LA SICUREZZA IN RETE

INTRODUZIONE

Negli ultimi anni Internet ha drasticamente cambiato il modo di acquisire, sfruttare ed inviare le informazioni. Specialmente ha avuto un notevole sviluppo nell'ambito del commercio elettronico (e-commercio). E' cambiato il modo di fare affari e di gestire il lavoro, in quanto Internet permette di abbattere le barriere geografiche ed inoltre aumenta notevolmente la velocità di trasmissione di dati e delle normali transazioni finanziare che vengono fatte nel mondo degli affari. Affinché siano incluse le transazioni monetarie, ed inoltre per mantenere la riservatezza e la segretezza di particolari informazioni, è fondamentale la sicurezza.
Un nuovo strumento software per lo scambio d'informazione sono gli AGENTI MOBILI definiti come sofisticate entità software che posseggono un'intelligenza artificiale che autonomamente viaggiano attraverso un ambiente di rete e prendono complesse decisioni per conto degli utenti.
L'impiego degli agenti mobili permette di diminuire notevolmente il consumo della banda di rete, evitando inoltre gli eccessivi traffici di informazione che possono intasare la rete.
L'utilizzo di questi agenti mobili ha però introdotto notevoli problemi di sicurezza che andremo ad esaminare di seguito.

AGENTI MOBILI

Un agente mobile può essere definito come un programma, tipicamente scritto in un linguaggio particolare, che può essere inviato da un computer client e trasportato verso un server remoto che ne permette l'esecuzione.
Gli agenti mobili sono da considerarsi un nuovo metodo importante per la realizzazione di transazioni e per ricerche di informazioni in rete.
Hanno rilevanti caratteristiche comportamentali che ne determinano la loro validità. Innanzi tutto posseggono la capacità di essere molto flessibili e molto efficienti, inoltre minimizzano l'uso della risorsa massimizzando la qualità della raccolta di informazioni. Un agente è inoltre in grado di sospendere la propria esecuzione, a favore di un altro processo, trasportandosi verso un'altro host remoto, sfruttando le procedure di questo host (se l'host concede il suo permesso) e riprendere l'esecuzione dal punto nel quale era stato sospeso. Diversi agenti possono inoltre essere usati per completare sottoproblemi indipendenti, i quali possono poi essere raggruppati per raggiungere un'unica soluzione globale. Possono inoltre risolvere sottoproblemi indipendenti cooperando tra loro.

Il concetto di agente mobile è descritto dalla figura sottostante:

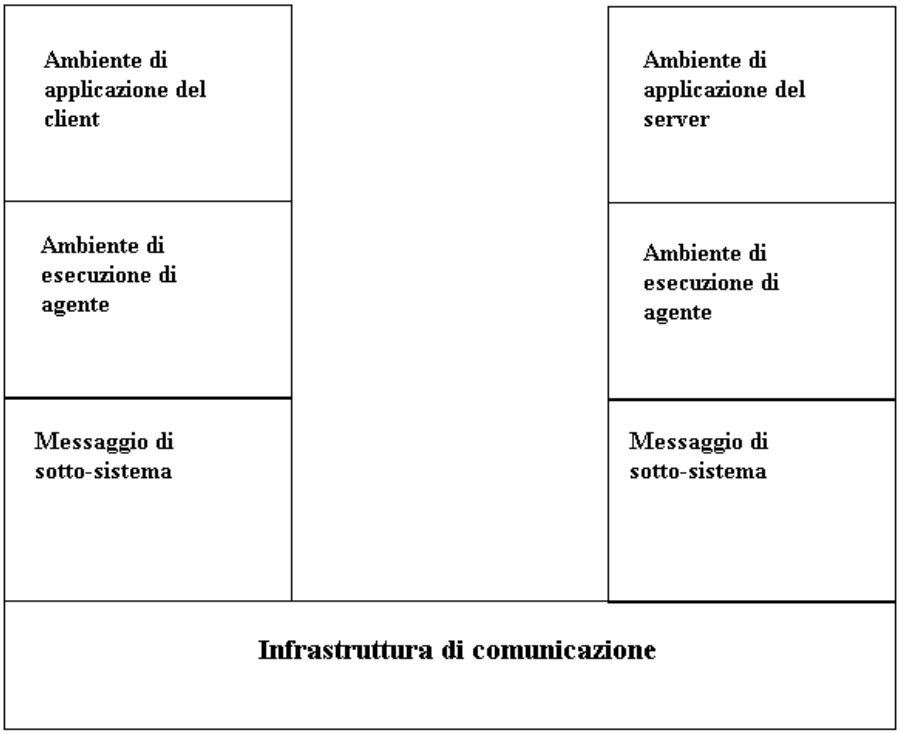

Fig.1

Un computer client è costituito da un ambiente di applicazione, come ad esempio può essere Microsoft Windows, contenete una o più applicazioni per l'interazione con il server. Queste applicazioni possono includere la ricerca e la raccolta di informazioni, transazioni, o e-mail e sono limitate per un certo ambiente di esecuzione.

L'ambiente di esecuzione di agente richiede di unire insieme diverse funzioni come ad esempio la gestione della memoria, del file system del timer e molte altre. In particolare l'ambiente di esecuzione di agente ha bisogno di legare al servizio di trasporto dei messaggi un ordine per la spedizione e la ricezione attraverso l'infrastruttura di comunicazione.

Quando un'applicazione deve spedire un e-mail o eseguire una transazione assemblerà le informazioni richieste e le passerà attraverso l'API (Application Program Interface) nell'ambiente di esecuzione di agente, nel quale verranno eseguite le istanze. L'ambiente di esecuzione di agente ha accesso ha molti e differenti programmi di agente che forniscono diversi servizi alle applicazioni dei clienti.

Vantaggi derivanti dall'utilizzo di agenti mobili:

⇒ Un agente è in grado di fornire un supporto migliore per clienti mobili, quali ad esempio computer portatili, che accedono alla rete in maniera intermittente, quando si connettono hanno bande passanti relativamente basse ed inoltre hanno limitata memoria e capacità.
⇒ Gli agenti permettono di facilitare la raccolta delle informazioni semantiche, cioè il valore di verità delle asserzioni fatte rispetto ogni mondo possibile definito anche modello.
⇒ Favoriscono inoltre le iterazioni real- time col server.
⇒ Il linguaggio scritto che li rappresenta fornisce un supporto migliore per ambienti eterogenei, in quanto ogni agente è espresso secondo un linguaggio interpretabile da opportune entità.
⇒ Un agente basato su query/transazioni può risultare molto robusto
⇒ Un agente basato su transazioni & query può essere espresso in maniera molto flessibile
⇒ Un agente basato su transazioni evita la necessità di preservare gli stati di processo
⇒ Gli agenti permettono l'utilizzo del commercio elettronico (e-Commercio)
⇒ L'uso di agenti permette di personalizzare lo user di ogni utente
⇒ Gli agenti mobili garantiscono luna manipolazione intelligente di mail.

IL PROBLEMA DELLA SICUREZZA

Ogni host di rete vuole essere sicuro che gli agenti mobili che giungono sono originati da sorgenti reali e che quindi non danneggeranno il sistema quando verranno eseguiti.
Gli agenti che interagiscono sono generati da differenti siti a volte sconosciuti. A volte può capitare che agenti contengano semplicemente dei virus che quindi potrebbero danneggiare l'intero sistema. Ecco perché gli agenti e i loro accessi ad informazioni e applicazioni devono essere controllati.
E' quindi importante costruire sistemi in grado di garantire la sicurezza delle operazioni.
Le operazioni illecite che devono essere impedite sono:

♦ Corruzione di agenti durante la loro memorizzazione. Per avere un "magazzino" sicuro che contiene gli agenti bisogna assicurarsi che gli agenti non vengano corrotti nel momento in cui sono memorizzati.

- Corruzione durante il trasferimento di agenti. Il trasferimento di un agente da un host all'altro deve avvenire in modo sicuro ed inoltre la sorgente deve essere verificata per assicurarsi che l'agente non sia stato corrotto durante il trasferimento. Questo aspetto è molto importante per garantire l'integrità e la privacy dell'agente e delle informazioni che possono esservi contenute.

- Intercettazione delle comunicazioni e corruzione. In un sistema di agenti distribuiti, la comunicazione tra agenti localizzati in differenti nodi deve essere sicura per assicurare che:
  - → Mittente e ricevente abbiano il permesso di comunicare
  - → Il trasferimento dei messaggi non deve essere facilmente decriptato da una terza parte.
  - → Il trasferimento dei messaggi tra le parti non deve subire modifiche

- Corruzione o guasto all'host sorgente. L'accesso degli agenti alle sorgenti degli host deve essere limitato. Ci si preoccupa quindi di dare autorizzazione di accesso solo ad agenti che potenzialmente non sono dannosi o non sono stati modificati dalla sorgente di host.

- Corruzione o guasto alle risorse di rete. A seconda dei comandi dati ad un agente esso può accedere solo a determinate risorse di rete.

Riassumendo quindi i tre problemi chiave per garantire la sicurezza sono:

- **PROTEZIONE DEGLI AGENTI**
- **PROTEZIONE DEGLI HOST**
- **PROTEZIONE DELLE COMUNICAZIONI**

Per la risoluzione di questo problema sono state sviluppate numerose architetture ciascuna delle quali affronta il problema in modo diverso. Di seguito ne esamineremo due completamente diverse.

L'ARCHITETTURA SAFER

SAFER sta per Secure Agent Fabrication Evolution and Roaming for E-commerce, fornisce una struttura complessa integrata per la gestione e la sicurezza degli agenti mobili impiegati in sistemi di e-commercio.
Di seguito analizzeremo come è costruito e come si comporta l'ambiente SAFER.

STRUTTURA DI UNA COMUNITA' SAFER

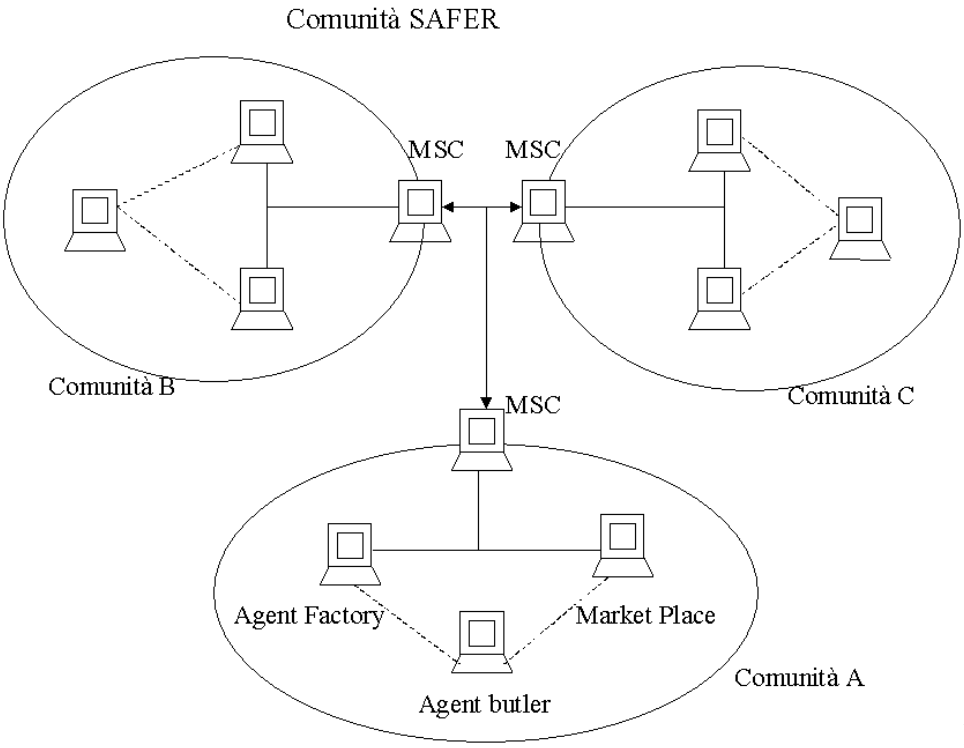

Fig.2

Ogni comunità SAFER è costituita da comunità di entità.
L'Agent factory è il responsabile della "fabbricazione" degli agenti mobili, i quali devono essere firmati in modo digitale dall'Agent factory per le successive verifiche di autenticità.
L'agent butler è l'entità responsabile della spedizione degli agenti mobili, e risponde alle richieste fatte dagli agenti.
Dietro a queste entità pubbliche della comunità SAFER ci sono molti host che appartengono alla comunità all'interno della quale vengono effettuate le transazioni di lavoro e l'e-commercio come ad esempio il Market Place mostrato in figura.
Il portiere di ogni comunità è MSC, un'entità che gestisce e controlla gli accessi e le uscite alla comunità.

PASSAPORTO, VISTO E CONTROLLO DELLE MIGRAZIONI

L'uso di certificazioni, quali passaporti e visti, permette all'MSC di gestire e controllare gli accessi alla comunità.
Passaporto e visto sono credenziali digitali che hanno lo scopo di ottenere i seguenti obbiettivi:

- Le comunità devono essere in grado di controllare se certi agenti sono stati autorizzati ad entrare o ad uscire.
- Ogni host deve essere in grado di eseguire una valutazione sull'agente mobile prima di permettergli di iniziare la sua esecuzione.
- Ogni agente deve essere provvisto delle adeguate credenziale affinché possano essere eseguite delle verifiche sulla sua identità, ed inoltre per garantirgli una certa protezione.

MSC

E' l'entità che ha lo scopo di controllare le migrazioni di agenti. Diviso in due parti, quali servizi di immigrazione e servizi di migrazione, ha come primaria attività la distribuzione e la stampa delle credenzialità.

Esegue due tipi di controlli:
CONTROLLO DI USCITE => gestisce l'emissione e la stampa dei passaporti, per agenti uscenti dalla comunità.
CONTROLLO DELLE ENTRATE => gestisce l'emissione e la stampa dei visti per agenti provenienti da comunità remote.

Funzioni	**MSC**	
	Immigrazione	**Emigrazione**
Controllo di uscita		**Emissione e stampa del passaporto**
Controllo di ingresso	**Emissione e stampa del visto**	

Tab.1

Il passaporto è un certificato che prova l'identità rilasciata da un MSC locale ed è l'unico documento ufficiale riconosciuto dalle altre comunità SAFER. Ogni agente

mobile per poter viaggiare da una comunità all'altra deve essere provvisto di passaporto. L'MSC, dopo avere eseguito opportuni controlli sull'autenticità dell'agente mobile, gli rilascia il passaporto e lo firma in maniera digitale.
In questo modo è possibile prevenire attacchi da parte di agenti non autorizzati.

Dopo avere ricevuto il passaporto un agente mobile è autorizzato a lasciare la comunità SAFER da cui proviene.Per entrare in una nuova comunità deve però richiedere il visto dall'MSC locale. Per il rilascio del visto l'MSC locale richiede una copia del passaporto dell'agente per verificarne l'integrità e l'autenticità. Una volta eseguiti e superati tutti i controlli necessari, l'MSC locale rilascia il visto di ingresso all'agente mobile richiedente.

Una volta giunto a destinazione l'agente sarà soggetto a controlli da parte degli host riceventi. L'host verificherà l'autenticità e l'integrità del passaporto, del visto e dell'agente stesso.

Ogni MSC al suo interno contiene delle liste nere che registrano i comportamenti scorretti degli agenti mobili che hanno talvolta soggiornato o hanno girato nelle comunità. Ogni lista nera può contenere la lista nera della comunità, del fabbricante e di tutte quelle parti associate all'agente che ha commesso scorrettezze.
L'MSC utilizza queste liste nere per eseguire i controlli degli accessi e delle uscite.

L'ARCHITETTURA SAS

L'acronimo SAS sta ad indicare Secure Actigen System, un'architettura che si propone di garantire la sicurezza di host, di agenti e delle comunicazioni.
Actigen non è altro che un nuovo sistema di agenti che offre un linguaggio ad alto livello completo. Il sistema mobile di agenti Actigen, basato sul modello WAVE, è un innovativo e originale spazio di programmazione, linguaggio e sistema utilizzato per eseguire processi paralleli di sistemi distribuiti aperti. Esso modella il mondo virtuale e fisico come un Knowledge Network (KN), il quale risiede in una rete fisica come Internet.

WAVE

WAVE è un nuovo modello ed una nuova tecnologia utilizzata per processi paralleli distribuiti in reti aperte di computer. WAVE è adatto per determinare modelli complessi di sistemi, anche per quelli la cui organizzazione è orientata agli oggetti.
Tutti i tradizionali modelli di calcolo sono basati sull'osservazione di programmi che operano su un insieme di dati passivi immagazzinati in memoria. Alcuni sistemi basati sull'acquisizione di conoscenza però, richiedono modelli complessi di calcolo, un esempio di questi sono i sistemi distribuiti in reti di computer che necessitano di essere eseguiti in parallelo. WAVE combina un modello organizzativo con tecnologie di controllo distribuite che possono essere utilizzate per l'esecuzione di processi eseguiti in parallelo in sistemi distribuiti aperti.
Il modello WAVE è basato sulla mobilità del codice di controllo, all'interno del quale speciali programmi ricorsivi , chiamati *waves* navigano in diverse tipologie di rete mentre vengono replicati e frazionati. *Waves* differenti (provenienti dallo stesso user o da user diversi) cooperano in uno spazio distribuito, formando società dinamiche le quali possono risolvere in maniera collettiva processi per l'acquisizione di conoscenza e problemi di controllo, senza il supporto di alcuna risorsa centrale. WAVE può avere origine da un qualunque nodo di una rete e risolvere problemi computazionali, di controllo o di interferenza in arbitrarie topologie di rete usando ogni (o tutte) risorsa di rete in parallelo.
Principali caratteristiche di WAVE:

1. *Knowledge Network (KN) e linguaggio WAVE mobile:*

WAVE modella ogni mondo di interesse come una KN i cui nodi e i collegamenti possono essere associati a volumi di informazione. Ogni nodo della KN ha un'unica identità che può essere usata dal mondo esterno o da ogni altro nodo.
La creazione di una KN arbitrario avviene per mezzo del linguaggio WAVE ricorsivo che programma, insieme con i dati intermedi "navigati" in rete, la KN.
Un nodo, o un gruppo di nodi, acquisiscono dati da processi locali ottenuti muovendo e frazionando le stringhe del programma WAVE utilizzato.

2. *Controllo distribuito, correttezza e dinamicità:*

Il sistema può evolvere nello spazio in un modello parallelo più sviluppato senza nessun controllo centralizzato. Un sistema WAVE può essere efficientemente utilizzato in nodi multi-utente, e i differenti programmi degli utilizzatori (user) possono essere sviluppati in maniera indipendente o cooperante nello stesso spazio KN, creando e/o modificando la rete KN stessa.

3. *Comunicazione con altri sistemi:*

WAVE ha un modello generale predefinito, ma contiene parti particolari per la gestione delle iterazioni di un sistema con altri. WAVE è implementato con il linguaggio C in UNIX. L'unità base che lo caratterizza e l'interprete del linguaggio, in sigla WI, che è installato su ogni host del sistema. Ogni WI è in grado di scambiare messaggi in maniera arbitraria con tutti gli altri WI.

L'intero sistema è attivato attraverso l'iniezione di programmi WAVE da parte dei nodi localizzati in ognuno degli host partecipanti. Incomincia così il funzionamento in parallelo, le stringhe del programma WAVE sono diffuse, insieme con alcuni dati locali, attraverso la KN dando vita al trasferimento fisico dei dati tra i vari WI che avviene quando attraversano e oltrepassano i confini delle partizioni di rete.
Molti programmi WAVE possono propagarsi nella rete in parallelo condividendo i dati di rete che diventano una comune e distribuita risorsa. Quando differenti programma WAVE si sovrappongono nei nodi dello stesso WI siamo in presenza di processi concorrenti.
Il modello dinamico può essere costruito con linguaggi convenzionali oppure orientati agli oggetti, quali C o C++, che guidano una simulazione ibrida (contenete il programma WAVE e i dati) nella quale la parte di programmazione WAVE viene utilizzata per controllare l'istantaneità, la distribuzione e la sincronizzazione degli oggetti.
Un programma WAVE è in grado di interrompere la sua esecuzione a favore di un altro processo invocato quando è necessario dal WI. Il programma WAVE viene rimesso in esecuzione una volta che il nuovo processo è stato completato.
Programmi WAVE e dati sono entità intercambiabili.
I nodi e i collegamenti della KN possono contenere dei programmi i quali possono essere attivati dinamicamente nello spazio della rete durante la propagazione Addizionalmente il contenuto di nodo e di collegamenti può essere usato per costruire nuovi programmi, i quali saranno attivati all'istanza successiva o in differenti punti dello spazio.
Così il parallelismo della propagazione dei programmi WAVE, l'uniforme e intercambiabile rappresentazione di programmi e dati e la dinamicità con cui vengono di modellate e cambiate le infrastrutture dei dati portano ad una conseguente flessibilità dei sistemi operanti.
Il paradigma WAVE fornisce quindi il pieno supporto per la mobilità degli agenti e può anche essere visto come l'intelligenza mobile.

STRUTTURA ACTIGEN

Il sistema di agenti mobili Actigen è supportato dal modello WAVE.
Vediamo ora come quali sono le strutture caratteristiche dell'architettura SAS:

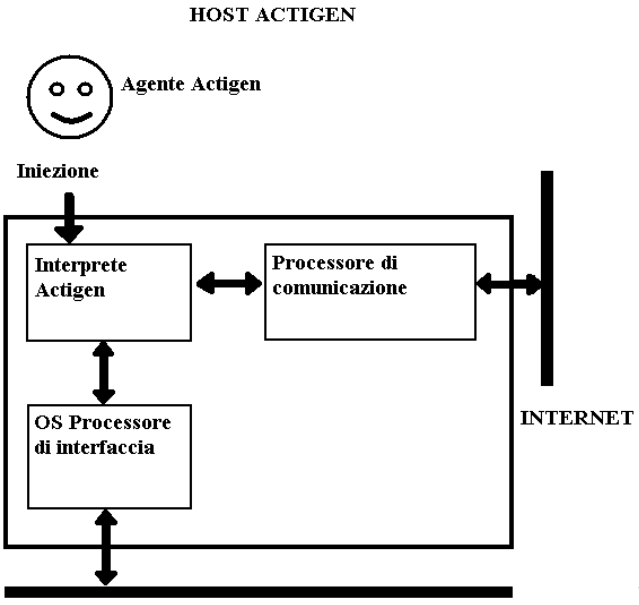

Fig.3

Un programma Actigen, chiamato agente Actigen (o semplicemente agente), migra autonomamente sotto il proprio controllo, da un nodo KN ad un latro, ed interagisce con l'ambiente locale per eseguire il lavoro assegnatogli dallo user che lo ha mandato. Il nodo fisico, all'interno del quale possono essere mappati uno o più nodi dell'Actigen KN, è chiamato Actigen host (o semplicemente host). Un host Actigen contiene tre componenti fondamentali (vedi Fig.3):

1) Un host Actigen interprete il quale esegue l'agente Actigen e gestisce il nodo locale KN.
2) Un processore Operating System (OS) di interfaccia, il quale abilita l'interprete Actigen per accedere e per aggiornare le risorse locali.
3) Un processore di comunicazione che manipola la rete di comunicazione per permettere all'agente Actigen di migrare da un host ad un'altro.

Uno schema relativo al SAS viene mostrato in Fig.4

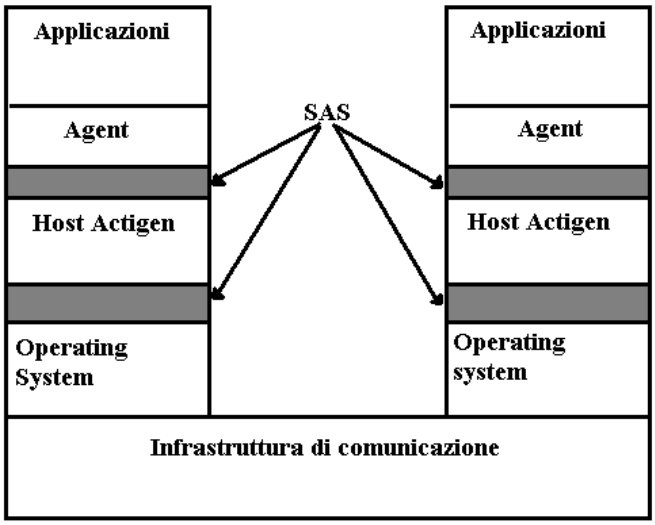

Fig.4

In una rete come questa un agente però potrebbe prendere l'intero controllo di un host ed accedere a tutte le sue risorse, oppure cambiare la topologia del nodo KN, o eseguire altri tipi di comandi non autorizzati, è quindi necessario incorporare all'interno dell'ambiente dei meccanismi di sicurezza.

SICUREZZA

Vediamo allora come vengono trattati in questa architettura i tre aspetti fondamentali della sicurezza.

➔ **Protezione della comunicazione:**

La comunicazione tra interpreti Actigen comporta la migrazione di agenti e di alcune informazioni del sistema Actigen. Proteggere la comunicazione significa fornire un canale sicuro tra gli host Actigen. Per questo scopo viene usato il protocollo SSL (Secure Socket Layer).
SSL fornisce una comunicazione criptata in modo tale da prevenire gli attacchi. Inoltre SSL provvede all'autenticazione di entrambe le parti coinvolte nella comunicazione.
Nel sistema Actigen il protocollo SSL è posizionato nel processore di comunicazione che è interposto tra la rete Internet e l'interprete, in modo che né il processore di interfaccia né l'interprete sono coinvolti.

➔ **Protezione degli agenti**:

Il sistema dovrebbe essere in grado proteggere l'integrità del codice e dei dati trasportati da un agente da host malintenzionati. Purtroppo però molti aspetti relativi a questo problema non sono ancora stati risolti.
Un agente è una composizione azioni sequenziali e parallele. Queste azioni sono eseguite dall'interprete Actigen durante la migrazione tra nodi di una KN. L'agente è

codificato come una stringa, nella quale sono esplicitate le azioni, accoppiata con un vettore, che contiene i dati. Un'azione codificata quindi non può essere modificata, ma può essere cancellata. Bisognerebbe quindi prevedere una protezione per gli agenti anche da eventuali cancellazioni che possono avvenire lungo il percorso. Non esiste però una reale protezione da questo problema, esiste solo un algoritmo che permette di verificare se durante la migrazione di un agente sono avvenute della cancellazioni. Questo algoritmo si basa sul fatto che un agente appartenente ad un determinato dominio, prima di poter migrare verso un altro dominio deve richiedere il passaporto all'AC locale, il quale, oltre a rilasciargli il documento esegue una copia dell'agente. L'agente migra quindi verso il nuovo dominio. Una volta concluse le operazioni che doveva eseguire tenta di ritornare al dominio di origine passando dall'AC locale che quindi lo confronta con la copia che ha memorizzato. Se l'AC verifica che l'agente e la sua copia sono ancora uguali allora non si sono verificate cancellazioni, altrimenti l'agente risulta modificato.

Dei tre aspetti quelli che assume un'importanza rilevante è la protezione degli host, in quanto si vuole fare in modo che ogni utente che si connette alla rete sia protetto da eventuali virus o attacchi esterni.

➔ PROTEZIONE DEGLI HOST

Risolvere il problema della protezione degli host comporta l'utilizzo di due processi fondamentali: Autenticazione e Autorizzazione. Questo comporta che a tutte le entità concorrenti nel SAS sia data un'identificazione.
Vediamo più in dettaglio come viene fatto tutto questo.
Diamo innanzitutto la definizione di PRINCIPALE quale entità la cui identità può essere autenticata. L'ambiente SAS è quindi composto da principali. Ogni principale esegue una specifica regola, ha le sue responsabilità e i suoi interessi. Ad ogni principale è anche associato un certificato che ne descrive le priorità, le regole e la chiave pubblica per poterlo decriptare. Dentro al sistema l'identità del principale è riconosciuta dalla sua firma e la politica di sicurezza che lo gestisce è determinata in base alla sua priorità.
I principali nel SAS sono i seguenti:

- Iniettore (user):

  E' il padrone dell'agente, colui che immette l'agente in rete.

- Agente Actigen (XA) :

  Un agente Actigen appartiene all'iniettore che lo ha immesso nella rete.

- Agente di autenticazione (AA):

  E' un agente stazionario che risiede accanto all'interprete Actigen.

- Centro di autenticazione (AC):

  Non è associato a nessun interprete e ha lo scopo di gestire il certificato dell'AA

INFRASTRUTTURA PER LA CERIFICAZIONE ACTIGEN (ACI)

Dare un'identificazione ad ogni principale dell'architettura SAS non è semplice. Certificazione assieme alla criptazione e alla firma digitale forniscono un ottimo supporto per la sicurezza e la privacy degli host.
L'ambiente SAS è organizzato in domini (Fig.5) ognuno dei quali comprende un AC e una serie di AA. Tutte le macchine dentro ad un dominio sono sottoposte al controllo di un'autorità chiamata Autorità di Certificazione (CA), così che la consistenza e la fedeltà degli agenti possono essere controllate e gestite attraverso una struttura di certificazione gerarchica.

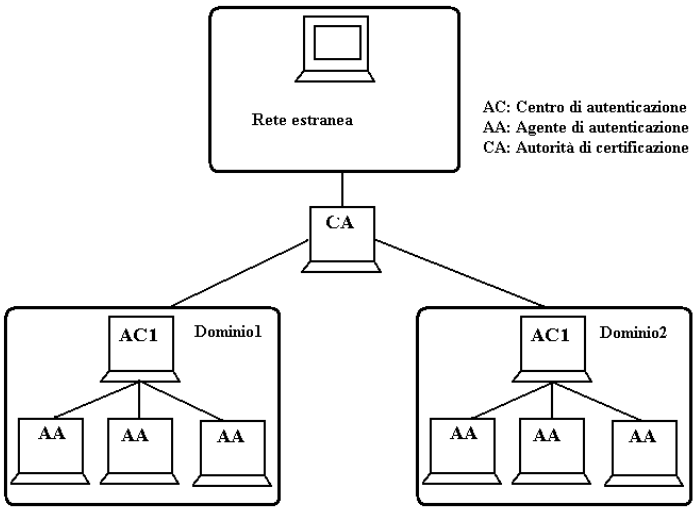

Fig.5

Servizio di certificazione Intra-dominio:
Dentro ad un dominio SAS sono utilizzati due ordini gerarchici di certificazione: AA determina lo user e AC determina l'AA. L'AA assegna ad uno user locale un certificato. Anche all'AA è associato un certificato che gli viene fornito dall'AC. Quindi l'identità di uno user è determinata da due certificati: il certificato dell'iniettore e il certificato associato all'AA. Un AC distribuirà a tutti i suoi AA i relativi certificati, in modo che ogni AA conosce il certificato degli altri. Il certificato dello user non deve essere distribuito in maniera così esplicita, sarà l'agente Actigen a trasportarlo durante la sua migrazione.
Un certificato sarà revocato quando il suo contenuto risulta modificato.
I certificati hanno valore solo dentro ad un dominio SAS.

Inter-dominio, Passaporti e Visti:
Per lo spostamento degli agenti tra i vari domini occorrono certificazioni particolari quali sono i passaporti e i visti. Il passaporto è rilasciato ad un agente dall'AC locale e permette la migrazione dell'agente al di fuori del dominio. Il visto è invece rilasciato dall'AC del dominio di destinazione e permette quindi l'ingresso in un nuovo dominio. Sia il passaporto che il visto hanno vita molto breve, non esistono particolari motivi per cui debbano essere revocati.

STRUTTURA DELL'AGENTE DI AUTENTICAZIONE

Nel progettare il SAS il goal principale che deve essere realizzato quindi è la sicurezza. Per il raggiungimento di questo scopo un ruolo importante è ricoperto dall'Agente di Autenticazione (AA) che ha le seguenti funzioni:

- gestione dell'account dell'utente
- gestione della certificazione dell'utente
- spiegamento di una politica di sicurezza
- controllo dell'iniettore dell'agente

L'Agente di autenticazione risiede vicino all'interprete Actigen. L'AA infatti agisce come un involucro che ricopre l'interprete, in modo che tutte le iterazioni tra l'interprete e l'ambiente passano attraverso l'AA che agisce secondo le norme di sicurezza.

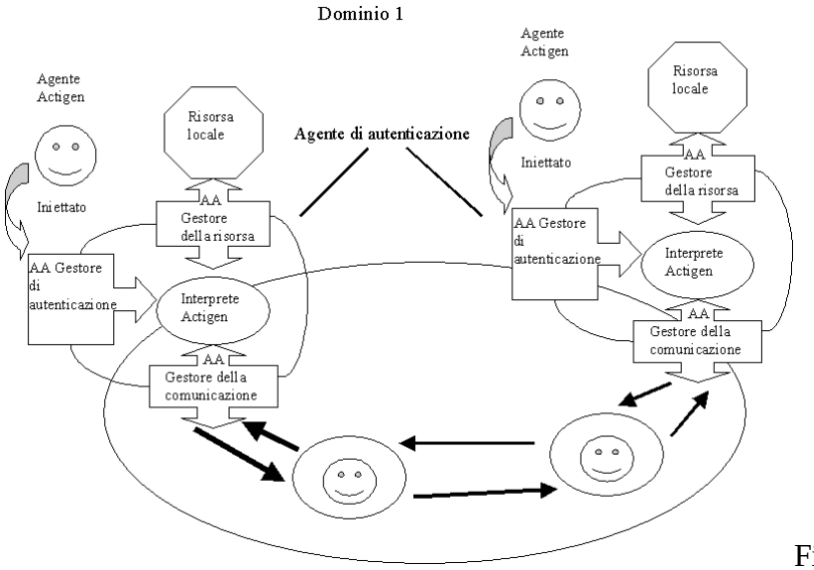

Fig.6

L'AA è composto da cinque parti fondamentali (Fig.6) che hanno diverse funzioni:

AA Gestore di interfaccia di utente: fornisce l'interfaccia per lo user per accedere e gestire l'AA, e quindi l'interprete Actigen. Lo user può iniettare un agente solo attraversi il gestore di interfaccia.

AA Gestore dell'autenticazione: controlla l'identità e i diritti associati ad uno user che inietta un agente; ha anche il compito di registrare l'agente che è stato iniettato.

AA Gestore della risorsa: esegue una supervisione degli accessi degli agenti alla risorsa locale che avviene tramite l'interprete. Il diritto di un agente è ereditato dal suo iniettore e l'identità dell'iniettore dell'agente viene controllata solo se richiesto dalla politica di sicurezza.

AA Gestore della comunicazione: Aggiunge all'agente un'informazione relativa alla sua identificazione (programma Actigen) quando l'agente esce dall'interprete, poi rimuove e controlla l'informazione di identificazione aggiunta quando l'agente arriva all'host Actigen da un'altro host. La crittografia per garantire una comunicazione sicura è applicata solo quando è ritenuta necessaria.

AA Gestore della politica di sicurezza: mantiene in memoria una tabella relativa ai permessi di accesso, la quale mappa i diritti relativi alle risorse locali. Il gestore della politica di sicurezza, insieme col gestore di risorsa permette di eseguire un controllo di accesso fatto su misura delle risorse ed inoltre ne garantisce la rispettabilità.

CONCLUSIONI

Il problema della sicurezza in rete al giorno d'oggi assume quindi un ruolo rilevante. E' quindi necessario prevedere delle infrastrutture che permettano di garantire e salvaguardare i dati che vengono immessi nella rete.
Esistono molti sistemi diversi, ognuno dei quali affronta il problema in maniera diversa.
L'ambiente SAFER si basa principalmente sull'uso di passaporti e visti che vengono controllati da un'entità posta all'ingresso della comunità. Questa soluzione risulta molto flessibile ed efficace.
L'architettura Actigen, oltre ad utilizzare le certificazioni, come nell'ambiente SAFER, si occupa in maniera distinta dei tre problemi fondamentali per la sicurezza. Inoltre prevede l'utilizzo di un agente per la protezione dell'host che controlla gli accessi da parte di agenti inviati da altri user.
Il progetto del SAS è eseguito in maniera molto generale e soprattutto è tenuto separato dall'interprete Actigen in modo tale che questa architettura possa essere adattata a diversi sistemi di agenti mobili.

INDICE

BIBLIOGRAFIA

1. Son T. Vuong, Peng Fu – A Secure Architecture and Design for Mobile Intelligent Agent System – Department of computer science the University of British Columbia, Vancouver, B.C. V6T 1Z4, Canada
2. Sheng Uei Guan, Tianaham Wang and Sim Heng ong – A Secure Approach for Mobile Agent Migration Control – Department of Electrical and Computer Engineering National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, 2002
3. Son Vuong and Ivailo Ivanov – Mobile Intelligent Agent Systems: WAVE vs. JAVA - Department of computer science the University of British Columbia, Vancouver, B.C. V6T 1Z4, Canada 1996
4. C. Harrison, D. Chess and Kershenbaum, Reserch report – Mobile Agents: Are they a good Idea, IBM Research Division, T.J. Watson Research Center, NY 10589, March 1995
5. Larry Korba, Member IEEE – Towards Secure Agent Distribuition and Comunication – National Research Council of Canada, 1999
6. Peter S. Sapaty and Peter M. Borst – WAVE: Mobile Intelligence in Open Networks -