PERIODICA POLYTECHNICA SER. EL. ENG. VOL. 48, NO. 1–2, PP. 85–100 (2004)

MODELLING LOCATION REVEAL ATTACKS IN MOBILE SYSTEMS

László ZÖMBIK* and Levente BUTTYÁN**

*Ericsson Hungary, Budapest
Department of Telecommunication and Mediainformatics
Budapest University of Technology and Economics
e-mail: laszlo.zombik@ericsson.com
**Laboratory of Cryptography and System Security (CrySyS)
Department of Telecommunications
Budapest University of Technology and Economics
e-mail: buttyan@crysys.hu

Received: Dec. 3, 2003

Abstract

We propose a novel approach for the modelling and discovery of location reveal attacks in mobile environments. Our approach is based on the theory of Communicating Sequential Processes (CSP). We demonstrate the power of our approach by analysing the MIPv4 protocol and by showing that it does not protect the location information of the mobile node appropriately.

In order to solve this problem we specify which communications should be encrypted within MIPv4. The so specified protocols were verified using our CSP-based model, and they were found secure.

Keywords: location hiding, formal analysis, CSP, IP mobility

1. Introduction

Using the Internet more widely for exchanging information, security requirements have also gained importance. Users nowadays would not only like to communicate, but they also would like to do it securely. Therefore, certain information is not intended to be available or disclosed to unauthorized entities. Furthermore, communicating peers have to be identified unambiguously. Besides, it must be ensured that information has not been changed, destroyed, or lost in an unauthorized or accidental manner.

With the deployment of IP mobility protocols [4, 5, 6], the binding of the identity to the location of the nodes has disappeared. This makes it possible to achieve an additional security service: the protection of location of the users. Location hiding has already settled in mobility systems, like GSM, UMTS or GPRS.

In an IP environment, where an honest communication should take place through hostile networks, keeping the location of the nodes unrevealed is not an easy task. An eavesdropper can listen anywhere in the network, or several of them can join and they even can monitor the entire honest communication.

On the one hand, this means that the attacker should not be able to find location-related information by analysing the content of IP packets. On the other hand, he should also be unable to deduce location-related information by statistical analysis of the network [7].

In this paper, we focus on the formal requirement and we give a procedure based on a formal verification technique to validate whether an attacker, which eavesdrops at several points in the network, is capable to identify the location of honest nodes.

Several formal methods exist that can be used to verify the security properties of a system [9, 10, 12, 11]. In this paper we use a CSP-based approach. CSP has already been used to verify secrecy and authenticity properties of key exchange protocols [1] as well as to analyse fair exchange protocols [13]. However, to the best of our knowledge it has never been used to verify properties related to location hiding in a mobile system. As for the mobile system we demonstrate our approach on the Mobile IPv4 protocols, but we note that with slight modifications, it can be applied to any mobility protocol such as MIPv6, GPRS, UMTS, etc.

The outline of the paper is the following: In Section 2 a short introduction to the CSP language is presented. In Section 3 the operation of the Mobile IPv4 protocol is briefly summarized. Section 4 is concerned with the general CSP model of a mobile IP system. Finally in Section 5, the location-hiding criteria and their formal verification in a Mobile IPv4 system are discussed.

2. Introduction to CSP

Concurrent systems, in which components can interact with each other, can be described and analysed using the CSP (Communicating Sequential Processes) [3] language.

In CSP, system components called processes communicate by events. Events consist of atomic structures. A typical event is denoted by $c.i.j.m$, which states that on communication channel c, i has sent message m to j. The related process behaviour can be described by the constructs $c!m \rightarrow P$ and $c?a : T \rightarrow P(a)$. The former means that the process initially sends out message m on channel c, and then it behaves as process P. The latter describes that the process waits for input $a \in T$ on channel c and once it received it, behaves like $P(a)$. STOP and SKIP are the most trivial processes, the former produces no traces, while the latter expresses that a process successfully terminated.

A process can be combined via various operators to obtain new processes. $Q = P \Box R$ means that Q can behave as either P or R. $P \|_A R$ means that P and R can work in parallel, however they have to synchronize on any event $a \in A$. Thus, they may have to wait for each other, and they must perform the event jointly. P and $R \mathbin{|||} R$) if there is no event the two processes interact. The hiding operator $P \backslash R$ or $P \backslash \{a\} \mid a \in \Sigma$ (where Σ is the set of all possible events) means that the corresponding process behaves as P, except that properties of R are missing, or events of $\{a\}$ do not exist.

The renaming operator applies a mapping ϕ to change an action into another. Thus, $P[\![\phi]\!]$ behaves like P, except that all visible events $\{a\}$ from P are relabelled by whatever ϕ associates with a. E.g., $P[\![a/b]\!]$ means that event a of process P are renamed to event b.

The guarded alternative $b\&P$ only allows a process to behave like P if the Boolean expression b is true, otherwise the process becomes STOP (i.e, $b\&P = P$ if b else STOP).

Let traces (P) be the set of finite sequences $(\langle a_i \rangle)$ of events that a process P possibly generates. The traces refinement $\sqsubseteq_T$ is defined between two processes in such a way that traces $(R) \subseteq$ traces $(P) \Leftrightarrow P \sqsubseteq_T R$.

3. IP Mobility

There are several protocols and systems, which can provide transparent connectivity to the Internet while the mobility of nodes is also ensured. The GPRS and UMTS systems offer mobility via their radio networks. Protocols in the lower layers handle the mobility issues, and the upper layer protocols grant IP communication.

There are IP-based micro-mobility protocols that handle the mobility of a small area efficiently. The IP macro-mobility protocols are used to handle global movements, and they do not necessarily use radio medium to achieve mobility.

The Mobile IPv4 and Mobile IPv6 protocols can be used for handling IP mobility without the need of any support from the lower layers (e.g. radio network).

They consist of two phases: the registration and the data communication phase. In the registration phase, the mobile informs its partners about its location change. After this, the data communication phase can be initiated. If the data communication is optimized, then triangle routing (v4) or route optimization (v6) is used. Otherwise, reverse tunnelling (v4) or return routability (v6) is employed.

Nowadays, the MIPv4 is used most commonly, therefore, our work is focused on this protocol.

3.1. The Mobile IPv4 Protocol

A MIPv4 system [4] consists of the Home Agent (HA) and the Foreign Agent (FA), which handle the mobility issues, a Mobile Node (MN), which moves from one location to another and a Correspondent Node (CN), which the MN communicates.

At the beginning, the MN registers with the new attachment point. If the FA is involved in the registration, then the MN sends a Registration Request (RReq) to the FA, including MN-HA, and MN-FA Authentication extensions (MNHAAuth1, MNFAAuth1), if they exists. The FA processes the request, it may attach an FA-HA Authentication extension (FAHAAuth1) and forwards it to the home network. The HA answers with a Registration Reply (RRep), which grants or denies the

Request, including MN-HA and FA-HA authentication extensions (MNHAAuth2, FAHAAuth2). The FA processes the Registration Reply and then relays it (it may attach an MN-FA authentication extensions, MNFAAuth2) to the MN to inform it about the disposition of its Request. In a simplified model of MIPv4, messages contain the source and destination IP addresses, the HA and the home IP addresses (MNIP) of the MN. The registration phase has two variants depending on whether the FA is involved or not. If the FA is involved, then the protocol is the following:

I.1. MN → FA: RReq: MNIP, FAIP, HAIP, MNHAAuth1, MNFAAuth1
I.2. FA → HA: RReq: FAIP, HAIP, MNIP, MNHAAuth1, MNFAAuth1
I.3. HA → FA: RRep: HAIP, FAIP, Result, MNIP, MNHAAuth2, FAHAAuth2
I.4. FA → MN: RRep: FAIP, MNIP, Result, HAIP, MNIP, MNHAAuth2, FAHAAuth2

If the FA is not involved in the registration, then the protocol has the following steps:

II.1. MN → HA: RReq: CCoA, HAIP, MNIP, MNHAAuth1
II.2. HA → MN: RRep: HAIP, CCoA, Result, MINP, MNHAAuth2

where CCoA is the co-located care of address, which is the temporary IP address that the mobile receives in the foreign network.

In the data communication phase, the IP packets travel in triangle between CN, HA and MN. However, if firewalls and ingress filtering are used, reverse tunnelling is used instead of triangle routing.

Both triangle routing and reverse tunnelling follow different procedures depending on whether the co-located or the Foreign Agent care of address is used. Hence, there are four different cases:

Triangle routing with co-located care of address

The communication when triangle routing is used consists of three steps. If the CN intends to communicate with the MN, then it addresses the MN by its home IP address. If the MN is away from the home network (the registration phase has been executed), then the HA grabs the message, encapsulates it, and sends it to the MN by using its co-located care of address. When the MN receives it, decapsulates the message. If the MN wants to send a message, then it directly addresses the CN.

III.1. CN → HA: Data: CNIP, MNIP, msg1
III.2. HA → MN: Data: HAIP, CCoA, {CNIP, MNIP, msg1}
III.3. MN → CN: Data: MNIP, CNIP, msg2

Reverse Tunnelling with co-located care of address

In reverse tunnelling, the MN does not address the CN, however, it uses the tunnel (encapsulates the messages) to the HA.

IV.1. CN → HA: Data: CNIP, MNIP, msg1
IV.2. HA → MN: Data: HAIP, CCoA, {CNIP, MNIP, msg1}
IV.3. MN → HA: Data: CCoA, HAIP, {MNIP, CNIP, msg2}
IV.4. HA → CN: Data: MNIP, CNIP, msg2

Triangle routing with FA care of address

In this situation the FA is installed between the MN and the HA, in such a way that the FA becomes the endpoint of the tunnel.

Triangle routing is modified, compared to the co-located care of address:

V.1. CN → HA: Data: CNIP, MNIP, msg1
V.2. HA → FA: Data: HAIP, FAIP, {CNIP, MNIP, msg1}
V.3. FA → MN: Data: CNIP, MNIP, msg1
V.4. MN → CN: Data: MNIP, CNIP, msg2

Reverse Tunnelling with FA care of address

The difference in reverse tunnelling compared to the reverse tunneling of co-located care of address is the following. The HA does not send any message directly to the MN, however, it encapsulates and sends them to the FA. The FA decapsulates, then forwards the message to MN. Messages from the MN use the same communication path.

VI.1. CN → HA: Data: CNIP, MNIP, msg1
VI.2. HA → FA: Data: HAIP, FAIP, {CNIP, MNIP, msg1}
VI.3. FA → MN: Data: CNIP, MNIP, msg1
VI.4. MN → FA: Data: MNIP, FAIP, {CNIP, MNIP, msg2}
VI.5. FA → HA: Data: FAIP, HAIP, {CNIP, MNIP, msg2}
VI.6. HA → CN: Data: MNIP, CNIP, msg2

4. General CSP Model of a Mobile IP

The general CSP model of a mobile IP system for formal verification of security properties can be divided into the specification part and the system description part (*Fig. 1*). In the specification part, the expected operation of the system is formalized, while the system description part models the operation of the actual system. If these two blocks behave equivalently, then the system conforms to its specification. This can be verified by analysing their traces. If the mobile system behaves properly, then the system is trace refined to its specification. Thus: specification $\sqsubseteq_T$ system.

The next subsection covers the general model of the mobile system, and Subsection 4.2 is concerned with the secret specification. In Section 5.1 we show how the secret specification is used in the specification of location-hiding.

4.1. The Mobile System

In the CSP-model of the MIPv4 system, each component (also called node) corresponds to the protocol participant (i.e. FA, HA, CN, MN). In addition, a special system component represents the attacker. These components communicate with each other via channels. In our model, each component has a *send* and a *receive* channel. The honest nodes use the send channel to send messages and the receive channel to receive messages. The attacker has the ability to inject, delete and modify messages sent by the honest participants. We model this ability of the attacker by allowing the attacker node in the model to access the *send* and *receive* channels of the honest nodes. More precisely, in our model, the attacker receives messages on the *send* channels and sends messages on the *receive* channels (see *Fig. 1*). This corresponds to the well-known DOLEV–YAO model [1].

In addition to the *send* and *receive* channels, we introduce some special channels for technical reasons. These are the *leak* and the *ClaimSecret* channels. The *leak* channel is used by the attacker to signal that a secret or other useful information is obtained. In this case the message contains information about the location-related facts, from which the attacker has learnt the position of the MN. These facts can be secrets, or they can be information about identification or location of the MN. The *ClaimSecret* channel is used to express that a data-element is a secret of a node, and it should not be obtained by the attacker:

channel send:	ALL_NODE.ALL_NODE.MSG
channel receive:	ALL_NODE.ALL_NODE.MSG
channel leak:	{Attacker}.{Env}.MSG
channel ClaimSecret:	ALL_NODE.{Env}.SECRET

where Attacker and Env are the processes of the attacker and the environment, respectively. The environment is a special process to which the attacker sends its *leak* signals. ALL_NODE means the set of nodes (including the Attacker) and SECRET is the information which is obtained or should be obtained by the attacker.

Messages sent on channels are built from atomic components, according to the following scheme:

$$\begin{aligned}
\mathrm{MSG} &= \mathrm{ATOM}|\mathrm{KEY(MSG)}|\mathrm{MSG.MSG}.\\
\mathrm{Key} &= \emptyset,\\
\mathrm{Atom} &= \mathrm{IPADDRESSES}|\mathrm{KEY}|\mathrm{msg1}|\mathrm{msg2}|\mathrm{MNFAAuth1}|\\
&\qquad \mathrm{MNFAAuth2}|\mathrm{MNHAAuth1}|\mathrm{MNHAAuth2}|\mathrm{garbage},\\
\mathrm{IPADDRESSES} &= \mathrm{CNIP}|\mathrm{MNIP}|\mathrm{FAIP}|\mathrm{HAIP}|\mathrm{CCoA},
\end{aligned}$$

The behaviour of the MN is the following. First, it tries to register by sending its registration request. According to the reply, either a new request is initiated or the MN checks the validity of the authentication extension. If the authentication is correct, the MN is capable to send and receive data, otherwise tries to register again. Once registered, the MN sends data according to triangle routing or reverse tunneling. In our model, the above described behaviour of the MN, when triangle routing is used, is encoded as follows:

$$\begin{aligned}
&\text{MN\_INIT} = \text{send!MNFA.RReq} \rightarrow \\
&\quad \Box_{\text{FA:ALL\_NODE}}\text{receive? FA.M.N.RRep} \rightarrow \\
&\qquad \Box(\text{Result == accepted}) \wedge (\text{MNHAAuth is correct})\&\ \text{MN\_DATA MN\_INIT} \\
&\text{MN\_DATA} = \\
&\quad \text{send!MN.CN.msg2} \rightarrow \text{MN\_DATA} \\
&\qquad \Box_{\text{FA:ALL\_NODE}}\text{FA.M.N.msg1} \rightarrow \text{MN\_DATA}
\end{aligned}$$

The behaviour of the other honest nodes is similar. The behaviour of the attacker, however, is very much different from the behaviour of the honest nodes.

The attacker can listen into the communication of the honest nodes. Based on the eavesdropped information, he can deduce new knowledge and build new or altered messages and try to mislead honest nodes. The main goal of the attacker is to obtain secrets or location-related informations in the communication.

The full description of the attacker is beyond of scope of this paper, however, the interested reader is referred to [1].

The attacker has a certain initial knowledge (S_0). He eavesdrops new messages and deduces new knowledge (S_1). Thus, the actual knowledge of the attacker is: $S = S_0 \cup S_1$.

The deduction operator $\nu \vdash \epsilon$ is defined as: Let be ν the set of facts, from which a new set of fact ϵ can be deduced. If the attacker knows ν, then he can expand his new actual knowledge S' with ϵ, so: $\nu \subseteq S \Rightarrow S' = S \cup \epsilon$.

The deduction rules are the following:

$$\begin{aligned}
&m \in S \Rightarrow S \vdash m \\
&S \vdash m \wedge S \subseteq S^* \Rightarrow S^* \vdash m \\
&k \in \text{KEY}, m \in \text{MSG} \Rightarrow k, m \vdash k(m) \\
&k \in \text{KEY}, k(m) \in \text{MSG} \Rightarrow k, k(m) \vdash m \mid m \in \text{MSG} \\
&k1 \in \text{KEY}, k(m) \in \text{MSG}, k \neq k1 \Rightarrow k1, k(m) \vdash \{\text{garbage}\} \\
&a \in \text{ATOM} \setminus \{\text{SECRET}\} \Rightarrow \emptyset \vdash a
\end{aligned}$$

The honest part of the Mobile IPv4 system (SYSTEM_HONEST) consists of the CN, HA, FA, and MN, and they work independently:

$$\text{SYSTEM\_HONEST} = (MN \ ||| \ \text{FA} \ ||| \ \text{HA} \ ||| \ \text{CN}).$$

The full system contains the attacker too:

$$\text{SYSTEM} = (\text{SYSTEM\_HONEST} \|_{\text{send, receive}} \text{ATTACKER})$$

Note that, according to the Dolev-Yao model, the attacker has access to all communications, thus, MN, CN, HA, FA do not communicate with each other directly,

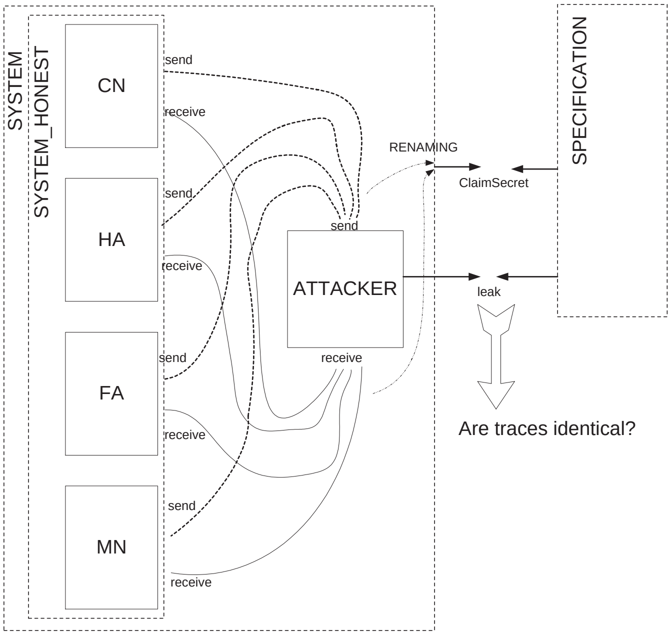

Fig. 1. Trace refinement of the mobile system against its security specifications

but only via the attacker. In our model this is expressed by the honest part of the system, is set to work parallel with the attacker in such a way that they synchronize events on the *send* and *receive* channels.

4.2. Secrecy Requirements

We want to use our model defined above to verify whether or not protocol-related information carried by the protocol messages remain secret (unknown) to the attacker. Therefore we need to define formally what we mean by a data element being secret to the attacker.

In our model we consider a data element secret if it is claimed to be secret, and the attacker can not leak it in his leak channel. Thus, if a ClaimSecret!a.Env.$\sigma\,|\sigma \in$ SECRET, $a \in$ HONEST message is sent by the honest nodes, then the attacker must not send a leak.Attacker.Env.$\sigma\,|\sigma \in$ SECRET message.

This can be expressed as a trace specification:

$$\begin{aligned}&\text{SECRET\_SPEC}_{a,b}(\sigma)\\&\quad = |||_{\forall\sigma\in\text{SECRET}}\text{ClaimSecret!a.Env.}\sigma \in \text{trace}|a \in \text{HONEST} \Rightarrow\\&\quad\neg(\text{leak.Attacker.Env.}\sigma \in \text{trace})\end{aligned}$$

Thus the task is to compare whether the traces of SYSTEM refine to its specification SECRET_SPEC.

However, the complex process SYSTEM does not have ClaimSecret channel. In the ClaimSecret channel a signal has to be sent before the first message is to be emitted.

This means that ClaimSecret!a.Env.σ message either can be added to each node as the initial message, or the first message of each node in SYSTEM can be renamed to it. We utilize the advantage of the renaming operator, thus the model of the mobile protocols remain undistorted.

Since only the $leak$ and $ClaimSecret$ events are important to the refinement, the unimportant events are hidden:

$$\begin{aligned}&\text{SYSTEM\_S}\\&\quad = \text{SYSTEM}[\![\text{first message}_a/\text{ClaimSecret!a.Env.}\sigma_{\forall\sigma\in\text{SECRET}}|\forall a \in \text{HONEST}]\!]\\&\qquad \backslash(\text{Events}\backslash\{\{\text{leak}\}, \{\text{ClaimSecret}\}\}).\end{aligned}$$

The mobile system is secure if:

$$\text{SECRET\_SPEC}(\forall\sigma \in \text{SECRET}) \sqsubseteq_T \text{SYSTEM\_S}.$$

5. Location Hiding

The definition of location privacy relies on the fact that some information is bounded to the location of the user (e.g. in cellular systems the network ID, area, cell ID carries this information). Some information identifies the mobile node unambiguously (e.g. in cellular networks, the IMSI number).

A trivial solution to keep the location of the MN hidden is that either *location information* or *identification information* of the communication is kept secret.

If the above information are well-known to the attacker, but only the *logical binding* between the location and identification information is unknown, then the location of the target still cannot be determined.

More specifically, in the first case, the attacker identifies that the MN participates in the communication, however, he cannot localize the MN. For example if the attacker eavesdrops an IP packet of communication between the CN and HA, which contains the home IP address of the mobile, then the attacker can only state that the mobile has been participated in the communication, but none of the endpoints are the MN.

In the second case, the attacker identifies a node, which is the originator or destination of a communication, however he cannot identify as his target, the MN. Since in the Internet there are plenty nodes communicating with each other, therefore, when the attacker eavesdrops a communication and localize its endpoints, he can not state that the endpoint of this communication is the MN.

In the third case, the attacker is only capable to find out the location, if he realises the fact that the MN was participated in the communication, and he must obtain another fact which is related to the position. Furthermore he has to prove the binding, so he has to prove that the position information belongs to the MN. For example consider the following case. The attacker has eavesdropped a message, whose destination is the CN, and the originator is an unknown node, (so not the HA) and its location is determined. Furthermore the message contains the home IP address of a Mobile node as a source. In addition the attacker knows somehow that the MN uses triangle routing with co-located care of address. In this case the attacker knows that the MN is involved in the communication, he knows the location of the unknown node. Furthermore he knows that the unknown node is not the HA, so in case of triangle routing with co-located care of address it must be the MN.

In our approach we supposed that the *identification information* of the MN is the MNIP.

We further supposed that there are three ways to determine the *location information* of the MN. In the first case the position information of any node can be determined if the node emitted or received a packet, and the attacker had heard it.

In the second case the foreign agent care of address and in the third case the co-located care of address determine the position information of the MN. The reason is that the messages are routed towards the mobile using those addresses. In addition, in case of reverse routing the MN uses those addresses to send messages. However this not necessarily means the location of the MN is revealed, if the attacker is unable to eavesdrop the MNIP, then the identified communication may belong to other mobiles, which are served by the same HA.

In the above, the attacker cannot prove the *logical binding*. Assuming the location and identification information above, there are three bindings. The first one is, when a mobile sends or receives a message which contains the MNIP, while the second binding is when the attacker eavesdroped somewhere a message which contained both the the MNIP and FAIP, and the third logical binding is that the MNIP and the CCoA is found in the same message.

Summary of the cases in which the location remains protected, is the following. Either the MNIP should be kept secret or the CCoA and FAIP are kept secret, and the attacker also does not communicate with the MN. If the above statements cannot be assumed, then the location of the MN is unrevealed if the following two statements are valid. The attacker is unable to find a message in which either the MNIP and the FAIP; or the MNIP and CCoA message elements are included. The attacker is unable to find any message which is sent from or to the MN and in which the MNIP is included.

5.1. Location Reveal Attacks on Mobile IPv4 Protocol Using CSP

Our CSP-based model is only capable to find location-related attacks, in which the attacker analyses the content of the IP packets, and further knows the originator and the destination of the message.

If we want to verify whether the location is protected, let us consider first the trivial case. The user location is protected, if the identification information remains secret. Thus the MNIP remains secret: SECRET_SPEC(MNIP) $\sqsubseteq_T$ SYSTEM_S. Unfortunately, this is not true, the attacker can receive the MNIP.

However, if the attacker does not receive any single message from the MN and he is not able to collect care of addresses, then the location of the MN still remains unrevealed:

(SECRET_SPEC(FAIP) $\sqsubseteq_T$ SYSTEM_S)$\wedge$
(SECRET_SPEC(CCoA) $\sqsubseteq_T$ SYSTEM_S)$\wedge$
(send!MN.a.msg $|\notin$ traces(SYSTEM_HONEST), a
$\in$ HONEST,msg $\in$ MSG)$\wedge$
(receive?a.MN.msg $|\notin$ traces(SYSTEM_HONEST), a
$\in$ HONEST,msg $\in$ $MSG)$.

Unfortunately, in MIPv4 the FAIP, CCoA are sent unprotected, and also the MN sent a message, which the attacker eavesdropped. Therefore the above cases do not result in a hidden location of the MN.

When the MNIP, CCoA and FAIP can be obtained by the attacker, and even the MN communicates directly to him, there is only one case, when the MN location remains unknown. According to our attacker model, this case is, when the attacker is not able to find any message originated from MN and containing MNIP, and besides, he is unable to find a message where the MNIP and the care of address of the MN appears.

The former condition is equal to:

(send!MN.a.msg | a $\in$ ALL_NODE,MNIP $\in$ msg)
$\notin$ trace(SYSTEM_HONEST)$\wedge$
(receive?a.MN.msg | a $\in$ ALL_NODE,MNIP $\in$ msg)
$\notin$ trace(SYSTEM_HONEST)

and the latter corresponds to:

send!a.b.msg | a, b $\in$ HONEST,MNIP $\in$ msg, (FAIP $\in$ msg $\vee$ CCoA $\in$ msg)
$\notin$ trace(SYSTEM_HONEST)

Thus we modified the attacker model, using the guarded alternative. When the above rules are violated, the attacker sends a *leak* signal:

$\Box_{a,\mathrm{msg}}$ send?MN.a.msg | $a \in$ ALL_NODE,MNIP $\in$ msg &
leak.Attacker.Env.(MN,MNIP)
$\Box_{a,\mathrm{msg}}$ receive!MN.a.msg | $a \in$ ALL_NODE,MNIP $\in$ msg &
leak.Attacker.Env.(MN,MNIP)
$\Box_{a,b,\mathrm{msg}}$ send?a.b.msg | $a, b \in$ HONEST;MNIP,FAIP $\in$ msg &
leak.Attacker.Env.(MNIP,FAIP)
$\Box_{a,b,\mathrm{msg}}$send?a.b.msg | $a, b \in$ HONEST; MNIP,CCoA $\in$ msg &
leak.Attacker.Env.(MNIP,CCoA)

Specification of the secure system where the attacker is unable to find any logical bindings between A and B:

$$\text{SPEC\_NOBIND}(A, B) = \neg\text{leak.Attacker.Env.}(A, B).$$

The system should be trace refined as:

$$\text{SPEC\_NOBIND(MN,MNIP)} \sqsubseteq_T \text{SYSTEM},$$
$$\text{SPEC\_NOBIND(MNIP,FAIP)} \sqsubseteq_T \text{SYSTEM},$$
$$\text{SPEC\_NOBIND(MNIP,CCoA)} \sqsubseteq_T \text{SYSTEM}.$$

5.2. *Hiding the Communication of Foreign Networks*

In the Dolev–Yao model, an attacker has access to the communication channels of the honest users everywhere. Thus, the location-related information, which is usually in cleartext in the local network, is easily obtained. We found with our model checker that the MNIP and the care of addresses were obtained by the attacker in the foreign network.

Besides, it is unlikely that the attacker has access to all foreign networks. Therefore, in our next approach, communication between the foreign network and the MN must be hidden from the attacker. This can be achieved by using the hiding operator.

Let R be the neighbour of user (the target, the location of which must be protected) U, and let X_i be an honest node such $X_i \neq R, U$. Then the system becomes:

$$\begin{aligned}&\text{SYSTEM\_HONEST}\\ &\quad = |||_i\{X_i\}\ |||\ (R\|_{\{\{\text{send}.U.R\},\{\text{receive}.U.R\},\{\text{send}.R.U\},\{\text{receive}.R.U\}\}} U)\\ &\qquad \setminus \{\{\text{send}.U.R\}, \{\text{receive}.U.R\}, \{\text{send}.R.U\}, \{\text{receive}.R.U\}\})\end{aligned}$$

These modifications are applied to the analysis of MIPv4 as follows.

In the co-located care of address case the system contains no FA, therefore, let the border router to the MN be R. The mobile system is drawn in *Fig. 2*:

$$\text{SYSTEM} = (\text{CN} \mid\mid\mid HA \mid\mid\mid \text{SYSTEM\_0}) \parallel_{\text{send},\text{receive}} \text{ATTACKER},$$

where

$$\text{SYSTEM\_0} = \big((R \parallel_{\{\text{send}.MN.R\},\{\text{send}.R.MN\},\{\text{receive}.MN.R\},\{\text{receive}.R.MN\}} MN) \setminus$$
$$\{\{\text{send}.MN.R\}, \{\text{send}.R.MN\}, \{\text{receive}.MN.R\}, \{\text{receive}.R.MN\}\}\big)$$

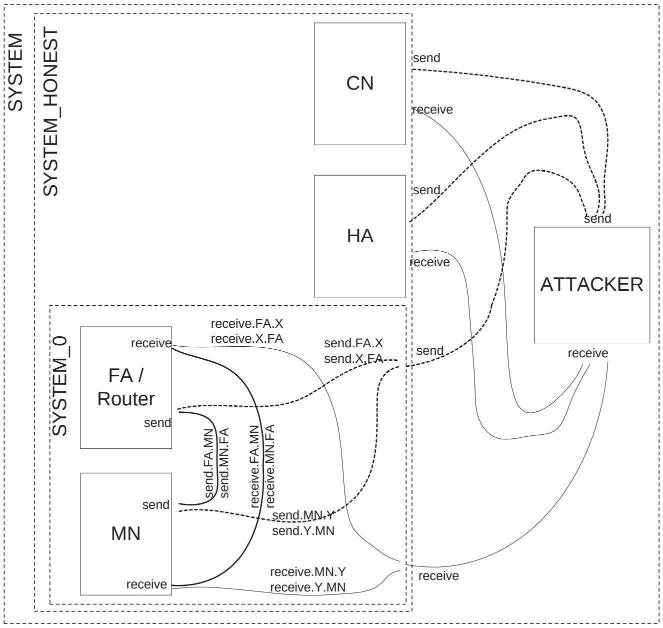

Fig. 2. Outline of a MobileIPv4 system.

The system is described in a similar way when FA care of address is used. The only difference is that in that case $R = FA$.

With this model, attacks are found in the registration phase, since all messages between HA and FA are sent unprotected and CCoA or FAIP and MNIP message elements are revealed. For the same reasons, this happens when reverse tunnelling is used. Furthermore, at triangle routing, the mobile sent a message toward the CN and it contained the MNIP.

Note that the hiding operator can be replaced by renaming operator. In this case, the messages which are required to be protected, are renamed as they became meaningless for the attacker.

5.3. *Using Encrypted Channels*

From the previous section, it is obvious that the location privacy in MIPv4 cannot be achieved without encrypting the communication of the user. Encryption is supported by IPSec [8] protocol.

Therefore let $k_{FA,HA}$ and $k_{MN,HA}$ be the encryption keys of the tunnelled communication, when FA and CCoA are used, respectively.

Using FA care of address, the registration phase does not leak the location of the MN:

I.1. MN → FA: MNIP, FAIP, HAIP, MNHAAuth1, MNFAAuth1

I.2. FA → HA: FAIP, HAIP, $k_{\mathrm{FA,HA}}$ (MNIP, MNHAAuth1, FAHAAuth1)

I.3. HA → FA: HAIP, FAIP, $k_{\mathrm{FA,HA}}$ (Result, MNIP, MNHAAuth2, FAHAAuth2)

I.4. FA → MN: FAIP, MNIP, Result, HAIP, MNHAAuth2, MNFAAuth2

Using Foreign Agent care of address, the data transfer phase, if reverse tunnelling is used, also keeps the location of the user secret:

VI.1. CN → HA: CNIP, MNIP, msg1

VI.2. HA → FA: HAIP, FAIP, $k_{\mathrm{FA,HA}}$ (CNIP, MNIP, msg1)

VI.3. FA → MN: CNIP, MNIP, msg1

VI.4. MN → FA: MNIP, FAIP, {CNIP, MNIP, msg2}

VI.5. FA → HA: FAIP, HAIP, $k_{\mathrm{FA,HA}}$(CNIP, MNIP, msg2)

VI.6. HA → CN: MNIP, CNIP, msg2

Using Foreign Agent care of address, in the data transfer phase if triangle routing is used, unfortunately it reveals the location-related information, since the MN sends a message that contains MNIP to CN unprotected.

V.1. CN → HA: CNIP, MNIP, msg1

V.2. HA → FA: HAIP, FAIP, $k_{\mathrm{FA,HA}}$ (CNIP, MNIP, msg1)

V.3. FA → MN: CNIP, MNIP, msg1

V.4. MN → CN: MNIP, CNIP, msg2

The same weakness has been found, when co-located care of address was used. Thus, the registration phase in reverse tunnelling keeps the location secret, however, triangle routing leaks it.

Therefore the next step is to protect the data traffic between the MN and the CN. IPSec can be used between them, with key $k_{\mathrm{MN,CN}}$:

V.1. CN → HA: CNIP, MNIP, msg1

V.2. HA → FA: HAIP, FAIP, $k_{\mathrm{FA,HA}}$ (CNIP, MNIP, msg1)

V.3. FA → MN: CNIP, MNIP, msg1

V.4. MN → CN: MNIP, CNIP, $k_{\mathrm{MN,CN}}$ (msg2)

Unfortunately, in step V.4. the MN is the originator of the message, and the MNIP data field is also contained. This means that this encryption still provides no protection.

Therefore, when triangle routing is used, our proposal is to build an IPSec tunnel between either the FA or the border router (R) of the foreign network and CN, using keys $k_{\mathrm{FA,CN}}$ or $k_{\mathrm{R,CN}}$. (Note that instead of security gateways network address translator can also be used.)

In this case all traffic from MN to CN can traverse securely.

If the FA is the security gateway, then step V.4 is changed as:

V.4a. MN → FA: MNIP, CNIP, msg2
V.4b. FA → CN: FAIP, CNIP, $k_{FA,CN}$ (MNIP, CNIP, msg2)

and if a border router is the security gateway:

V.4a. MN → R: MNIP, CNIP, msg2
V.4b. R → CN: R, CNIP, $k_{R,CN}$ (MNIP, CNIP, msg2)

6. Conclusion

Hiding the location of nodes is a new security requirement that gains importance in mobile environments, especially in IP based mobile communication. In this paper we presented a formal approach based on CSP to verify protocols against this new security requirement. The essence of our model is to represent protocol participants as CSP processes that communicate over various channels. In addition to the honest participants, the attacker is represented as a CSP process too. In our attacker model, he can listen to all communication of the protocol participants, and modify any messages they exchange. The goal of the attacker is to obtain information, principally data element that reveals the location of the mobile node. The attacker is not allowed to perform statistical analysis of observed data and to try to deduce the location of the mobile node in this manner.

We demonstrated the power of our approach by analysing the MIPv4 protocol. We have shown that the protocol does not protect the location information of the mobile node properly, hence an attacker can easily learn where the mobile is located. We have assumed that the attacker does not search from the foreign networks, since there are plenty of them, and the attacker does not have the power to listen to all foreign networks. Based on this assumption, we have presented that if there is no encryption between the Home Agent and the other endpoint of the tunnel, the reverse tunnelling provides no protection against location discovery. Furthermore, triangle routing only provides location privacy, if in addition to the above encrypted communication, the data traffic between CN and the mobile is tunnelled, via either a border router or the FA, using encryption. The proposed configuration is verified correctly in our model checker. To the best of our knowledge no similar study has ever been performed before in the research community.

References

[1] GOLDSMITH, M. – LOWE, G. – ROSCOE, B. – RYAN, P. – SCHNEIDER, S., *Modelling and Analysis of Security Protocols*, Addison-Wesley, 1998.

[2] ROSCOE, A. W., *The Theory and Practice of Concurrency*, Prentice Hall, 1997.

[3] HOARE, C. A. R., *Communicating Sequential Processes*, 1997.

[4] PERKINS, C., IP Mobility Support for IPv4 RFC3344, RFC3344, 2002.

[5] JOHNSON – PERKINS – ARKKO, Mobility Support in IPv6, Internet Draft, Work in Progress, 2003.

[6] SOLIMAN – CASTELLUCCIA – EL-MALKI – BELLIER, Hierarchical Mobile IPv6 Mobility Management (HMIPv6), Internet Draft, Work in Progress, 2003.

[7] SÁNDORFI, T. – LÁNCOS, J. – KUBINSZKY, F. – ZÖMBIK, L., Location Hiding Attacks in IP Networks, Communication 2003.

[8] KENT, S. – ATKINSON, R., Security Architecture for the Internet Protocol RFC2401, RFC2401, 1998.

[9] BURROWS, M. – ABADI, M. – NEEDHAM, R. M., A Logic of Authentication, 1989.

[10] MEADOWS, C., The NRL Protocol Analyser: An Overview, *Journal of Logic Programming*, 1996.

[11] FÁBREGA, F. J. T. – HERCZOG, J. C. – GUTTMAN, D. J., Honest Ideals on Strand Spaces, In: *11th IEEE Computer Security Foundations Workshop*, 1998.

[12] ABADI, M. – GORDON, A. D., A Calculus for Cryptographic Protocols: The Spi Calculus, *Information and Computation*, 1999.

[13] SCHNEIDER, S., Formal Analysis of a Non-Repudiation Protocol, *Proceedings of the 11th Computer Security Foundations Workshop*, 1998.