the 'First Contact' phase.

The UDP message on Port 505 structure is shown in figure 7.

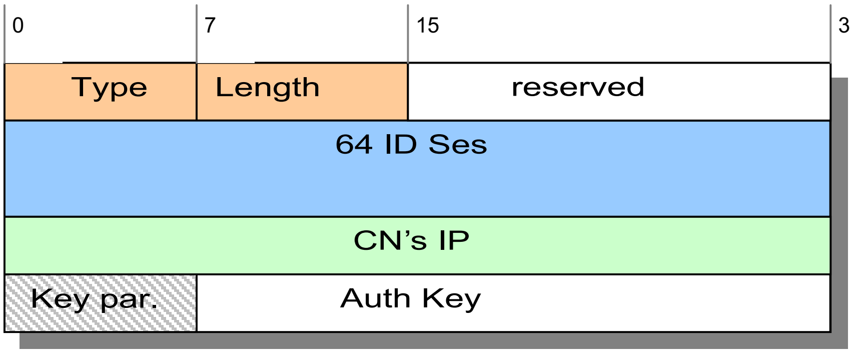


figure 7

Type	5
Length	includes the length of this datagramm in octets
64 Id Ses	the session ID with the CN
CN's IP	includes the CN IP address
Key par	the Parameters of the Sec Key :the length of the following Key in octets , algorithm, mode of operation. See 6.2.1
Auth Key	the Authentication Key

If there are many current connection the MN may sand many datagramms (fig. 7) for all connections in one UDP packet, chaining them one after onather.

3.6. Session key generation

Once the authentication and the public key exchanged have been performed the Mobile Node is able to communicate with a correspondent node. In the model presented here we specified the requirement of authenticating the correspondent node before establishing any communication with the Mobile Node.

The Home Agent MUST perform the initial authentication of the correspondent node and MUST create and distribute an authentication key that will be shared between the mobile entities, including the correspondent node.

The session key can have a variable extension to allow the use of any encryption algorithm selected by the communicating parties. *The basic principle is :*

- *For authentication of the packets exchanged between the nodes is used one shared authentication key between FA, HA, MN, CN. In this way the data flow can be authenticated form all nodes.*
- *For decryption/encryption of the packets is used one encryption key shared only between MN,CN.*

The FA,HA will not be able to decrypt the packets, only the correspondent node and Mobile Node are capable of doing that. In that way confidentiality between the mobile and correspondent nodes is still guaranteed.

The correspondent and Mobile Nodes MUST generate a session key, using the Diffie-Hellman algorithm, to perform data flow encryption. This key will be shared only between them. We recommend the use of **Elliptic Curve Diffie-Hellman** key exchange, because it's cryptographicaly more powerful then the standard RSA based version.

Once again : There will be one data flow authentication key between the Home Agent, Foreign Agent, correspondent node and Mobile Node, and one data encryption key shared between the correspondent node and Mobile Node.

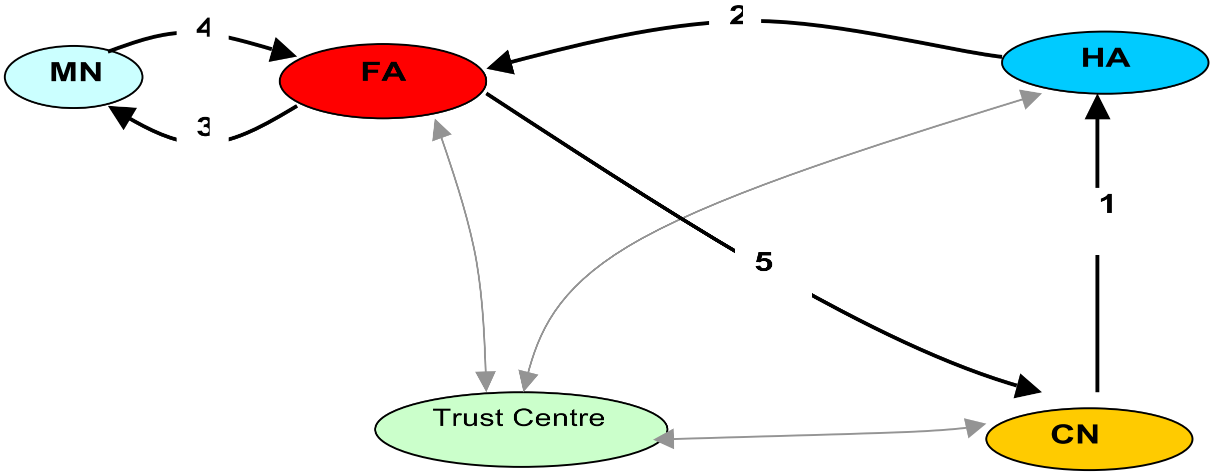


figure 8

The correspondent node - Mobile Node session key is not modified when the Mobile Node changes Foreign Agents, the new Foreign Agent MUST receive the authentication key from the Mobile Node so that the data flow continues as before.

Before communicating with the Mobile Node, the correspondent node needs to authenticate itself to the Home Agent by using its public key registered with the Trusted Centre, and receive the authentication key distributed by the Home Agent. All messages, including the session key exchange messages, are sent as **UDP on port 505**

The structure of the all UDP messages is simular to the structure of the MIP extention and is presented on figure 9.

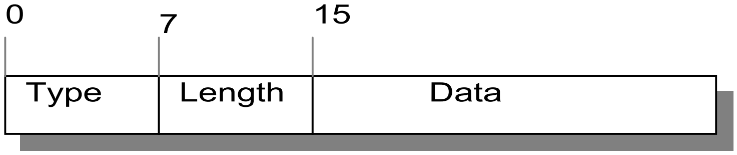


figure 9

The type field set to the number of the message from 1 to 5

There is no difference where the communication challenge (Who starts the communication first) comes from , the procedure is similar. We describe in details, when it comes form CN. Briefly is described if it comes MN.

The correspondent node sends a 'Request Communication' message to the Home Agent, which includes a random generated number representing the Y value of the Diffie-Hellman algorithm, the identification of the correspondent node, the IP address of the Mobile Node, a nonce, and the security algorithms supported by the correspondent node.

V = $\mathcal{E}_{Kr\ CN}$ [Ycn || CN ID || CN IP || q Value || α Value || Nonce CN]

1 **Request Comm. Msg. = $\mathcal{E}_{Ku\ HA}$ [V || Supp Alg || MN Addr]**

Ycn	first value in Diffie-Hellman exchange algorithm
CN Id	Id Number of the CN ,provided form the trust Centre (CA)
Supp Alg	Flag showing which security algorithms are supported by CN
MN Addr	IP Address of the MN
CN IP	IP address of the CN
q Value	q Value in DH algorithm the modulator
α Value	the power value in DH algorithm

Nonce CN Nonce value calculated form CN

The complete structure of the 'Request Communication' message is shown in section 6.3.1.

The security algorithms are represented as flags, if the flag has a zero value it means that the correspondent node does not support a specific security algorithm (SHA-1, MD5). This will allow the Mobile Node and Foreign Agent to select an algorithm that they implement, for detail see 5.1

The information will be encrypted with the Public Key of the Home Agent and the private key of the correspondent node, depending on the type of information to be sent. The authentication of the message will be performed by the Home Agent using the Public Key of the correspondent node.

The Home Agent MUST authenticate the message and check its integrity, if everything is correct it MUST generate the session key for authentication Notice that there are not extra authentication extension , the authentication procedure is made with using the encryption with public key/private key.

Then consequently comes the question for protection against '**cut and paste**' attacks. The protection is realised with putting the CN ID and CN IP in the protected part of the message. In this way there is no possibility of cut and paste attacks.

HA sends a 'Correspondent Information' message to the Mobile Node that includes the session key, the Y value, correspondent nodes' identification, public key, and IP address, a 'Hello value' and the security algorithms supported by the correspondent node.

2 **$U=\mathcal{E}_{Ku\ MN}$ [Ses Auth Key || V || Ku CN]**
Corr. Inform. Msg = Hello Val || Supp Alg || FA IP || U || Session ID

Hallo Val Control String
Session ID Sequential number, specifying the Current CN-MN session
Fa IP The IP of FA
see V message

The complete structure of the 'Correspondent Information' message is shown in section 6.3.2.

! Notice that MN,CN,FA communicate using **data flow protection**., that is the reason why we don't need to protect the whole sent information. *Only in the phase 'first contact' the parities communicate without using data flow protection technique*. After the 'first contact' the parities share a secret keys and all communications are protected with data flow protection. That is why only the first message CN to Ha must be totally protected

Since the packets are received by the Foreign Agent this needs to be involved in the process of establishing the communication parameters, it will verify the 'Hello value', extract the authentication key, set the security algorithm that it supports and send this information along with the one received from the Home Agent to the Mobile Node. To protect the message

3 **FA Retriving. Msg = U || Sec Par || Mn IP || Session ID**

is used as mention before the data flow protection and encryption with the keys arranged in the ‘first contact’ phase.

The Mobile Node MUST reply to the Foreign Agent with a 'Mobile Response' message by sending a random number that represents the Y value of the Diffie-Hellman algorithm, the nonce generated by the correspondent node and indicate the security algorithms it supports. This information will be protected with data flow protection.

4 **L =$\mathcal{E}_{K\ MN}$ [$\mathcal{E}_{Ku\ CN}$ [Ymn || Nonce CN]]**
Mobile Respond = L || Supp Alg || Ses Auth Key || Session ID || FA IP || Nonce CN

DH_{mn} Flag DH Flag of message 1,which describes which DH version is used.

If *DH_{mn} Flag is differefrom to the DH Flag* values send from the CN in the first message. That means, that the MN does not support the DH algorithm suggested by CN and MN suggests DH_{mn} Flag algorithm. In this case the CN should challenge MN for a new Connection request and with DH method suggested from MN. And the current procedure is cancelled.

If *DH_{mn} Flag is equal to the DH Flag* send in Message 1 that means that MN supports the algorithm and accept it and sends his DH value.

The complete structure of the 'Correspondent Response' message is shown in section 6.3.4.

Finally, the Foreign Agent MUST send the 'Communication Reply' message to the correspondent node. It will basically contain the same information as the one contained in the 'Mobile Response' message.

5 **Comm.Reply Msg = $\mathcal{E}_{Ku\ CN}$ [Ses Key || Ses Par || Nonce CN ||Session ID]|| L**

The complete structure of the 'Communication Reply' message is shown in section 6.3.5

The mobile entities, including the correspondent node, share a session key for authentication, and the Mobile Node and correspondent node have the material to calculate the encryption key by using the Diffie-Hellman algorithm. They will use as encryption algorithm from the specified in the security parameters field (supported algorithms – "Supp Alg") of the messages.

Now the nodes are able to communicate with the added value of authentication and encryption of the data flow. It is important to notice that the establishment of security associations between the correspondent nodes with the mobile entities is optional, but it is significant to consider the authentication and protection of information generated by a correspondent node since it will also have access in an indirect way to the protected networks.

If MN starts the communication the message exchange work absolutely in the same way. The MN sends to FA the first message and FA forwards the Message to HA. The MN generates the MN Nonce (corresponding to Nonce CN in first case) The FA generates the session authentication key and authenticates the CN. With the use of Trust centre (CA) the FA can find the CN Public Key and all needed parameter.

In this way the **calculation load is shared** between FA, HA. The Scheme is shown in figure 10.

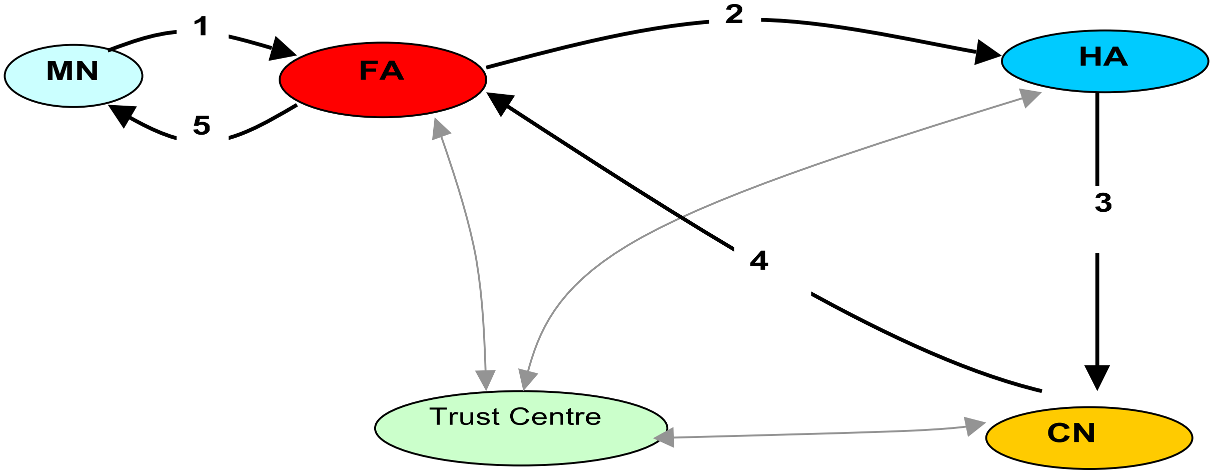


figure 10

3.7 Mobile data flow

The data field of the packet that flows from the Mobile Node to the correspondent node, or vice versa, contains information about the security association built between the Mobile Node and the correspondent node during the process described in section 3.5. This security association makes reference to information for encryption and authentication of the packets exchanged between them.

The structure of the packet, from the Mobile Node's point of view, will be as follows:

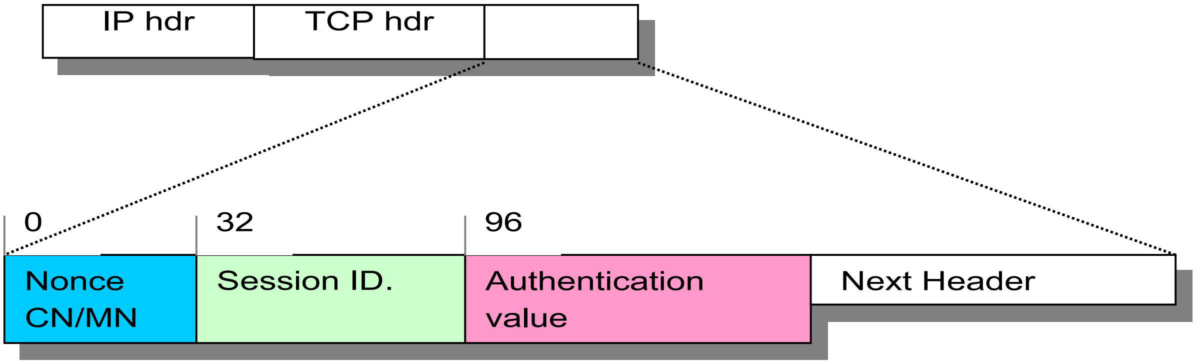


figure 11

The authentication value is added to verify the integrity of the packet, and is calculated using a keyed MD5 or HMAC of the *entire IP packet*. The key is the one shared between the mobile entities and the correspondent node, and it is distributed as described in section 3.5.

The Nonce MN/CN will be used as a first step for authentication, if the value is not correct then the packet is considered invalid and therefore it is discarded. It is used to protect against replay attacks and to serve against minor **Denial of Service** attacks.

The **Session ID** is generated form Ha (or FA) in 3.5 and is used to indentify the session and the *corresponding secure parameters*. The **Secure parameters** are showing the encryption and authentication function and mode of operation. If Session ID is set to 0 that means, that this is a service message between FA,HA or CN and the secure parameters are the parameters exchanged in 'fist contact' phase. The secret key are the keys established in the 'first contact'

We believe, that the **Elliptic Curve** *will play a very significant role in the next generation secure mechanisms* . We recommend to use algorithms based on elliptic curves, because of the high security level of encryption and the key size.

The message received by the Home Agent from the correspondent node will be verified by using first the CN nonce value and then the authentication value. The Home Agent will **tunnelled** the packet using **IP in IP** or **TCP in TCP** (see section 4 for more details) to the Foreign Agent modifying the authentication value to protect the new TCP header.

The Foreign Agent will authenticate the packet, **de-tunnelled** it and delivered it to the Mobile Node, building a new authentication value. The Mobile Node can prove the authentication and can decrypt the message by using the shared key with the correspondent node. Notice that the encryption key is different from the authentication key.

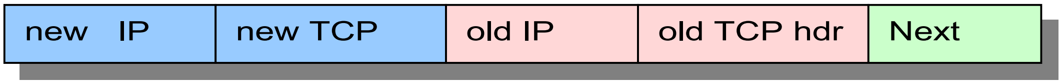
4 TCP in TCP

When there is a need of tunnelling ‘Ip in Ip’ , for example between HA and FA we recommend the use of TCP in TCP as mentioned earlier, the IP in IP tunnelling used in mechanisms such as IPsec is not optimal because it interferes with the use of:

- Optimisation for GEO satellite connections. Techniques such as 'TCP connection splitting'. GEO's are a very affordable technology that will probably be part of the future.
- Firewall filtering.
- Network monitoring.
- Enchanted proxies for better performance of TCP connections

If tunnelling is still needed a better option will be TCP in TCP, which will solve the problems IP in IP presents.

It has a simple mechanism to construct it, new TCP and IP headers are added in front of the original packet as shown in the following diagram, and the original packet can be encrypted and authenticated.



As disadvantages, the packet will contain redundant information (extra TCP header) and the traffic will be incremented. But it will work as a tunnelling friendly to all earlier mentioned techniques. There is no increasing of the packets number which is on our opinion also very important for wireless connections and keeps the packets number minimal.

The TCP header might contain information that an attacker can use, but other techniques such as port masking can be used. Our recommendation is to work with TCP in TCP tunnelling since MVPNs do not require the TCP in TCP tunnelling that IPsec or other mechanisms provides.

5. Error messages

In case of encountering an error during the communication of nodes MVPN specifies the use of error messages. Upon the arrival of the error message, the receiver will not need to perform any specific mechanism, these messages have informational purposes only.

The errors occurred during the 'first contact' phase will be sent as UDP packets at port 434 and if there are errors during any phase after ‘first contact’ errors as UDP on port 505. If data flow protection is possible to be used then it MUST be used. If there is no valid secret key between the nodes than, the error message is send not encrypted.

Following is the structure of the error message:

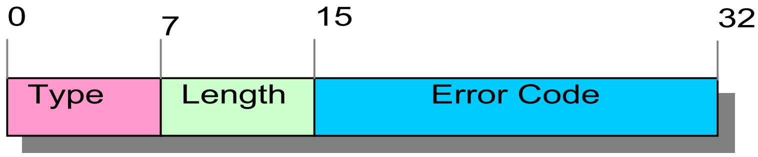


figure 13

Error code

01	Authentication key not valid
02	Session key not valid
03	No support offered for MVPN
04	Public key not valid
05	Home Agent not found
06	Trusted Centre not found
06	Poor structure of the message
07	other

The mobile entities can also use the error messages specified by the MIP.

6. Messages structure

6.1 C notation

For the specification of the message are used the standard **C notation.** It is supposed that the notation is common to the reader. Here are remembered briefly once again the main data and structure types.

6.1.1 Vectors

A vector (single dimensioned array) is a stream of homogeneous data elements. The size of the vector may be specified at documentation time or left unspecified until runtime. In either case the length declares the number of bytes, not the number of elements, in the vector.

The syntax for specifying a new type *T'* that is a fixed length vector of type *T* is :

```
T T'[n];
```

Here *T'* occupies *n* bytes in the data stream, where *n* is a multiple of the size of *T*. The length of the vector is not included in the encoded stream.

6.1.2 Numbers

The basic numeric data type is an unsigned byte (`uint8`). All larger numeric data types are formed from fixed length series of bytes.

```
uint8 uint16[2];
uint8 uint24[3];
uint8 uint32[4];
uint8 uint64[8];
```

6.1.3 Enumerateds

An additional sparse data type is available called `enum`. A field of type `enum` can only assume the values declared in the definition. Each definition is a different type. Only enumerateds of the same type may be assigned or compared. Every element of an enumerated must be assigned a value, as demonstrated in the following example. Since the elements of the enumerated are not *ordered*, they can be assigned any unique value, in any order.

```
enum { e_1(v_1), e_2 (v_1), ... , e_n (v_N), [(n)] } T_e;
```

Enumerateds occupy as much space in the byte stream as would its maximal defined ordinal value. The following definition would cause one byte to be used to carry fields of type `Color`.

```
enum { red(3), blue(5), white(7) } Color;
```

6.1.4 Constructed types

Structure types may be constructed from primitive types for convenience. Each specification declares a new, unique type. The syntax for definition is much like that of C.

```
struct {
    $T_1$ $f_1$;
    $T_2$ $f_2$;
    ...
    $T_n$  $f_n$;
} [T];
```

The fields within a structure may be qualified using the type's name using a syntax much like that available for enumerateds. For example, *$T.f_2$* refers to the second field of the previous declaration. Structure definitions may be embedded.

6.1.5 Variants

Defined structures may have *variants* based on some knowledge that is available within the environment. The selector must be an enumerated type that defines the possible variants the structure defines. There must be a case arm for every element of the enumeration declared in the select. The body of the variant structure may be given a label for reference. The mechanism by which the variant is selected at runtime is not prescribed by the presentation language.

```
struct {
    $T_1$ $f_1$;
    $T_2$ $f_2$;
      ....
    $T_n$ $f_n$;
    select (E) {
      case $e_1$: $T_{e1}$;
      case $e_2$: $T_{e2}$;
        ....
      case $e_n$: $T_{en}$;
    } [$f_v$];
} [$T_v$];
```

6.2 Messages in the ‘first contact’ phase

6.2.1 Registration request form MN. Extension Structure

We are describing only the **extension payload structure**. The rest part of the packet is s in MIP[1] described.

The variable ‘MValues’ is encrypted with `Public Key of HA (Ku HA) and the private key of MN (Kr MN).

```
struct {
        uint8           Auth_Alg; /* the value is taken from in 7.3 table for
                                      authentication flag. */

        select (Auth_Alg)  {

          case          HMAC_SHA1,HMAC_MD4      :       uint8  Key[16];
          case else                             :       uint8  Key[7];
        };
}  Authentication_Key;

struct {

        uint32                  Nonce_MN;
        uint64                  DVC_MAC_ID;
        unit64                  USER_ID;
        Authentication_Key      Mn_Auth_key;

} PayloadM

struct {
        uint8           Length;  /*  the value is set to the length of MValues in octets */
        PayloadM        MValues; /* the value is encrypted with KuHA and KrMN*/
} M;

struct {

        M               EncryptedMValues;
        uint64          Other_Info;
        uint64          HA_ID;
        uint16          ICMP_Nonce;

} Mobile_Information_Extension;
```

6.2.2 Foreign Agent Information Extension.

The Values of the Flag values are set recording the defined values for Supported encryp-,

-tion algorithm and authentication algorithm in **Table 7.3**

```
struct {

        uint8           Supp_Auth_Alg;
        uint8           Supp_Encr_Alg;
} Supp_Alg;

struct {

        uint32FA_ADR;
        uint32FA_Nonce;
        Supp_Alg    ValueSuppAlg;

} PayloadF;

struc {

        uint8           Length;  /*  the value is set to the length of FValues in octets */
        PayLoad         FValue; /* the value is encrypted with KuHA and KrFA*/

} F;

struc {

        F               EncryptedFValues ;
        Uint64          FA_ID

} Foreign_Information_Extension;
```

6.2.2 Home Agent Information Extension

The values of PK_Key_Typ are described in ***7.1.1***

```
struct {

        uint8           PK_Key_Typ;
        select (PK_Key_Typ)  {

                RSA_xxx, ECC_xxx :   {

                        uint8  Key_Length;  /*The value include the lenght  of
```

```
                            the fowolling  key */

            opaque      Key[Key_Lenght]
        };

        RAS_512     :   uint8  Key[64];
        RAS_768     :   uint8  Key[96];
        ECC_113     :   uint8  Key[15];
        ECC_153     :   uint8  Key[21];
    };
}  Public_Key;


struct {
    uint8       Encr_Alg; /* the value is taken from in 7.3 table for
                    encryption flag. */

    select (Encr_Alg)  {



     case       RC5_CBC,DES_CBC     :   uint8  Key[7];
     case       DES_CBC_40          :   uint8  Key[5];
     case       3DES_CBC            :   uint8  Key[21];
     case else                      :   uint8  Key[8];

    };
}  Encryption_Key;




struct {

    Public_Key      KuFa;
    Uint32          MN_Nonce;

} PayLoad_W;




struct {
    uint8       Length;  /*  the value is set to the length of MValues in octets */
    Payload_W WValues; /* the value is encrypted with KuMN */
} W;

struct { /* The hole structure is encrypted with KuMN */
```

```
        uint32                  FA_Nonce;
        uint32                  ResultInfo;
        Public_Key                      KuMN;
        uint32                  MN_Nonce;
        Encryption_Key                  Session_Encr_Key;
        Authentication_key              Session_Auth_Key
        Authentication_Key              MN_Auth_Key;
        W                               WValue;
} Home_Information_Extension;
```

6.2.3 Foreign Reply Extension

```
struct { /* The hole structure is encrypted with KuMN and KrFA */

        uint32                          Akn;
        uint32                  MN_Nonce;
        Encryption_Key                  Session_Encr_Key;
        Authentication_key              Session_Auth_Key;

} T;


struc {

        T               TValue;
        W               WValue;

} Foreign_Reply_Extension;
```

6.3 Session key generation

6.3.1 Request Communication Message

```
struct {
        uint8           DH_Alg; /* the value is taken from in 7.4 table for
                                   Diffie-Hellman  key exchange algorithm. */

        select (DH_Alg)  {

          case          DH_512              : {
                                                uint8  Key[64];
                                                uint8 q_value [64];
```

```
                                        uint8 q_value [64];
                                        }
     case        DH_768             :{
                                        uint8  Key[96];
                                        uint8 q_value [96];
                                        uint8 q_value [96];
                                    }

     case        ECDH_113           :{
                                    uint8  Key[15];
                                     /* for ECC are used the recommended
                                        in 7.2  public values   */
                                    }
     case        ECDH_131           :{
                                    uint8  Key[17];
                                    /* for ECC are used the recommended
                                    in 7.2  public values    */
                                    }
    };
}  DH_Exchange_Value;

struc {

    DH_Exchange_Value    Ycn;
    uint64               CN_ID;
    uint32               CN_IP;
    uint32               Nonce_CN;
} PayLoad_V;

struct {
    uint8        Length;  /*  the value is set to the length of VValues in octets */
    PayloadV     VValues; /* the value is encrypted with KrCN */
} V;

struc {
    uint16Supp_Alg ;
    uint32MN_IP
    V            Vvalue;
} Request_Comm_Message;
```

6.3.2 Correspondent Node Information Message

```
struct {
        Authentication_Key        Ses_Auth_Key;
        V                         Vvalue;
        Public_Key                KuCN

} PayLoad_U;

struct {
        uint8           Length;  /*  the value is set to the length of UValues in octets */
        Payload_U  UValues; /* the value is encrypted with KuMN */
} U;

struct {

        uint64Hallo_value;
        uint16Supp_Alg;
        uint32FA_IP
        U               Uvalue;
        Uint64          Session_ID;

} Corresponden_Node_Information_Extension;
```

6.3.3 FA Retriving Message

```
struct {

        U               Uvalue;
        Uint16          Supp_Alg;
        Uint32          MN_IP;
        Uint64          Session_ID;

} FA_Retriving_Message;
```

6.3.4 Mobile Respond

```
struct {

        DH_Exchange_Value     Y_MN;
        uint32                     Nonce_CN;
} PayLoad_L;

struct {
        uint8           Length;  /*  the value is set to the length of LValues in octets */
        Payload_L   LValues; /* the value is encrypted with KrMN and KuCN*/
} L;

struct {
        L                              Lvalue;
        uint32                 Supp_Alg
        Authentication_Key       Sess_Auth_Key;
        uint64                 Session_ID;
        uint32                 FA_IP;
        uint32                 Nonce32;
} Mobile_Respond;
```

6.3.5 Communication Reply Message

```
struct {

        Authentication_Key       Ses_Auth_Key;
        uint32                 Supp_Alg;
        uint32                 Nonce_CN;
        uint64                 Session_ID;
} PayLoad_K;

struct {
        uint8          Length;  /*  the value is set to the length of KValues in octets */
        Payload_K   KValues; /* the value is encrypted with KuCN */
} K;

struct {

        K              KValue;
        L              LValue;

}  Communicati_Reply_Message;
```

7. Mode of operation and encryption methods

7.1 Nodes authentication procedure

7.1.1 Encryption Algorithm for the PKI structure

In the authentication are used PKI structure for encryption decryption with public private key. The algorithms ,which should be used are **RAS** or **ECC** algorithms. This two algorithm are also the main used algorithms in practice. We believe that the future belongs the ECC algorithm as stronger with smaller key length.

Cipher	"PK_Key_Typ"	Description	Key Size bit

RSA_512	1	RSA encryption method with fixed key sized.	512
RSA_768	2	See RAS_512	768
RSA_xxx	3	RSA with flexible key size	xxx
ECC_113	4	Elliptic Curve encryption algorithm with fixed key sized	113
ECC_163	5	Elliptic Curve encryption algorithm with fixed key sized	163
ECC_xxx	6	Elliptic Curve encryption algorithm with flexible key sized	xxx

In the case that is flexible size is choose, than the field “KeyLen” should follow “KeyTyp” as described in chapter 6.

7.1.2 Mode of Operation

Because RSA and ECC work with fixed sized blocks, which size depends of the key size and secondary parameters chosen for the algorithm. The protect whole flexible lengthe size text should be chousen a some kind cipher mode of operation. There are four basics possibilities for **modes of opperation** as known, but only *chipher-block chaining* **CBC** can operate and gives the needed security ,because he use the same algorithm for encryption-decryption and chains logically the blocks and is also *self-synchronising* mode.

7.1.3 Packet Authentication

For the Message Authentication the phase ‘first contact’ are used in MIP recommended algorithms[1].The default authentication algorithm uses keyed-MD5 in "prefix+suffix" mode to compute a 128-bit "message digest" of the registration message. The default authenticator is a 128-bit value computed as the MD5 checksum.

The SPI value in the authentication extention defined in MIP as:

Security Parameter Index (SPI) An index identifying a security context between a pair of nodes among the contexts available in the Mobility

Security Association. SPI values 0 through 255 are reserved and MUST NOT be used in any Mobility Security Association.

In MVPN for SPI are use values specifying the authentication algorithm and the modes of operation.

Hash function	Assigned value for SPI	Description	Key size	MAC size
CBC_DES_SHA1	1	The MAC value is calculated with Hash function SHA1, for keying is used CBC mode of DES	56	160
CBC_DES_MD5	2	The MAC value is calculated with Hash function MD5 for keying is used CBC mode of DES	56	128
ECM_DES_MD5_64	3	The MAC value is calculated with Hash function MD5 for keying is ECM of DES (there is only one block)	56	64
HMAC_SHA1	4	It’s used HMAC algorithm with SHA1 with key size 128 bits	128	160
HMAC_MD5	5	It’s used HMAC algorithm with MD5 with key size 128 bits	128	128

7.2 Diffie-Hellman key exchange

There are two possiblities for DH exchange algorithm. MVPN can use the both of them.

Key exchange Suite	“DH_Alg”	description	Key size bit
DH_512	1	Classical Diffie-Helman exchange algorithm	512
DH_768	2	Classical Diffie-Helman exchange algorithm	768
ECDH_113	3	Elliptic Curves Diffie-Hellman exchange algorithm	113

ECDH_131	4	Elliptic Curves Diffie-Hellman exchange algorithm	131

When the nodes use the classical Diffie-Hellman exchange they are free to chose the public parameter. We recommend to use inpruved base public values ,which will garantie for strong keys.

For Elliptic Curve are recommended to be used to following parameter.

Basic	Yes
Field size	113
Irreducible polynomial	$x^{113}+x^{9}+1$
Elliptic curve E	$y^{2}+xy = x^{3} +ax^{2}+b$; over GF 2^{113}
Seed	10E723AB 14D69E6 76875615 1756FEBF 8FCB49A9
Parameter a	003088 250CA6E7 C7FE649C E85820F7
Parameter b	00E8BE E4D3E226 0744188B E0E9C723
Generating point G	009D73 616F35F4 AB1407D7 3562C10F 00A528 30277958 EE84D131 5ED31886
Order of G	010000 00000000 00D9CCEC 8A39E56F
Factor K	02

Basic	Yes
Field size	163
Irreducible polynomial	$x^{163} + x^{8} + x^{2} + x +1$
Elliptic curve E	$y^{2}+xy = x^{3} +ax^{2}+b$; over GF 2^{163}
Seed	D2C0FB15 760860DE F1EEF4D6 96E67686 56151754
Parameter a	07 2546B543 5234A422 E0789675 F432C894 35DE5242

Parameter b	00 C9517D06 D5240D3C FF38C74B 20B6CD4D 6F9DD4D9
Generation point G	07 AF699895 46103D79 329FCC3D 74880F33 BBE803CB 01 EC23211B 5966ADEA 1D3F87F7 EA5848AE F0B7CA9F
Order of G	04 00000000 00000000 0001E60F C8821CC7 4DAEAFC1
Cofactor K	02

7.3 Encryption algorithms and modes of operation , Mac building in “data flow protection” phase

The value “Supp Alg” - supported algorithms is build from two parts.

- Supported encryption modes and algorithms. “Supp EncrAlg” flag has 8 bit size.
- Supported authentication algorithms. “ ” flag has also 8 bit size.

Supp Alg = Supp Encr Alg || Supp Auth Alg

“Supp Alg” has 16 bit size.

The exact Values of the flags are calculated using **logical bit OR** function of all supported algorithms of the node. If one node receives a “Supp Alg” flag ,he should build his own “Supp Alg” flag referring to his computational power and then build the flag for common supported algorithms with **logical bit AND** of received flag and his own. If it’s needed he should forward the resulted flag to third Mobile Node to inform him about the supported algorithms of the nodes.

All encryption algortims are used in **CBC mode** ,because of his functionality and popularity.

cipher	“Supp Encr Alg” flag bit	Description	Key size
RC5_CBC_56	1	The use of RC5 algorithm in CBC mode	56
DES_CBC_40	2	Simple DES algorithm	40
DES_CBC_56	4	Normal DES in CBC Mode	56
3DES_CBC	8	Triple DES algorithm	168

BWF_CBC	16	Blowfish encryption algorithm	128
CAST_CBC	32	CAST-128 encryption algorithm	128
IDEA_CBC	64	IDEA algorithm	128

Authentication algorithms for building MAC functions in Dataflow protection transfer

Hash function	“Supp Auth Alg” flag	Description	Key size	MAC size
CBC_DES_SHA1	1	The MAC value is calculated with Hash function SHA1, for keying is used CBC mode of DES	56	160
CBC_DES_MD5	2	The MAC value is calculated with Hash function MD5 for keying is used CBC mode of DES	56	128
ECM_DES_MD5_64	4	The MAC value is calculated with Hash function MD5 , then is truncated to 64 bits, for keying is ECM of DES (there is only one block)	56	64
HMAC_SHA1	16	It’s used HMAC algorithm with SHA1 with key size 128 bits	128	160
HMAC_MD5	32	It’s used HMAC algorithm with MD5 with key size 128 bits	128	128
EBM_BWF_MD5	64	The MAC value is calculated with Hash function MD5 , for keying is used Blowfisch 128 (there is only one block)	128	128
DSS	128	Digital Signature Standard	PKI	160

8. Conclusion

Virtual Private Networks have proved to be an acceptable solution for the implementation of secure communications between private networks that need to use a public network as a communication path. There have been several proposals for the creation of such VPNs and they seem to be working according to the requirements of each application.

These proposals, including IPsec, were created thinking on a fixed environment and even though they have had good results on some of their applications, in others they do not allow the establishment of a secure environment without sacrificing the performance of the network.

One example is their use on a mobile environment. VPN mechanisms were thought also as global solutions when in practice it is necessary to create a specific solution for a specific situation based in the same foundations. IPsec provides two services and two different modes that offer several options for the implementation of a secure channel but it does not allow the modification of the protocol to obtain the 'best' of two worlds: performance and security for mobile environment.

Security protocols available nowadays can provide the same security level in a mobile environment, we are not discussing the effectiveness of such protocols only their applicability to mobile entities. Certainly IPsec can be used for the implementation of VPNs in a mobile world but it does not guarantee that its use will be the most optimal.

The advantages provided by a mobile protocol can be affected if other required processes are not the adequate, such is the case of authentication. Mobile environments brought with them different characteristics and more risks that need to be taken in consideration before choosing or implementing a security framework.

The protocol presented in this document proves the establishment of a Mobile Virtual Private Network by using a Public Key Infrastructure that will handle the authentication of the mobile agents during the registration of the Mobile Node. The model used describes the application of such MVPNs in private networks belonging to the same corporation, which makes feasible the implementation of such PKI and other features presented in this document, such as the use of SMART CARDS and the creation of public and private keys by the Home Agent.

MVPN relies on the cryptographic basis of encryption and authentication, providing mechanisms to protect against **man-in-the-middle** attacks, spoofing and some denial of service attacks. It also protects the home and foreign networks by authenticating the correspondent node. MPVN is a protocol optimised for:

- Small number of messages.
- Compatibility with Mobile IP.

- High security-high quality of Mobile Virtual Private Networks.
- Fast data transmission after the authentication procedure.
- Practical use and simplicity.

Mobile IP still presents some security breaches that cannot be covered with MVPN, other mechanisms need to be created to guarantee full protection against denial of service attacks to the home and Foreign Agents.

MVPN provides an authentication framework that can also be applied to the protocols such as IIP [9], it will be necessary to modify the protocol but it should be compatible with other MVPN implementations.

9. References

[1] Charles Perkins, editor. IP Mobility support. RFC 2002, October 1996.
[2] Stephen Kent and Randall Atkinson. Security architecture for the Internet Protocol. RFC 2401, November 1998.
[3] Stephen Kent and Randall Atkinson. IP Authentication header. RFC 2402, November 1998.
[4] Stephen Kent and Randall Atkinson. IP Encapsulating Security Payload (ESP). RFC 2406, November 1998.
[5] Yongguang Zhang, editor. The implication of end-to-end IPsec. Internet-draft, March 2000.
[6] Ferguson Neils and Schneier Bruce. A Cryptographic Evaluation of IPsec. Counterpane Internet Security, Inc.
[7] Secure Mobile Networking Project. Final Report. University of Portland, June 1999.
[8] Rivest, R. The MD5 Message-Digest Algorithm, RFC 1321, April 1992.
[9] Nam Yap, C. et al. Itinerant Internet Protocol, draft, June 2000.

Questions about this document can also be directed to the authors:

Vesselin Tzvetkov
Centre for Mobile Communications
Department of Electronic and Electrical Engineering
University of Sheffield
Regent Court
211 Portobello Street
Sheffield S1 4DP, England
Phone: +44 114 222 2108
Fax: +44 114 222 8299
E-mail: tzvetkov@europe.com
Web : http://go.to/vesselin

Erika Sanchez
Centre for Mobile Communications
Department of Electronic and Electrical Engineering
University of Sheffield
Regent Court
211 Portobello Street
Sheffield S1 4DP, England
Phone: +44 114 222 2108
Fax: +44 114 222 8299
Web: http://www.dcs.shef.ac.uk/~esanchez
E-mail: E.Sanchez@dcs.shef.ac.uk