Scuola dottorale di Ingegneria

Sezione di Ingegneria dell'Elettronica Biomedica,

dell'Elettromagnetismo e delle Telecomunicazioni

XXIII ciclo

Architettura distribuita sicura per piattaforma di comando e controllo basata su A.I.S. (Automatic Identification System)

Ing. Antonio Vollero

A.A. 2010/2011

Docente Guida: Prof. Alessandro NERI

Coordinatore: Prof. Lucio VEGNI

INDICE

LISTA DEGLI ACRONIMI

ADC	Analog to Digital Converter
ADS	Advanced Design System
AGC	Automatic Gain Control
AH	Authentication Header
AIS	Automatic Identification System
ASM	AIS Service Management
ATON	Aids To Navigation
BAS	Basic AIS Service
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CA	Certificate Authority
CBC	Cipher Block Chaining
DMZ	Demilitarized Zone
DSC	Digital Selective Calling
ESP	Encapsulating Security Payload
ETA	Estimated Time of Arrival
GBS	Ground Base Station
GPS	Global Positioning System
GMDSS	Global Maritime Distress Safety System
GMSK	Gaussian Minimum Shift Keying
IALA	International Association of Marine Aids to Navigation and Lighthouse Authorities
IEC	International Electrotechnical Commision
IF	Intermediate Frequency
IMO	Internazional Marittime Organization
IMSO	International Maritime Satellite Organization
IPSEC	Internet Protocol Security
ITU	InterNational Telecommunications Union
LO	Local Oscillator
LNA	Low Noisy Amplifier
LOS	Line of sight
LSS	Logical Shore Station

MAC	Message Authentication Code
MDS	Minimum Detectable Signal
Mhz	Mega Hertz (10^6)
MMSI	Maritime Mobile Service Identity
NBDP	Narrow Band Direct Printing
NRZI	Non Return to Zero Inverted
OSI	Open System Interconnection
PI	Presentation Interface
PSS	Physical Shore Station
RF	Radio Frequency
SA	Security Association
SART	Search and Rescue Transmitter
SNR	Signal to Noise Ratio
SOLAS	Safety of Life At Sea
SPD	Security Policy Database
SSL	Secure Socket Layer
TLS	Transport Layer Security
VHF	Very High Frequency
VDL	VHF Data Link
VTS	Vessel Traffic Service

INTRODUZIONE.

1. Overview.

Prima in ambito aereo e poi anche in quello marittimo, al fine di prevenire i rischi di collisione con ostacoli o altri veicoli, sono stati introdotti sistemi di ausilio alla navigazione. Tali sistemi, installati a terra o a bordo, offrono supporto alla condotta dei mezzi fornendo elaborazioni basate sull'interazione con la realtà esterna, acquisendo ed elaborando dati in tempo reale.

In tale contesto, in ambito marittimo, si inserisce l'*Automatic Identification System*, sistema di ausilio alla navigazione che consente l'identificazione automatica delle unità navali dotate di apposito transponder, reso obbligatorio su particolari unità navali e con determinate dimensioni, dal Capitolo V della Convenzione di Londra del 1 Novembre 1974 sulla sicurezza della vita umana in mare (c.d. SOLAS[1]).

Si immagini una situazione in cui la visibilità sia ridotta dalla nebbia durante l'attraversamento di uno schema di separazione del traffico o si è in avvicinamento notturno alla costa in un tratto di mare in cui il traffico navale è elevato, se il radar risulta quasi inservibile a causa di un piovasco che ne riduce la capacità di discriminazione. In tali situazioni occorre poter contare su un sistema che consenta di "vedere ed essere visti".

L'AIS è in pratica un sistema che consente di acquisire informazioni sul traffico navale, supportando l'attività di monitoraggio tesa ad assicurare la sicurezza nell'ambito della navigazione marittima. Esso presuppone l'acquisizione dei dati basata sulla trasmissione continua e reciproca tra le navi (*ship to ship*) e fra le navi e le stazioni di base a terra (*ship to shore*), tramite due canali VHF dedicati. Le informazioni scambiate consentono di ricostruire la situazione dell'area interessata e possono essere classificate in tre diverse tipologie:

- di carattere statico;
- di carattere dinamico;
- relative al viaggio in corso.

[1] Safety of Life At Sea, convenzione convocata dall'*Internazional Marittime Organization* (IMO) ed entrata in vigore in Italia l'11 Settembre 1980, a seguito della ratifica avvenuta con L. 313/1980.

La prima categoria di informazioni si riferisce all'identificazione della nave, i relativi dati sono impostati nel sistema al momento dell'installazione a bordo e tendenzialmente non sono modificabili (es. nome della nave, numero M.M.S.I.[2], dimensione e tipologia della nave et c.).

Le informazioni dinamiche sono quelle aggiornate automaticamente e vengono prelevate tramite connessione con gli strumenti di bordo della nave, in tale ambito collochiamo, ad esempio, i dati relativi a posizione, velocità e rotta seguita dalla nave.

Infine le informazioni della terza categoria riguardano il pescaggio[3] della nave, l'eventuale presenza di carichi pericolosi a bordo, le destinazioni e le previsioni circa la rotta pianificata e l'arrivo previsto.

2. Obiettivi della ricerca.

Scopo di questo lavoro è esaminare il funzionamento dell'AIS, analizzando i principi su cui si basa, valutando i potenziali usi ed evoluzioni per tale sistema. L'intero studio portato avanti risulterà di supporto allo sviluppo di un sistema di monitoraggio navale asservito all'intera area mediterranea, realizzabile tramite interconnessione di diverse resti AIS nazionali dei singoli Stati costieri, ponendo particolare attenzione su come assicurare uno scambio dati con adeguati livelli di sicurezza (SSL, IpSec).

Lo metodologia adottata in tale studio si è distinta in tre fasi ben distinte.

Si è iniziato con un primo periodo dedicato all'esame della documentazione, sia generale che tecnica, relativa alla tecnologia in oggetto

Una volta acquisite le nozioni indispensabili per comprendere il funzionamento del sistema, si è avviata una seconda fase pratica. Grazie alla possibilità di frequentare il Reparto di Ricerca e Sviluppo del Corpo delle Capitanerie di Porto – Guardia Costiera, si è potuto toccare con mano e testare sul campo il reale funzionamento dell'AIS, nonché le diverse tipologie di impieghi ed le possibili nuove applicazioni realizzabili.

[2] M.M.S.I. è l'acronimo di Maritime Mobile Service Identity, in pratica è un codice a nove cifre che identifica in maniera univoca gli apparati radio installati a bordo di navi o presso stazioni costiere. Tale codice viene trasmesso automaticamente durante le comunicazioni. Le prime tre cifre identificano il paese d'appartenenza (l'Italia per esempio ha prefisso 247), mentre le ultime sei identificano in maniera univoca la stazione di bordo o di terra.

[3] Per pescaggio si intende l'altezza della parte che rimane immersa nell'acqua e che intercorre quindi tra la linea di galleggiamento ed il punto inferiore estremo della chiglia.

Infine, sulla base dell'accurata conoscenza acquisita, ci si è addentrati in un'ultima fase dedicata allo studio e al supporto nella realizzazione di una AIS di livello europeo, in cui ci si è trovati coinvolti in prima linea.

La tesi è organizzata come segue:

il primo capitolo fornisce una descrizione degli standard su cui si basa il funzionamento dei transponders che ricevono e trasmettono dati AIS, partendo dall'esame dalle tecnologie su cui si basa lo scambio dati, sino ad arrivare alla codifica utilizzata per incapsulare i messaggi.

Il secondo capitolo illustra come deve essere realizzata una rete AIS costiera che consenta, attraverso lo scambio dei dati AIS, di ricostruire la situazione del traffico navale nell'area interessata.

Nel terzo e quarto capitolo viene analizzato il problema della sicurezza informatica, illustrando le metodologie di cui si può usufruire per garantire standard di sicurezza adeguati a garantire la protezione dei dati. Vengono presentate dettagliatamente, come scelte progettuali prese in considerazione, la sicurezza a livello IP (IPSec.) e SSL/TLS.

Nel quinto ed ultimo capitolo viene esposto il risultato dell'analisi svolta, grazie alla quale si sviluppata l'architettura di una rete di monitoraggio di gerarchia superiore, che raccogliendo i dati dalle varie nazioni costiere permette di controllare il movimento delle unità navali in tutto il bacino mediterraneo.

Infine vengono riportate le conclusioni relative alla reale creazione di tale sistema, al quale il sottoscritto ha fornito il suo contributo nelle vesti di Ufficiale della Marina Militare impiegato presso il Reparto Ricerca e Sviluppo del Corpo delle Capitanerie di Porto – Guardia Costiera.

CAPITOLO 1

In questo capitolo analizzeremo la tecnologia su cui si basa il funzionamento di un transponder AIS, le parti di cui risulta composto e le norme che ne regolano il funzionamento.

Si analizzeranno quindi i canali radio su cui opera, la codifica utilizzata in fase di trasmissione e l'incapsulamento dei messaggi contenti le informazioni di interesse.

La maggior parte delle specifiche di progetto per un transponder AIS sono stabilite da organismi internazionali (IALA[4], ITU[5], IEC[6]), in questo modo si ha una standardizzazione che garantisce piena compatibilità e interazione tra sistemi realizzati da produttori diversi.

1.1 Transponder AIS.

Un transponder AIS opera trasmettendo informazioni in banda VHF marittima. Rispetto ad un radar presenta il vantaggio di trasmettere su una frequenza meno sensibile alle interferenze di pioggia o mare e, inoltre, consente di ottenere dei dati ulteriori rispetto al semplice eco di ritorno (che consentono di ricavare solo posizione, rotta e velocità del target "battuto").

La comunicazione continua e reciproca tra transponder AIS avviene tramite due canali in banda VHF marittima, riservati per tale utilizzo dall'ITU:

- ***AIS 1 centrato su 161.975 MHz***
- ***AIS 2 centrato su 162.025 MHz***

I dati trasmessi vengono codificati tramite una modulazione GMSK.

[4] IALA, *International Association of Marine Aids to Navigation and Lighthouse Authorities.*

[5] ITU, *International Telecommunications Union*, che fornisce indicazioni inerenti i messaggi radiotrasmessi e ha stabilito le frequenze dedicate ai segnali AIS.

[6] IEC, *International Electrotechnical Commision*, che ha stabilito le specifiche di incapsulamento del messaggio AIS prima di essere trasmesso.

L’AIS ha l’obiettivo di fornire a chi naviga una migliore conoscenza della situazione circostante, rivelandosi come uno strumento per ridurre il rischio di collisioni grazie al costante aggiornamento delle informazioni trasmesse e ricevute.

I dati relativi alle notizie di tipo statico sono trasmessi ogni 6 minuti o in caso di modifica dei dati oppure su richiesta, mentre le informazioni dinamiche vengono inviate con una frequenza che varia a seconda della velocità con cui si muove l’unità.

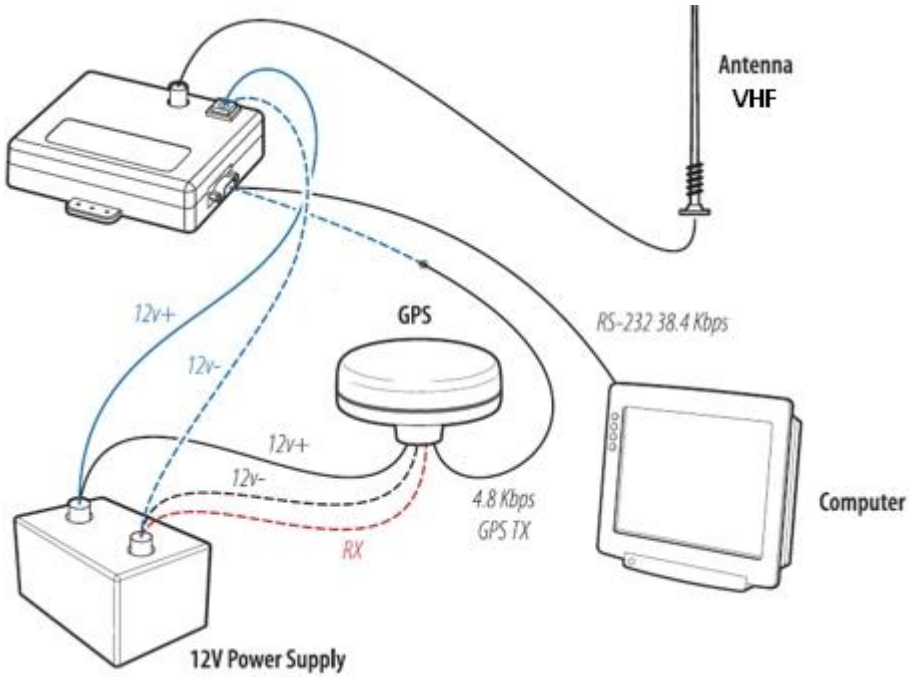


Figura 1.1: elementi costitutivi di un transponder AIS.

Un sistema AIS risulta quindi composto da un trasmettitore VHF, tre ricevitori VHF[7], un antenna e da dei link di collegamento con:

- ricevitore satellitare che consenta timing e localizzazione (ad es. GPS);
- sensori di bordo;
- display di visualizzazione per le informazioni ricevute.

[7] Due ricevitori per i dati veicolati attraverso i canali AIS1 e AIS2, il terzo riservato alla ricezione dei messaggi DSC. Il *digital selective calling* (DSC) è una importante funzionalità di sicurezza che può essere presente negli apparati di trasmissione VHF marini che opera sul canale 70. Premendo un singolo pulsante (tipicamente rosso) il sistema provvede a trasmettere l’identificativo della barca ed anche la posizione da cui si invia la richiesta di aiuto, se lo strumento è interfacciato con un GPS

Inoltre esistono due tipi distinti di transponder AIS:

- ***Classe A***, progettato per ricevere contemporaneamente su entrambi i canali AIS, è riservato all'utenza professionale e, in generale, a tutte le stazioni per le quali sussiste un obbligo di installazione ai sensi della convenzione SOLAS[8];

- ***Classe B***, versione semplificata e più economica, adoperata su pescherecci, imbarcazioni da diporto ed in generale per le unità che non sono obbligate all'installazione di quello di Classe A. Sostanzialmente un transponder di Classe B presenta alcune limitazioni:

    - una minor frequenza di trasmissione dei dati (ad es. per unità che si muovono ad una velocità di 14 nodi trasmette ogni 30 sec., contro i 10 sec di un transponder di Classe A)
    - non trasmette l'identificativo IMO dell'unità navale
    - non trasmette l'ETA (*estimated time of arrival*)
    - riceve ma non trasmette messaggi di sicurezza in formato testo
    - non fornisce informazioni prelevabili da strumenti di bordo, come la velocità di accostata o l'immersione della carena.

[8] Sono navi SOLAS le navi mercantili superiori a 300 tsl (tonnellate stazza lorda) impiegate su rotte internazionali, quelle superiori a 500 tsl impegnate in viaggi non internazionali e le navi passeggeri con più di 12 passeggeri, indipendentemente dal tipo di viaggi che effettua.

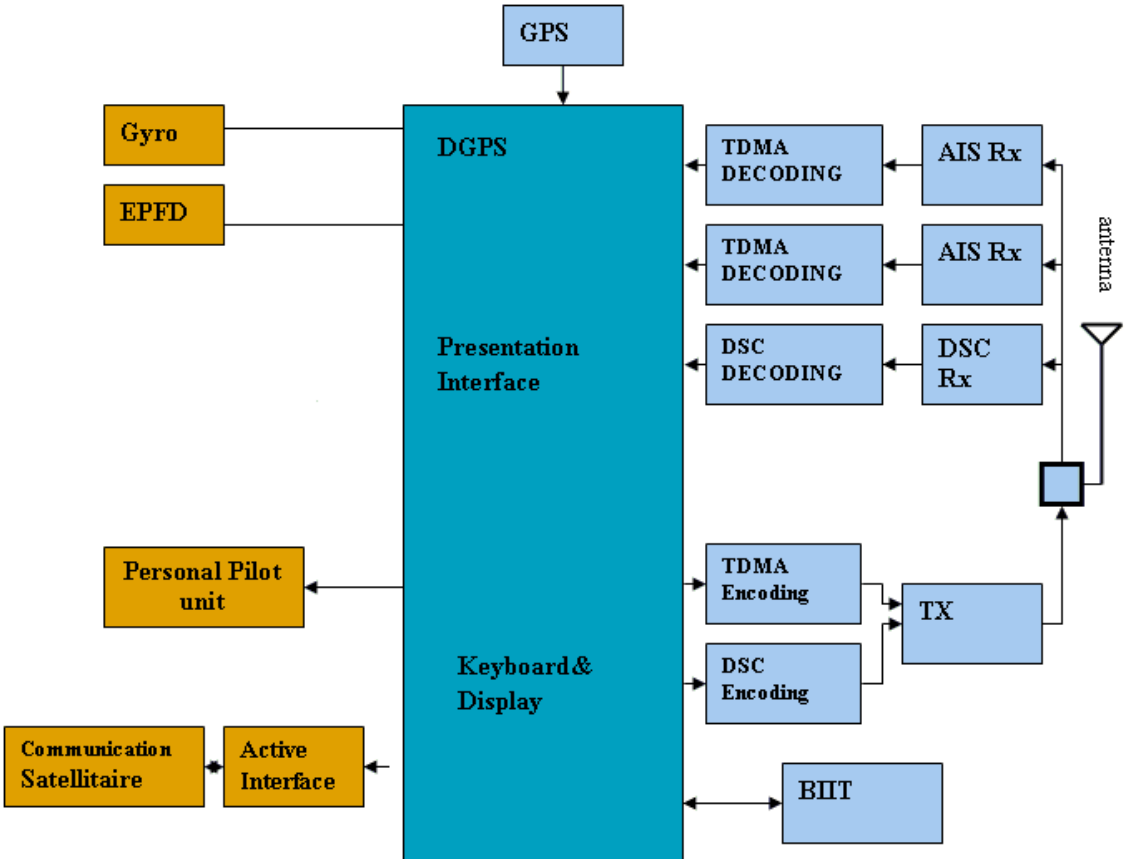


Figura 1.2: schema a blocchi di un transponder AIS di bordo.

Per quanto invece riguarda transponder installati lungo le coste, ne esistono di varie tipologie e con diverse finalità:

- ***AIS Base Station***, installate dalle autorità nazionali, responsabili del monitoraggio del traffico navale, per acquisire i dati trasmessi dalle navi. Tramite una adeguata rete di *base station* una nazione può costituire la propria rete AIS nazionale, in grado di raccogliere dati da sensori periferici ed indirizzarli verso un server centrale dove vengono fusi e ridistribuiti in maniera tale da fornire un quadro d'insieme del traffico navale in prossimità delle coste.

- ***AtoN (AIS aids to Navigation)***, che trasmettono dati relativi alle boe e ai fari utilizzando il medesimo *VHF Data Link*, in modo da poter visualizzare la posizione di tali elementi su display con cartografia elettronica o su radar in grado di interfacciarsi con un ricevitore VHF.

- ***AIS SART (Search And Rescue Transmitter)***, è un dispositivo attivato in caso di emergenza (condizione di distress), esso periodicamente trasmette sui VDL AIS1 e AIS2 la propria posizione ed un messaggio di **distress** che segnali a chi riceve tali informazioni, richiedendo di intervenire in aiuto della nave.

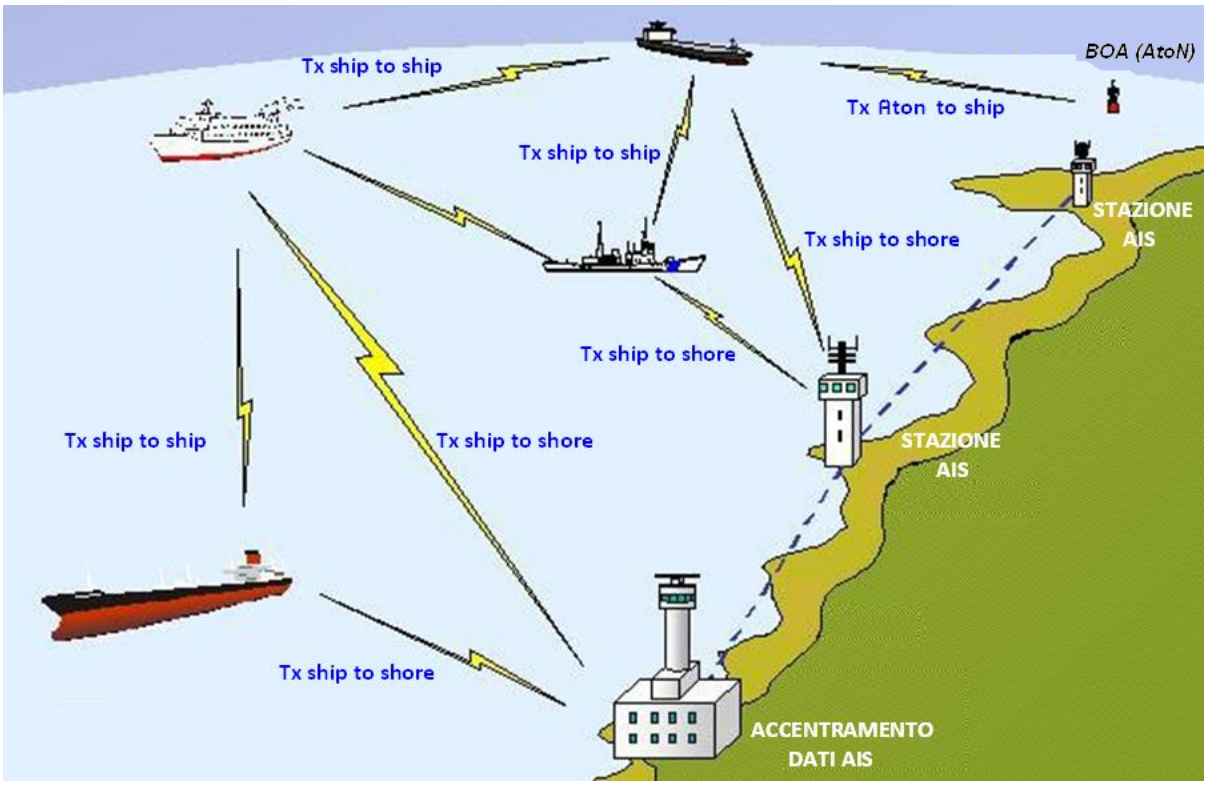


Figura 1.3: uno scenario di scambio dati AIS tra transponder di bordo, *base station* costiere ed AtoN.

1.1.1 Radiocomunicazioni.

Il sistema AIS si basa su data link VHF, non possiamo quindi omettere un richiamo ai relativi principi di radiocomunicazioni.

La variazione di campi elettrici o magnetici, originati da cariche elettriche in movimento, avviene sempre in maniera accoppiata tra di loro e l'insieme dei due viene chiamato ***campo elettromagnetico***.

La propagazione di un'onda avviene attraverso successivi concatenamenti come indicato nella figura 1.4. Ne risulta che l'onda elettrica e l'onda magnetica non sussistono mai separate ma coesistono sempre in una stretta interdipendenza in quanto le variazioni di campo dell'una provocano la nascita del campo in variazione che costituisce l'altra e

viceversa. L'insieme delle due onde fra loro legate costituisce ***l'onda elettromagnetica*** che è l'effettiva entità che si propaga nello spazio di cui l'onda elettrica e quella magnetica non sono che due aspetti particolari.

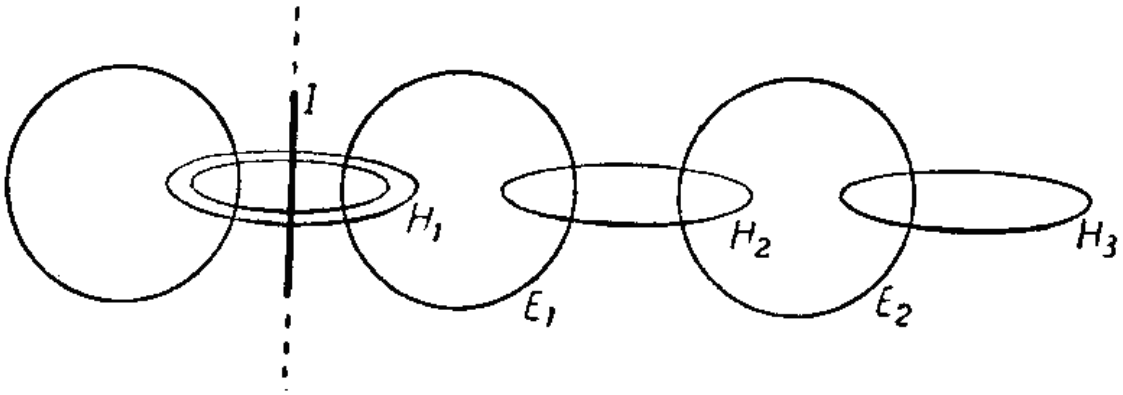


Figura 1.4: schematizzazione del concatenamento dei campi magnetico ed elettrico in variazione.

Grazie agli studi di James Clerck Maxwell prima e gli esperimenti di Heinrech Rudolph Hertz poi, oggi sappiamo che la radiazione di un'onda elettromagnetica segue i medesimi principi di quella luminosa, quindi è assimilabile ad un'onda piana ovvero è una distribuzione di campo elettromagnetico, in cui in ogni punto i vettori campo elettrico e campo magnetico sono perpendicolari fra loro e giacciono su piani perpendicolari alla direzione di propagazione r.

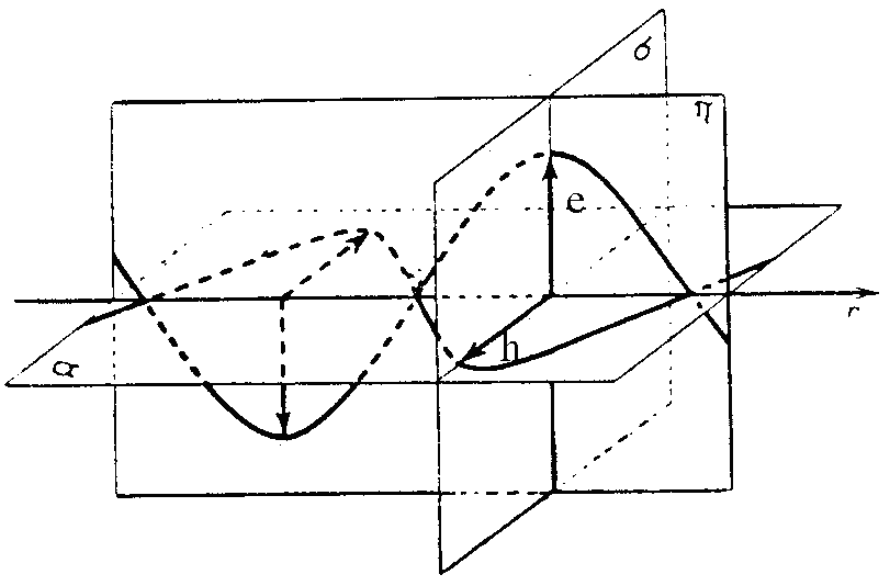


Figura 1.5: propagazione di un'onda piana lungo la direzione r.

Un’onda che si propaga risulta caratterizzata da:

- T il periodo dell’onda, cioè il tempo trascorso tra il presentarsi di due successive creste;
- • la lunghezza dell’onda cioè la distanza tra due successive creste;
- E, H i valori massimi assunti dai rispettivi campi;
- f = 1/T la frequenza dell’onda, cioè il numero di oscillazioni complete nell’unità di tempo (cicli al secondo oppure Hertz);
- $c = \frac{l}{T} = lf$ la velocità di propagazione dell’onda (c • 3 x 108 m/s).

Le onde elettromagnetiche vengono classificate secondo la frequenza o la lunghezza d'onda, la preferenza per un criterio o per l'altro è esclusivamente un fatto di consuetudine.

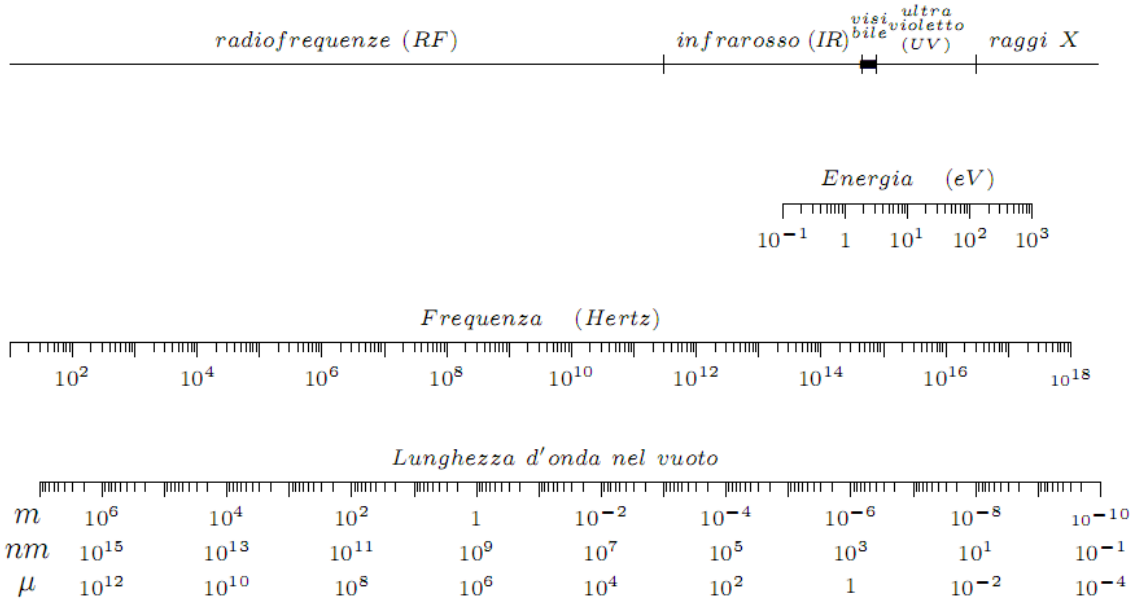


Figura 1.6: relazione frequenza-lunghezza d’onda per le radiazioni elettromagnetiche.

A frequenze molto basse (lunghezze d'onda molto alte) il campo elettrico ed il campo magnetico si comportano, in pratica, come agenti fisici indipendenti tra loro e l'energia dei campi resta localizzata attorno alla sorgente, per frequenze superiori il campo

elettromagnetico si manifesta come onde che trasportano energia (radiazione elettromagnetica).

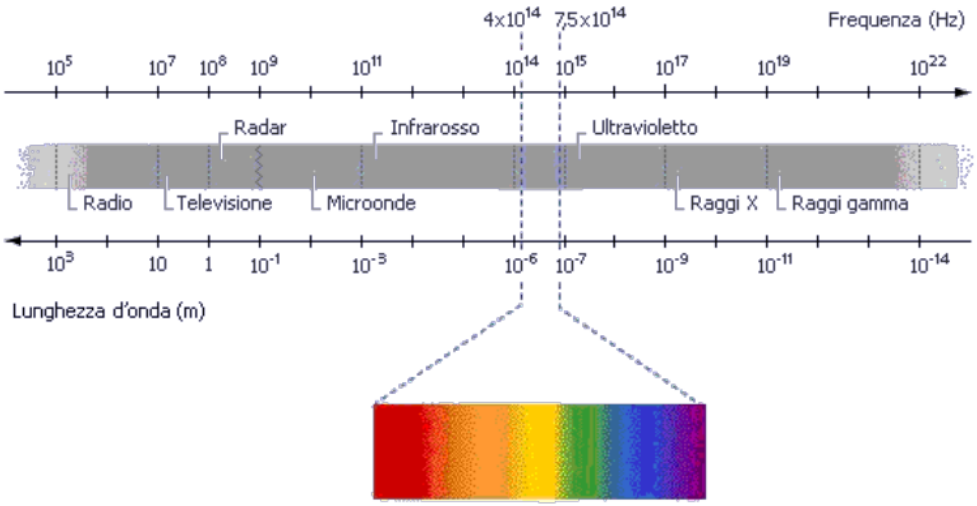


Figura 1.7: spettro di frequenze delle radiazioni elettromagnetiche

Ogni banda dello spettro viene a sua volta suddivise in bande più piccole, ognuna con la propria denominazione. Concentriamo la nostra attenzione sulle ***onde radio*** ovvero sulla suddivisione della banda a radiofrequenza, quella compresa tra i 30 Hz ed i 3000 Ghz.

Frequenza	Lunghezza d'onda	Denominazione
30 - 300 Hz	10^7 - 10^6 m	**ELF** (extremely low frequency)
300 - 3000 Hz	10^6 - 10^5 m	
3 - 30 KHz	10^5 - 10^4 m	**VLF** (very low frequency)
30 - 300 KHz	10^4 - 10^3 m	**LF** (low frequency)
300 - 3000 KHz	10^3 - 10^2 m	**MF** (medium frequency)
3 - 30 MHz	10^2 - 10 m	**HF** (high frequency)
30 - 300 MHz	10 - 1 m	**VHF** (very high frequency)
300 - 3000 MHz	1 m - 10 cm	**UHF** (ultra high frequency)
3 - 30 GHz	10 - 1 cm	**SHF** (super high frequency)
30 - 300 GHz	1 cm - 1 mm	**EHF** (extremely high frequency)
300 - 3000 GHz	1 mm - 100 μ	

. **Figura 1.8:** suddivisione della banda a radiofrequenza.

Le onde LF, MF ed HF vanno sotto il nome di radiofrequenze, mentre le VHF ed UHF sono anche note come onde ultracorte. Le SHF ed EHF sono usate nelle trasmissioni satellitari e nella realizzazione dei radar.

1.1.2 Propagazione delle onde elettromagnetiche.

Le onde e.m. irradiate da un'antenna trasmittente possono giungere all'antenna ricevente seguendo diversi percorsi. Nel caso di un'antenna prossima al suolo è conveniente distinguere tali percorsi in due grandi classi: percorsi che si svolgono nello spazio libero e percorsi che si svolgono in prossimità del suolo o sotto l'influenza di esso.

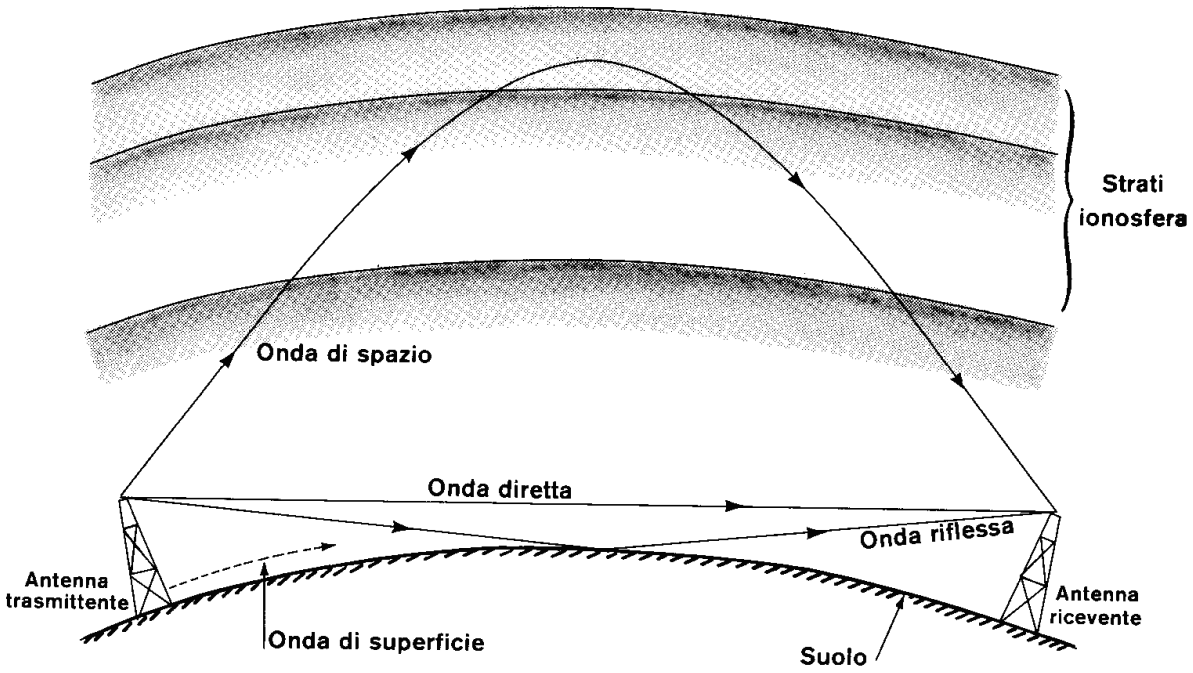


Figura 1.9: tipi di propagazione.

Le onde e.m. irradiate da un'antenna trasmittente possono giungere all'antenna ricevente seguendo diversi percorsi.

L'***onda spaziale*** provvede alle comunicazioni a grande distanza e, in campo nautico, era sfruttata nel sistema OMEGA.

L’***onda di superficie*** ha importanza soprattutto nel campo delle onde medie e lunghe (dalle ELF sino alle HF) e provvede alle comunicazioni a piccola distanza, alla radiodiffusione e, in campo nautico, nel sistema Loran e per il radiogoniometro. La propagazione avviene mediante cammino dell’onda lungo la superficie terrestre, scavalcando le colline, superando laghi, fiumi ed anche mari. La propagazione è molto condizionata dalla conducibilità del terreno, più questi è un buon conduttore più l’onda si propaga lontano

L’***onda diretta e l’onda riflessa,*** a causa della curvatura terrestre, hanno importanza solo quando l’antenna trasmittente o la ricevente o entrambe sono assai sopraelevate come nella tecnica delle comunicazioni ad onde ultra corte e microonde per il radar.

I transponder AIS operano in VHF sfruttando quindi l'onda diretta che viaggia dal trasmettitore al ricevitore, che devono essere visibili l'un l'altro. In realtà la traiettoria dell'onda non è esattamente una retta, ma segue quasi la curvatura terrestre determinando degli ampi archi di cerchio a seguito della rifrazione determinata dalla diversa densità degli strati dell'atmosfera. Quindi la propagazione nella troposfera risente della variazione dell’indice di rifrazione ***n***, che diminuisce con la quota. Così si viene a determinare una traiettoria concava verso la superficie terrestre con un aumento della portata; l’incremento di distanza di collegamento, per effetto della rifrazione, è di circa 4/3 rispetto al limite di visibilità ottico.

I segnali che sono emessi in gamma VHF si propagano con una portata (D) massima teorica che dipende essenzialmente dalla quota delle antenne, ricevente e trasmittente, in portata ottica (H_1, H_2), e, nel caso della banda VHF marina, si esprime con la relazione:

$$\mathbf{D = 2{,}35\ (\bullet\ H_1 + \bullet\ H_2)}$$ con D espresso in miglia nautiche, H_1 e H_2 in metri.

Vale la pena fare un cenno ad un’altra particolare situazione che si potrebbe creare in fase di propagazione per onda diretta: l’***effetto condotto***. Questo si verifica quando si ha una variazione anomala di pressione, umidità e temperatura, ciò provoca che l’indice di rifrazione ***n*** ad una certa altezza decresce molto più rapidamente con la quota andando a creare una sorta di zona di transizione in cui l’aria cambia le sue caratteristiche in maniera molto rapida. Questa circostanza crea per le onde e.m. uno strato di intrappolamento, in cui subisce una serie di “rimbalzi” tali da comportare un aumento della portata rispetto alla propagazione normale..

Questo fenomeno si verifica, ad esempio, in mare quando aria fredda si insinua lungo la superficie e solleva preesistenti masse di area calda generando una zona di transizione dove l'aria cambia rapidamente e crea uno strato di intrappolamento per le onde trasmesse in VHF.

1.1.3 Modulazione GMSK/FM.

Per *modulazione* s'intende la variazione prodotta su una grandezza fisica di un segnale, in dipendenza dell'informazione che si vuole trasmettere.

Nella trasmissione di una sequenza numerica il processo di modulazione assolve la funzione di far corrispondere forme d'onda che si adattino alle caratteristiche fisiche del mezzo trasmissivo.

Le principali caratteristiche richieste ad una tecnica di modulazione da utilizzare in un sistema di radiocomunicazione sono:

- elevata efficienza spettrale al fine di poter servire un elevato numero di utenti con bande di frequenza contenute;
- robustezza nei confronti del rumore termico e degli altri tipi di disturbo (in particolare le interferenze), al fine di garantire un'elevata qualità del servizio;
- buona insensibilità alle distorsioni introdotte da elementi non lineari, in modo da consentire agli stadi finali degli amplificatori in trasmissione di lavorare in prossimità della saturazione (zona non lineare) dove più alta è l'efficienza (ovvero più basso è il consumo).

Il metodo più semplice per inviare dati attraverso una modulazione di frequenza FM consiste nello spostare la frequenza della portante RF in uscita su un predeterminato valore per rappresentare l' "1" logico e su uno opposto per rappresentare il valore logico '0'. Questa tecnica è nota come ***Frequency Shift Keying*** (FSK), essa può dare ottimi risultati ma richiede una larghezza di banda molto ampia.

Una tecnica che offre un buon compromesso nei confronti dei vari parametri sopra indicati e è la tecnica di modulazione ***GMSK*** (***G****aussian* ***M****inimum* ***S****hift* ***K****eying).* Per ridurre la larghezza di banda di trasmissione richiesta, i dati di input vengono prefiltrati prima della modulazione, mediante una specifico filtro passa basso.

Per i transponder AIS la modulazione GMSK è stata scelta fra i vari tipi di modulazione candidate come compromesso fra efficienza spettrale, complessità del modulatore.

La modulazione digitale GMSK utilizzata nei ricevitore AIS lavora con un Bit Rate pari a 9,6 Kbit/s (9600 bps).

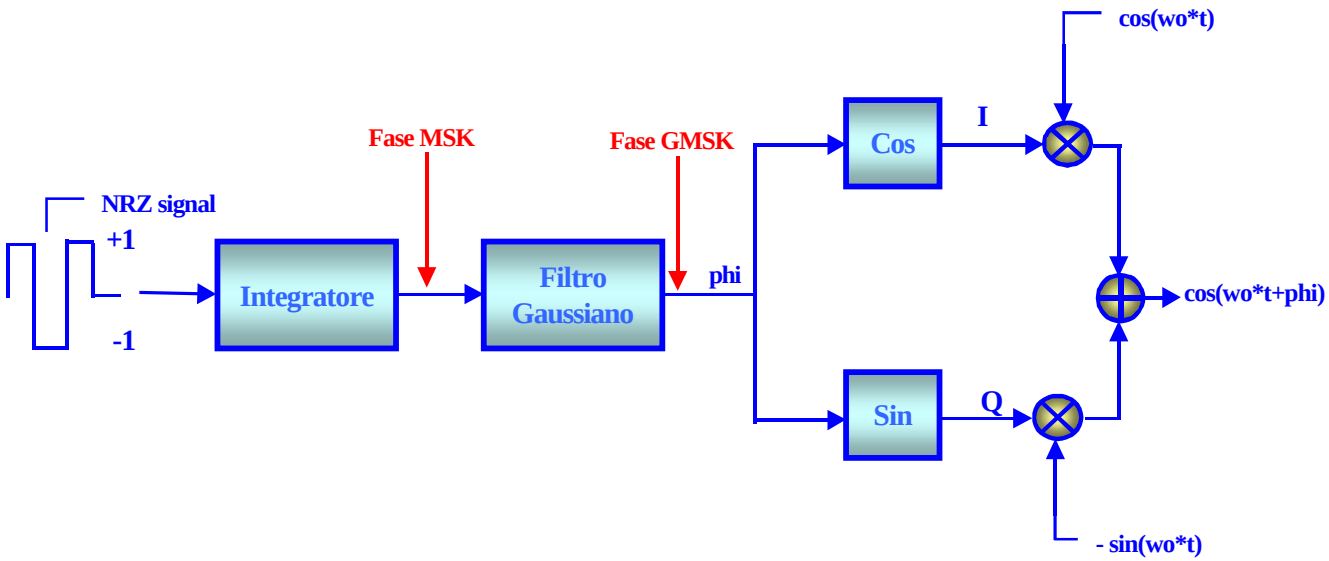


Figura 1.10: schema a blocchi del modulatore GMSK.

1.2 Accesso al canale di comunicazione.

Per l'invio delle informazione dinamiche delle unità navali in movimento esistono diverse modalità di trasmissione nell'utilizzare il canale di comunicazione offerto sui due canali AIS 1 e AIS 2, i quali vengono suddivisi in *timeslots* che possono essere impegnati con diverse tecniche.

Le *base stations* terrestri trasmettono secondo degli schemi fissi, in modo da evitare problemi di collisioni, la tecnica utilizzata è la FATDMA (*Fixed Access Time Division Multiple Access*).

Invece gli *Aids to Navigation* possono lavorare sia in FATDMA che in modalità casuale, ovvero in RATDMA (*Random Access Time Division Multiple Access*).

Diverso è il discorso per i trasponder a bordo, che, ovviamente, non possono avere un meccanismo di accesso fisso (a meno che non venga standardizzato uno schema FTDMA a livello mondiale). Quindi per ottenere le migliori performance occorre adoperare tecniche che consentano il minor numero di collisioni in forza di un algoritmo che porti ad una efficiente allocazione del canale.

I trasponder in classe A adoperano un approccio *Self-Organized Time Division Multiple Access*: SOTDMA, un protocollo semplice ed efficace che, senza il bisogno di un controllo centrale, grazie alla sincronizzazione assicurata dall'*Universal Time Clock* del ricevitore GPS riesce ad auto-organizzare e gestire la trasmissione nei *timeslots.*

Quelli in classe B usano un *Carrier-Sense Time Division Multiple Access*: il protocollo CSTDMA "ascolta la portante" prima di trasmettere, se trova il canale occupato aspetta e riprova successivamente secondo una modalità di trasmissione stabilita a priori.

1.2.1 SOTDMA.

Il protocollo SOTDMA consénte di gestire fino a 2250 rapporti al minuto sullo stesso canale radio.

SOTDMA permette l'uso in divisione di tempo del canale radio per brevi trasmissioni (*bursts*) tra più utenti senza collisioni di accesso e senza necessità di interventi di sincronizzazione degli accessi.

L'accesso al canale opera senza una funzione centrale di polling e richiede che vi sia una precisa sincronizzazione temporale tra tutti gli utenti. La precisione richiesta viene ottenuta ricavando il tempo UTC dai satelliti della costellazione GPS.

ll canale radio è temporalmente organizzato dal protocollo SOTDMA in mainframe di 1 minuto, ciascun mainframe è a sua volta suddiviso in 2250 *timeslot*, ognuno della durata di 26,67 ms, che costituiscono la finestra di accesso di un *burst* di trasmissione.

La trasmissione dei dati avviene a 9600 bps, per cui in un *timeslot* (di durata 26,67 ms) sono è possibile scambiare messaggi di 256 bit. Per trasmettere 1 bit occorrono pertanto 104,17 μs.

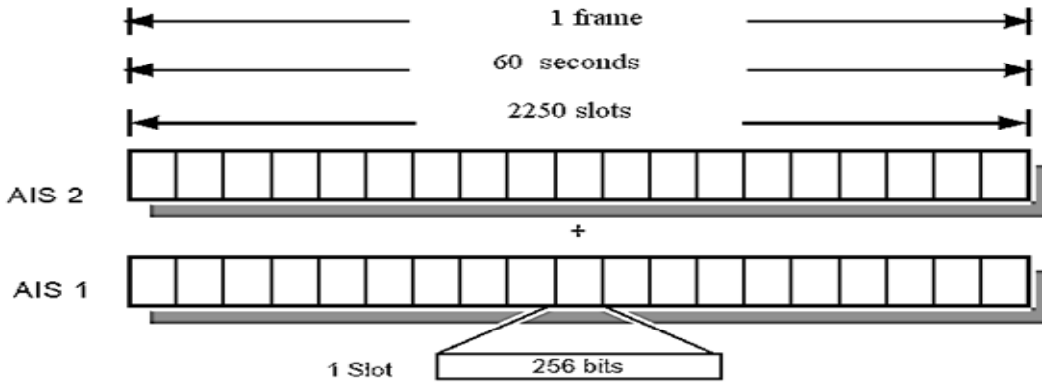


Figura 1.11: suddivisione del *timeslot* operata dal protocollo SOTDMA.

Le stazioni AIS in trasmissione si sincronizzano continuamente l'una con l'altra per evitare di utilizzare i medesimi slot temporali. Se non esistono slot disponibili la stazione non può trasmettere, fatto questo che abbassa le capacità di trasmissione dati da parte del sistema, si rende pertanto necessario fare in modo di limitare il numero di slot occupati.

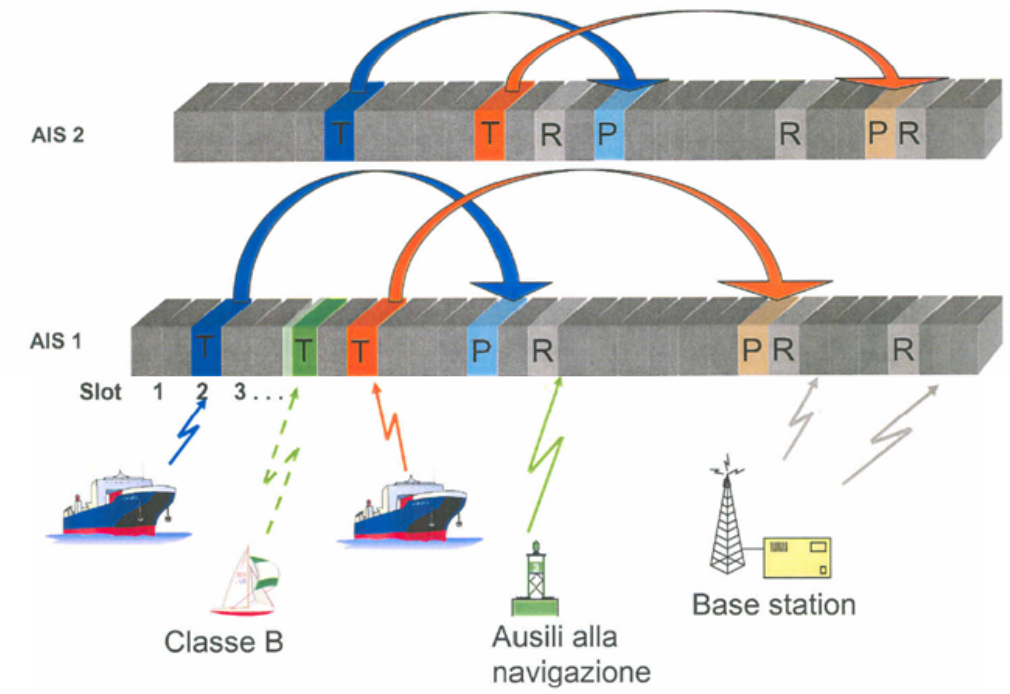


Figura 1.12: meccanismo di sincronizzazione.

La fase iniziale dell'algoritmo prevede un ascolto del canale per un minuto, dopo il quale comincia la fase di trasmissione vera e propria.

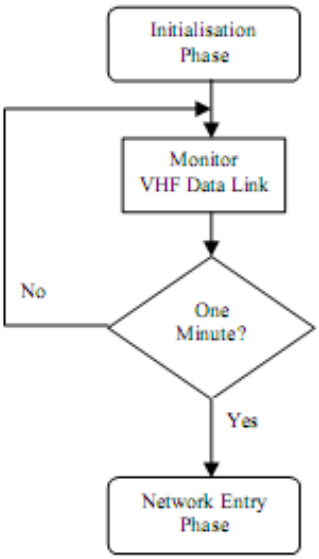


Figura 1.13: fase di inizializzazione nel protocollo SOTDMA.

Durante questa fase iniziale viene costruita una mappa della situazione presente intorno al trasponder che si avvia a trasmettere, quindi raccoglie informazione su quali slot sono stai occupati e quanti altri trasmettitori ci sono attivi.

Quando inizia la trasmissione, impegna il primo slot libero in maniera da "presentarsi" agli altri eventuali trasmettitori.

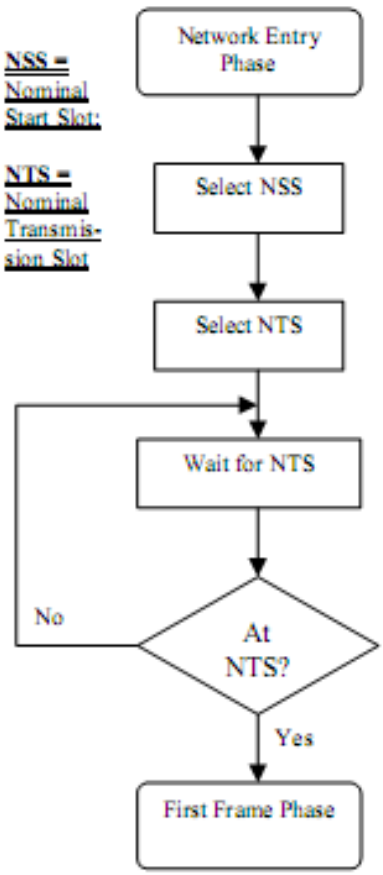


Figura 1.16: fase di presentazione nel protocollo SOTDMA.

I principi utilizzati nell'elaborazione dell'algoritmo sono quelli indicati nella tabella seguente, in cui sono indicati i criteri di scelta per ogni tipo di parametro.

Symbol	Name	Description	Min	Max
NSS	Nominal Start Slot	This is the first slot used by a station to announce itself on the data link. Other repeatable transmissions are generally selected with the NSS as a reference.	0	2249
NS	Nominal Slot	The nominal slot is used as the centre around which slots are selected for transmission of position reports. For the first transmission in a frame, the NSS and NS are equal. Any NS is derived using the equation below: **NS = NSS + (n * NI); (0 <= n < RR)**	0	2249
NI	Nominal Increment	The nominal increment is given in number of slots and is derived using the equation below: NI = 2250 / RR	75	1225
RR	Report Rate	This is the desired number of position reports per frame. When a station uses a report rate of less than one report per frame, ITDMA allocations are used. Otherwise SOTDMA is used.	1/3	30
SI	Selection Interval	Selection Interval. The selection interval is the collection of slots which can be candidates for position reports. The SI is derived using the equation below: SI = {NS – (0.1 * NI) to NS + (0.1 * NI)}	0.2 * NI	0.2 * NI
NTS	Transmission Slot	The slot, within a selection interval, currently used for transmissions within that interval.	0	2249
TMO_MIN	Minimum Timeout	The minimum number of frames that an SOTDMA allocation will occupy a specific slot.	3	3
TMO_MAX	Maximum Timeout	The maximum number of frames that an SOTDMA allocation will occupy a specific slot.	TMO_MIN	8

Il NSS per annunciare la propria presenza viene selezionato casualmente tra lo slot corrente e lo slot seguente, mentre il NTS viene scelto tra quelli all'interno dell'intervallo SI, ovvero un insieme di possibili slot ricavati tramite la relazione

$$SI = \{ NS - (0{,}1 * NI) \text{ to } NS + (0{,}1 * NI) \}.$$

Una volta arrivati allo slot scelto per la trasmissione (NTS), inizia la fase di trasmissione dei frame secondo lo schema seguente, in cui l'offset viene utilizzato per indicare la distanza tra la trasmissione in corso e quella successiva, esso viene incrementato in base alle intenzioni di trasmissioni della stazione.

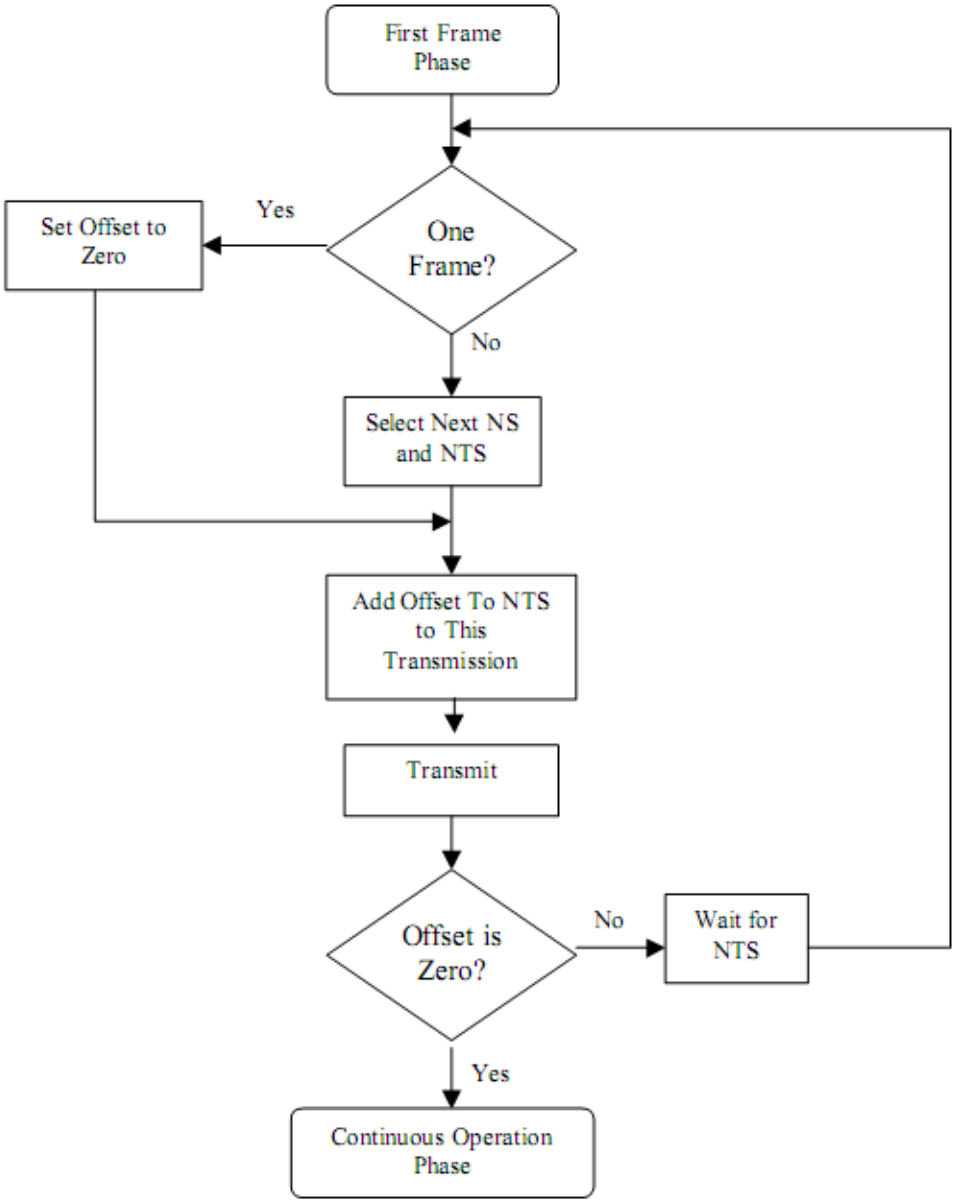


Figura 1.15: fase di trasmissione del primo frame.

Fino a quando l'offset ha un valore diverso da zero, il transponder aspetta il successivo NTS e ripete il ciclo. Quando l'offset viene impostato a zero, vuol dire che si è giunti al termine della fase di invio frame ed il trasponder inizia a trasmettere in modalità continua.

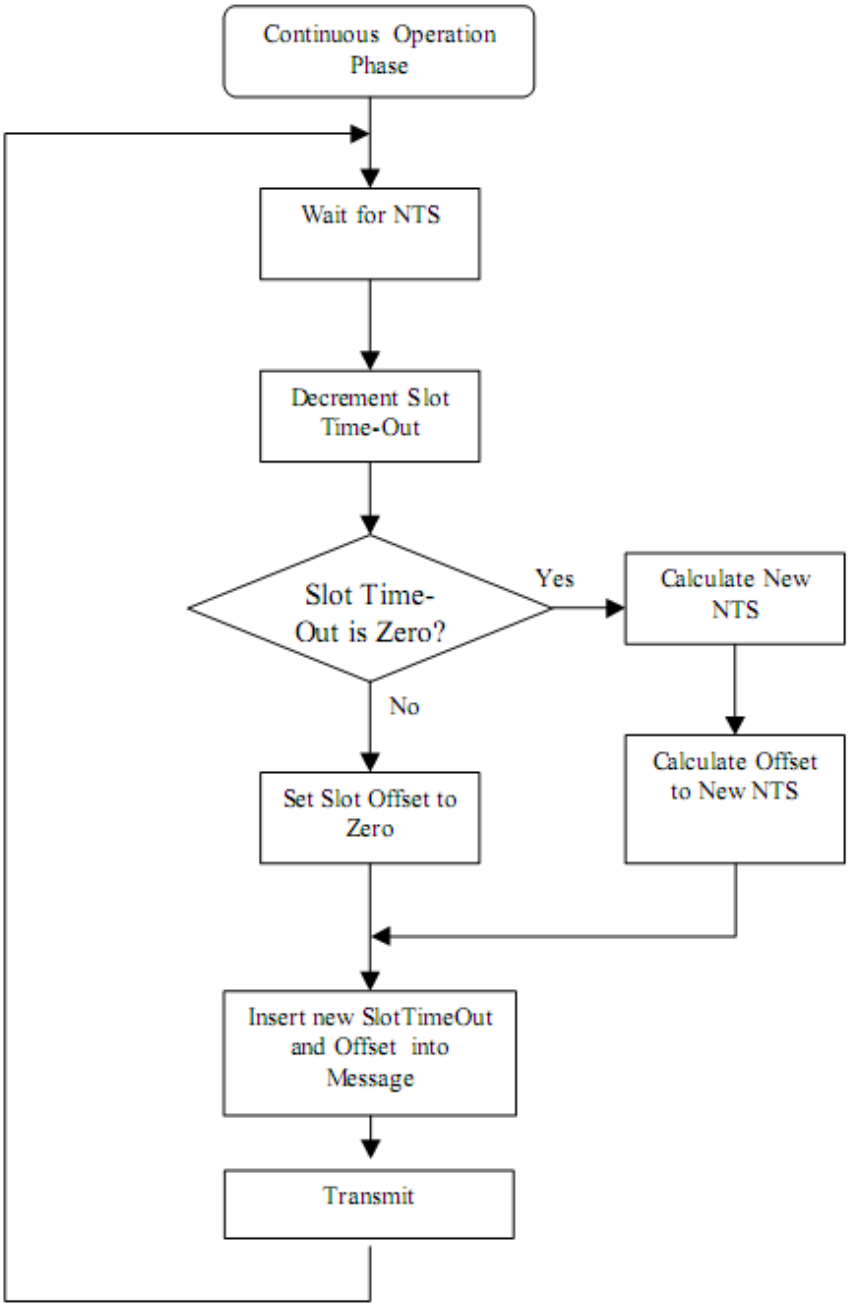


Figura 1.16: trasmissione in modalità continua auto-organizzata.

Il trasponder continua a trasmettere in modalità continua fino a quando non viene spento, ogni utente ascolta costantemente il canale radio e memorizza tutte le informazioni ricevute su un data-base interno al transponder.

Dall'analisi del data-base il transponder è in grado di generare una mappa degli utenti presenti, individuandone l'identità, la posizione, i dati cinematici di navigazione ed altre eventuali informazioni inviate. Inoltre, analizzando in ciascun messaggio la parte relativa all'intenzione di utilizzazione futura dei timeslot, il transponder è in grado di ricavare la

mappa dell'occupazione dei *timeslot* nei successivi 4 minuti, andando così a prenotare lo slot libero per il suo prossimo invio.

Questo meccanismo, molto semplice nel suo concetto, è alla base dell'efficienza del protocollo SOTDMA.

1.1.1.2 La struttura del messaggio SOTDMA.

Dei 256 bit disponibili in un *timeslot* allocato tramite il protocollo SOTDMA solo 168 costituiscono il messaggio utile, mentre gli altri 88 bit sono utilizzati per la struttura di protocollo (sincronizzazione, start e stop *message*, controllo di parità).

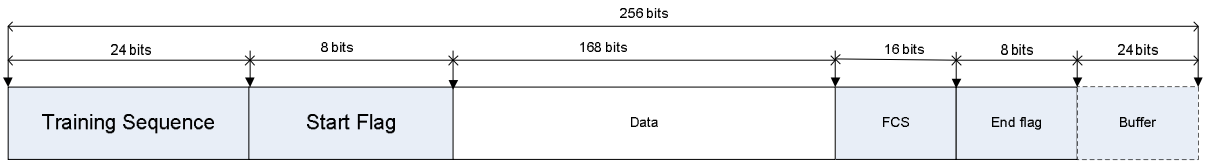


Figura 1.17: il pacchetto dati.

Nei 168 bit utili è contenuto un blocco di 19 bit, importante per il processo di auto-organizzazione dell'accesso al canale radio. Il blocco fornisce informazioni usate dall'algoritmo di allocazione degli slot del protocollo SOTDMA e informazioni sullo stato di sincronizzazione.

La suddivisione dei bit dedicati è la seguente:

Ramp Up	8 bits	
Training Sequence	24 bits	Necessary for synchronisation
Start Flag	8 bits	In accordance with HDLC.
Data	168 bits	Default
CRC	16 bits	In accordance with HDLC.
End Flag	8 bits	In accordance with HDLC.
Buffering	24 bits	Bit stuffing and distance delays.
Total	**256 bits**	

Figura 1.18: suddivisione bit del pacchetto.

1.2.2 RATDMA.

Per l'invio delle informazioni statiche, che viene effettuato dalle unità in movimento ogni 6 minuti indipendentemente dalla velocità con cui procedono, viene adoperato l'algoritmo schematizzato del diagramma di flusso che segue.

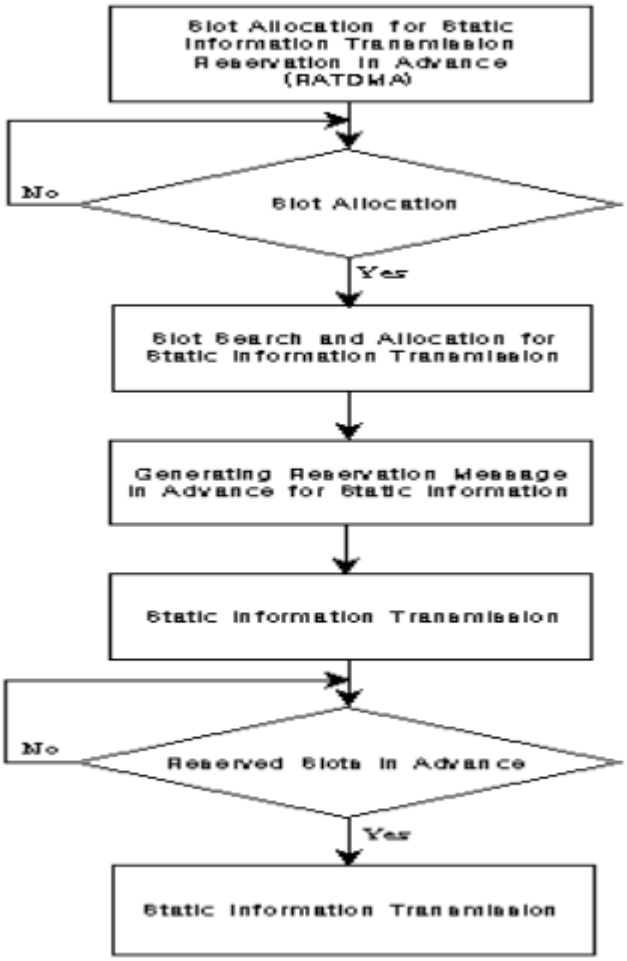


Figura 1.19: algoritmo per la trasmissione delle informazioni statiche.

1.3 OSI layer per l'AIS.

E' possibile considerare l'AIS secondo il modello OSI (Open System Interconnection), nella figura seguente sono illustrati gli strati di una stazione AIS, da quello fisico a quello di trasporto, e gli strati offerti al livello applicativo (sessione, presentazione e applicazione).

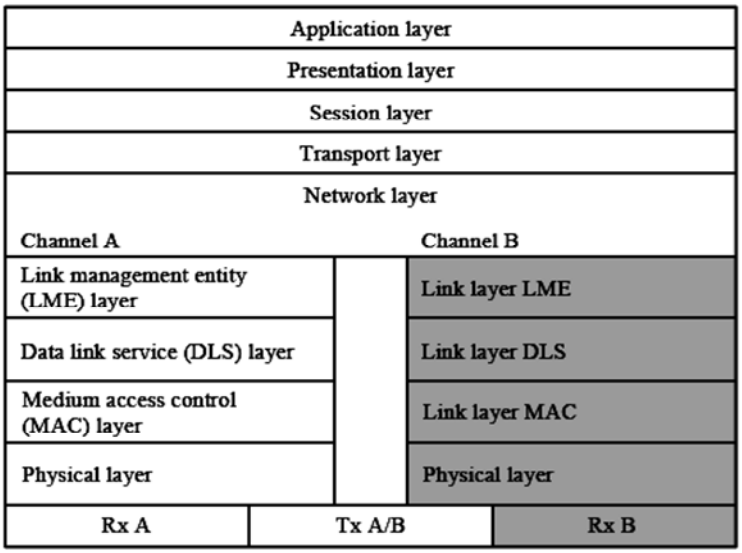


Figura 1.20: AIS nel modello OSI.

Soffermandoci sui primi 4 layers (*physical layer*, *link layer*, *network layer, transport layer*), lo strato "trasporto" si occupa della conversione dei dati in pacchetti e della loro trasmissione/ricezione nella giusta sequenza.

Lo strato di rete è invece responsabile della corretta assegnazione di priorità ai messaggi da trasportare, occupandosi della risoluzione dei problemi di congestione del data link.

Il link layer, suddiviso in 3 sottostrati, specifica come deve essere effettuato l'incapsulamento e la rilevazione degli errori nella fase di trasferimento dati. In esso distinguiamo:

- il *Medium Access Control* (MAC), che si occupa dell'accesso al mezzo di trasferimento, garantendo la sincronizzazione tramite il protocollo SOTDMA.

- Il *Data Link Service* (DLS), che raggruppa i bit del livello fisico in pacchetti. controlla e gestisce gli errori di trasmissione e regola il flusso della trasmissione fra sorgente e destinatario (controllo di flusso) in base alle informazioni passate dallo strato MAC.

- Il *Link Management Entity* (LME) controlla le operazioni svolte dagli DLS, MAC e fisico.

Lo strato fisico è responsabile del trasferimento del bit-stream dall'originatore verso lo strato superiore, con dati codificati in NRZI (*non return to zero inverted*), nella tabella che segue si riportato i principali parametri caratterizzanti tale layer.

Parametro	Valore
Frequency range	156.025 - 162.025 (banda marittima)
Canali (MHZ)	AIS1 : 161.975 - AIS2 : 162.025
Larghezza Canale (kHz)	12.5 in acque territoriali / 25 in alto mare
Bit Rate (bit/s)	9600
Potenza di uscita (w)	High =12.5 / Low = 2
Modulazione	GMSK
Codifica	NRZI
Metodo di accesso	SOTDMA

1.4 Tipi di messaggi nell'AIS.

Le informazioni trasmesse dal livello di presentazione di un trasponder AIS vengono suddivise in diversi tipi di messaggio con differenti validità temporali, pertanto richiedono una diversa frequenza di aggiornamento.

Discorso a parte per i *safety message* (messaggi relativi ad avvisi per la sicurezza) che vengono inviati secondo le esigenze del momento, senza alcuna schedulazione prestabilita. I *safety message* hanno un formato libero inserito manualmente e indirizzabile ad uno specifico destinatario, ad un gruppo di destinatari o trasmesso a tutte le navi e le stazioni in grado di ricevere le informazioni.

La velocità delle navi ed il loro stato di navigazione vanno ad aggiornare i messaggi "dinamici", garantendo la necessaria accuratezza per identificare la posizione della nave. Analogo discorso viene applicato ai contenuti dei messaggi di informazione "statici" e "relativi al viaggio", che hanno però un *rate* di trasmissione minore, in modo da non sovrapporsi a quelli con una maggiore priorità.

Le informazioni "statiche" della nave vengono impostate nella configurazione del trasmettitore e prevede un cambiamento solamente nel caso in cui la nave cambi il proprio nome o subisca una variazione della classe.

Informazioni Statiche	Trasmesse ogni 6 minuti e/o su richiesta dell'autorità competente
MMSI	Maritime Mobile Service Identity. Settatato al momento dell'installazione – notare che potrebbe richiedere una modifica se la nave dovesse cambiare proprietà
Call Sign e nome	Settato al momento dell'installazione – notare che potrebbero richiedere una modifica se la nave dovesse cambiare proprietà
Numero IMO	Settato al momento dell'installazione
Lunghezza e larghezza	Settato al momento dell'installazione o in caso di variazione
Tipo di nave	Selezionabile da una lista
Ubicazione e posizionamento dell'antenna fissa	Settato al momento dell'installazione o potrebbe essere cambiato per i vascelli bi-direzionali o per quelli montati con molteplici posizioni di antenna fissa
Altezza sopra la chiglia	Settata al momento dell'installazione (da poppa a prua e da dritta a sinistra) trasmessa a discrezione del Comandante della nave e su richiesta dell'autorità competente

Le informazioni "dinamiche" vengono aggiornate automaticamente dai sensori della nave connessi al transponder AIS.

Informazioni Dinamiche	Trasmesse con una frequenza che dipende dalla velocità e dalla rotta
posizione della nave	Aggiornata automaticamente dalla posizione dei sensori di posizione connessi all'AIS. La precisione dell'indicazione può essere maggiore
Indicatore di tempo e di posizione in UTC	Aggiornata automaticamente dal principale sensore di posizionamento della nave connesso all'AIS (ad es. il GPS)

Course over ground (COG)	Aggiornata automaticamente dal principale sensore di posizionamento della nave connesso all'AIS, a condizione che quel sensore calcoli la COG (questa informazione potrebbe non essere disponibile)
Speed over ground (SOG)	Aggiornata automaticamente dal principale sensore di posizionamento della nave connesso all'AIS, a condizione che quel sensore calcoli il SOG (questa informazione potrebbe non essere disponibile)
Heading	Aggiornata automaticamente dal sensore heading connesso all'AIS
Stato di navigazione	Le informazioni sullo stato di navigazione devono essere inserite manualmente ed eventualmente modificate laddove necessario, per esempio: - Navigazione tramite motore - Nave all'ancora - Not under command (NUC) - Nave in condizioni di manovrabilità ridotta - Nave ormeggiata - Limitata dal pescaggio - In secca - Impegnata in operazioni di pesca - Navigazione tramite vela In pratica poiché tutti questi dati riguardano la COLREG, qualsiasi cambiamento che necessita dovrebbe venire attuato allo stesso tempo in cui vengono modificate le luci o la forma.
Rate of turn (ROT)	Aggiornata automaticamente dal sensore ROT della nave o derivata dalla girobussola. (Questa informazione potrebbe non essere disponibile).

Le informazione “relative al viaggio” vengono inserite manualmente e aggiornate durante il tragitto.

Informazioni relative al viaggio	Trasmesse ogni sei minuti, quando un dato è modificato oppure su richiesta
Pescaggio della nave	Da inserire manualmente all’inizio del tragitto usando il massimo del pescaggio e approntando delle modifiche dove richiesto (ad esempio dopo lo scarico della zavorra prima di entrare in porto)
Carico pericoloso (tipo)	Da inserire manualmente all’inizio del tragitto confermando che si tratti di trasporto di carico pericoloso o meno vale a dire: - DG merci pericolose - HS sostanze dannose - MP inquinamento marino Non sono richieste informazioni circa le quantità
Destinazione ed ETA	A discrezione del Comandante della nave. Da inserire manualmente all’inizio della tratta, aggiornandole laddove ce ne sia la necessità
Ruolino di marcia (waypoints)	Descrizione testuale inserita manualmente all’inizio della tratta e aggiornata se richiesto.
Numero di persone a bordo	Incluso l’equipaggio. A discrezione del Comandante della nave e solo su richiesta di un’autorità competente.

Le informazioni dinamiche, statiche e relative al viaggio vengono trasmesse tramite un insieme di ventidue diversi tipi di messaggio (così come specificato nella raccomandazione ITU-.1371) che, non solo contengono i dati di dettaglio della nave, ma anche tutte quelle informazioni e comandi necessari alla gestione e sincronizzazione del sistema.

Il seguente elenco mostra il raggruppamento dei messaggi di primario interesse per gli operatori dell'AIS e indica le modalità operative associate ad ogni messaggio (AU = autonoma, AS = assegnato, IN = polling/interrogativo).

Verrà poi di seguito fornita una descrizione più dettagliata della struttura dei messaggi più rilevanti.

Identificativo Messaggio	Descrizione	Modalità di Operazione
1,2,3	Rapporto di posizione – pianificato, assegnato o in risposta al polling	AU,AS
4	Rapporto di Base Station - posizione, UTC/data e corrente numero slot	AS
5	Dati statici e relativi al viaggio- Classe A	AU,AS
6,7,8	Messaggi binari – indirizzati, per conoscenza o trasmessi	AU,AS,IN
9	Rapporto SAR standard di posizione aeromobile	AU,AS
10,11	UTC/data – richiesta di informazioni e risposta	AS,IN
12,13,14	Messaggio relativo alla sicurezza – indirizzato, per conoscenza o trasmesso	AS,IN
15	Interrogazione- richiesta di specifici tipi di messaggio	AU,AS,IN
16	Modalità di assegnazione del comando – da parte dell'autorità competente	AS
17	DGNSS trasmissione del messaggio binario	AS
18,19	Rapporto di posizione Classe B - rapporti standard ed estesi	AU,AS
20	Gestione del link dati – riserva delle slot per stazione base	AS
21	Rapporto di ausilio alla navigazione – rapporto di stato e posizione	AU,AS,IN
22	Gestione del Canale	AS

1.4.1 Descrizione e codifica ITU per i messaggi AIS

Ognuno dei messaggi AIS è costituito da una serie di campi codificati secondo la seguente tabella ASCII[9] a 6 bit.

6-Bit ASCII				Standard ASCII			6-Bit ASCII				Standard ASCII		
Chr	Dec	Hex	Binary	Dec	Hex	Binary	Chr	Dec	Hex	Binary	Dec	Hex	Binary
@	0	0x00	00 0000	64	0x40	0100 0000	!	33	0x21	10 0001	33	0x21	0010 0001
A	1	0x01	00 0001	65	0x41	0100 0001	"	34	0x22	10 0010	34	0x22	0010 0010
B	2	0x02	00 0010	66	0x42	0100 0010	#	35	0x23	10 0011	35	0x23	0010 0011
C	3	0x03	00 0011	67	0x43	0100 0011	$	36	0x24	10 0100	36	0x24	0010 0100
D	4	0x04	00 0100	68	0x44	0100 0100	%	37	0x25	10 0101	37	0x25	0010 0101
E	5	0x05	00 0101	69	0x45	0100 0101	&	38	0x26	10 0110	38	0x26	0010 0110
F	6	0x06	00 0110	70	0x46	0100 0110	`	39	0x27	10 0111	39	0x27	0010 0111
G	7	0x07	00 0111	71	0x47	0100 0111	(	40	0x28	10 1000	40	0x28	0010 1000
H	8	0x08	00 1000	72	0x48	0100 1000	)	41	0x29	10 1001	41	0x29	0010 1001
I	9	0x09	00 1001	73	0x49	0100 1001	*	42	0x2A	10 1010	42	0x2A	0010 1010
J	10	0x0A	00 1010	74	0x4A	0100 1010	+	43	0x2B	10 1011	43	0x2B	0010 1011
K	11	0x0B	00 1011	75	0x4B	0100 1011	,	44	0x2C	10 1100	44	0x2C	0010 1100
L	12	0x0C	00 1100	76	0x4C	0100 1100	-	45	0x2D	10 1101	45	0x2D	0010 1101
M	13	0x0D	00 1101	77	0x4D	0100 1101	.	46	0x2E	10 1110	46	0x2E	0010 1110
N	14	0x0E	00 1110	78	0x4E	0100 1110	/	47	0x2F	10 1111	47	0x2F	0010 1111
O	15	0x0F	00 1111	79	0x4F	0100 1111	0	48	0x30	11 0000	48	0x30	0011 0000
P	16	0x10	01 0000	80	0x50	0101 0000	1	49	0x31	11 0001	49	0x31	0011 0001
Q	17	0x11	01 0001	81	0x51	0101 0001	2	50	0x32	11 0010	50	0x32	0011 0010
R	18	0x12	01 0010	82	0x52	0101 0010	3	51	0x33	11 0011	51	0x33	0011 0011
S	19	0x13	01 0011	83	0x53	0101 0011	4	52	0x34	11 0100	52	0x34	0011 0100
T	20	0x14	01 0100	84	0x54	0101 0100	5	53	0x35	11 0101	53	0x35	0011 0101
U	21	0x15	01 0101	85	0x55	0101 0101	6	54	0x36	11 0110	54	0x36	0011 0110
V	22	0x16	01 0110	86	0x56	0101 0110	7	55	0x37	11 0111	55	0x37	0011 0111
W	23	0x17	01 0111	87	0x57	0101 0111	8	56	0x38	11 1000	56	0x38	0011 1000
X	24	0x18	01 1000	88	0x58	0101 1000	9	57	0x39	11 1001	57	0x39	0011 1001
Y	25	0x19	01 1001	89	0x59	0101 1001	:	58	0x3A	11 1010	58	0x3A	0011 1010
Z	26	0x1A	01 1010	90	0x5A	0101 1010	;	59	0x3B	11 1011	59	0x3B	0011 1011
[	27	0x1B	01 1011	91	0x5B	0101 1011	<	60	0x3C	11 1100	60	0x3C	0011 1100
\	28	0x1C	01 1100	92	0x5C	0101 1100	=	61	0x3D	11 1101	61	0x3D	0011 1101
]	29	0x1D	01 1101	93	0x5D	0101 1101	>	62	0x3E	11 1110	62	0x3E	0011 1110
^	30	0x1E	01 1110	94	0x5E	0101 1110	?	63	0x3F	11 1111	63	0x3F	0011 1111
–	31	0x1F	01 1111	95	0x5F	0101 1111							
Space	32	0x20	10 0000	32	0x20	0010 0000							

Figura 1.21: tabella ASCII a 6 bit.

Vediamo ora il dettaglio dei campi dei messaggi 1, 2 e 3 che si occupano del *position report*.

[9] Estratta dalla raccomandazione ITU 1371-3

Parameter	Number of bits	Description
Message ID	6	Identifier for this message 1, 2 or 3
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
User ID	30	MMSI number
Navigational status	4	0 = under way using engine, 1 = at anchor, 2 = not under command, 3 = restricted manoeuvrability, 4 = constrained by her draught, 5 = moored, 6 = aground, 7 = engaged in fishing, 8 = under way sailing, 9 = reserved for future amendment of navigational status for ships carrying DG, HS, or MP, or IMO hazard or pollutant category C, high speed craft (HSC), 10 = reserved for future amendment of navigational status for ships carrying dangerous goods (DG), harmful substances (HS) or marine pollutants (MP), or IMO hazard or pollutant category A, wing in grand (WIG); 11-14 = reserved for future use, 15 = not defined = default
		0 = low (= 10 m) 0 = default Longitude in 1/10 000 min (±180°, East = positive (as per 2's complement), West = negative (as per 2's complement). 181= (6791AC0$_h$) = not available = default)
Latitude	27	Latitude in 1/10 000 min (±90°, North = positive (as per 2's complement), South = negative (as per 2's complement). 91° (3412140h) = not available = default)
COG	12	Course over ground in 1/10 = (0-3599). 3600 (E10$_h$) = not available = default. 3 601-4 095 should not be used
True heading	9	Degrees (0-359) (511 indicates not available = default)
Time stamp	6	UTC second when the report was generated by the electronic position system (EPFS) (0-59, or 60 if time stamp is not available, which should also be the default value, or 61 if positioning system is in manual input mode, or 62 if electronic position fixing system operates in estimated (dead reckoning) mode, or 63 if the positioning system is inoperative)
special manoeuvre indicator	2	0 = not available = default 1 = not engaged in special manoeuvre 2 = engaged in special manoeuvre (i.e.: regional passing arrangement on Inland Waterway)
Spare	3	Not used. Should be set to zero. Reserved for future use.
RAIM-flag	1	Receiver autonomous integrity monitoring (RAIM) flag of electronic position fixing device; 0 = RAIM not in use = default; 1 = RAIM in use. See Table
Communication state	19	See Table 46
Number of bits	168	

Figura 1.22: struttura messaggio *position report.*

La struttura del messaggio 4 è analoga a quella del messaggio 11. Il messaggio 4 è quello trasmesso periodicamente dalla *base station* con le informazioni di sincronizzazione UTC e i dati di posizione, mentre il messaggio 11 viene usato dalle stazioni mobili come risposta ad una interrogazione effettuata tramite il messaggio 10.

Parameter	Number of bits	Description
Message ID	6	Identifier for this message 4 or 11 4 = UTC and position report from base station 11 = UTC and position response from mobile station
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. Refer to § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
User ID	30	MMSI number
UTC year	14	1-9999; 0 = UTC year not available = default
UTC month	4	1-12; 0 = UTC month not available = default; 13-15 not used
UTC day	5	1-31; 0 = UTC day not available = default
UTC hour	5	0-23; 24 = UTC hour not available = default; 25-31 not used
UTC minute	6	0-59; 60 = UTC minute not available = default; 61-63 not used
UTC second	6	0-59; 60 = UTC second not available = default; 61-63 not used
Position accuracy	1	1 = high (> 10 m) 0 = low (< 10 m) 0 = default The PA flag should be determined in accordance with Table 47
Longitude	28	Longitude in 1/10 000 min (±180°, East = positive (as per 2's complement), West = negative (as per 2's complement); 181= ($6791AC0_h$) = not available = default)
Latitude	27	Latitude in 1/10 000 min (±90°, North = positive (as per 2's complement), South = negative (as per 2's complement); 91 = (3412140_h) = not available = default)
Type of electronic position fixing device	4	Use of differential corrections is defined by field position accuracy above: 0 = undefined (default) 1 = global positioning system (GPS) 2 = GNSS (GLONASS) 3 = combined GPS/GLONASS 4 = Loran-C 5 = Chayka 6 = integrated navigation system 7 = surveyed 8 = Galileo 9-15 = not used
Spare	10	Not used. Should be set to zero. Reserved for future use
RAIM-flag	1	RAIM (Receiver autonomous integrity monitoring) flag of electronic position fixing device; 0 = RAIM not in use = default; 1 = RAIM in use see Table 47
Communication state	19	SOTDMA communication state as described in § 3.3.7.2.1, Annex 2
Number of bits	168	

Figura 1.23: struttura dei messaggi 4 e 11.

Il messaggio 5 viene usato solo dai transponder in classe A e da quelli montati sui velivoli SAR, qualora ci siano da trasmettere informazioni relative al viaggio.

Parameter	Number of bits	Description
Message ID	6	Identifier for this Message 5
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. Refer to § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
User ID	30	MMSI number
AIS version indicator	2	0 = station compliant with Recommendation ITU-R M.1371-1 1 = station compliant with Recommendation ITU-R M.1371-3 2-3 = station compliant with future editions
IMO number	30	1-999999999; 0 = not available = default
Call sign	42	7 = 6 bit ASCII characters, @@@@@@@ = not available = default
Name	120	Maximum 20 characters 6 bit ASCII, as defined in Table 44 "@@@@@@@@@@@@@@@@@@@@" = not available = default
Type of ship and cargo type	8	0 = not available or no ship = default 1-99 = as defined in § 3.3.2 100-199 = reserved, for regional use 200-255 = reserved, for future use
Overall dimension/ reference for position	30	Reference point for reported position. Also indicates the dimension of ship (m) (see Fig. 42 and § 3.3.3)
Type of electronic position fixing device	4	0 = undefined (default) 1 = GPS 2 = GLONASS 3 = combined GPS/GLONASS 4 = Loran-C 5 = Chayka 6 = integrated navigation system 7 = surveyed 8 = Galileo, 9-15 = not used
ETA	20	Estimated time of arrival; MMDDHHMM UTC Bits 19-16: month; 1-12; 0 = not available = default Bits 15-11: day; 1-31; 0 = not available = default Bits 10-6: hour; 0-23; 24 = not available = default Bits 5-0: minute; 0-59; 60 = not available = default
Maximum present static draught	8	In 1/10 m, 255 = draught 25.5 m or greater, 0 = not available = default; in accordance with IMO Resolution A.851
Destination	120	Maximum 20 characters using 6-bit ASCII; @@@@@@@@@@@@@@@@@@@@ = not available
DTE	1	Data terminal equipment (DTE) ready (0 = available, 1 = not available = default) (see § 3.3.1)
Spare	1	Spare. Not used. Should be set to zero. Reserved for future use.
Number of bits	424	Occupies 2 slots

Figura 1.24: struttura del messaggio 5.

Da notare che il campo *type of ship* può assumere una serie di valori così come riportati nella tabella seguente.

Identifiers to be used by ships to report their type	
Identifier No.	**Special craft**
50	Pilot vessel
51	Search and rescue vessels
52	Tugs
53	Port tenders
54	Vessels with anti-pollution facilities or equipment
55	Law enforcement vessels
56	Spare – for assignments to local vessels
57	Spare – for assignments to local vessels
58	Medical transports (as defined in the 1949 Geneva Conventions and Additional Protocols)
59	Ships according to RR Resolution No. 18 (Mob-83)

Identifiers to be used by ships to report their type			
Other ships			
First digit(1)	**Second digit(1)**	**First digit(1)**	**Second digit(1)**
1 – Reserved for future use	0 – All ships of this type	–	0 – Fishing
2 – WIG	1 – Carrying DG, HS, or MP, IMO hazard or pollutant category A	–	1 – Towing
3 – See right column	2 – Carrying DG, HS, or MP, IMO hazard or pollutant category B	3 – Vessel	2 – Towing and length of the tow exceeds 200 m or breadth exceeds 25 m
4 – HSC	3 – Carrying DG, HS, or MP, IMO hazard or pollutant category C	–	3 – Engaged in dredging or underwater operations
5 – See above	4 – Carrying DG, HS, or MP, IMO hazard or pollutant category D	–	4 – Engaged in diving operations
	5 – Reserved for future use	–	5 – Engaged in military operations
6 – Passenger ships	6 – Reserved for future use	–	6 – Sailing
7 – Cargo ships	7 – Reserved for future use	–	7 – Pleasure craft
8 – Tanker(s)	8 – Reserved for future use	–	8 – Reserved for future use
9 – Other types of ship	9 – No additional information	–	9 – Reserved for future use

Figura 1.25: valori *type of ship* (tipo di nave).

Il messaggio 6 è di tipo binario, inviato per trasmettere dati a fini di *safety*, metre il 7 è un messaggio binario di *acknowledge*. Il messaggio numero 8 è usato per trasmettere dati binari in *broadcast*.

Parameter	Number of bits	Description		
Message ID	6	Identifier for Message 6; always 6		
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. Refer to § 4.6.1, Annex 2; 0-3; default = 0; 3 = do not repeat any more		
Source ID	30	MMSI number of source station		
Sequence number	2	0-3; refer to § 5.3.1, Annex 2		
Destination ID	30	MMSI number of destination station		
Retransmit flag	1	Retransmit flag should be set upon retransmission: 0 = no retransmission = default; 1 = retransmitted		
Spare	1	Not used. Should be zero. Reserved for future use		
Binary data	Maximum 936	Application identifier	16 bits	Should be as described in § 2.1, Annex 5
		Application data	Maximum 920 bits	Application specific data
Maximum number of bits	Maximum 1 008	Occupies 1 to 5 slots subject to the length of sub-field message content. For Class B mobile AIS stations the length of the message should not exceed 2 slots.		

Parameter	Number of bits	Description
Message ID	6	Identifier for Messages 7 or 13 7 = binary acknowledge 13 = safety related acknowledge
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI number of source of this acknowledge (ACK)
Spare	2	Not used. Should be set to zero. Reserved for future use
Destination ID1	30	MMSI number of first destination of this ACK
Sequence number for ID1	2	Sequence number of message to be acknowledged; 0-3
Destination ID2	30	MMSI number of second destination of this ACK; should be omitted if no destination ID2
Sequence number for ID2	2	Sequence number of message to be acknowledged; 0-3; should be omitted if no destination ID2
Destination ID3	30	MMSI number of third destination of this ACK; should be omitted if no destination ID3
Sequence number for ID3	2	Sequence number of message to be acknowledged; 0-3; should be omitted if no destination ID3
Destination ID4	30	MMSI number of fourth destination of this ACK; should be omitted if no destination ID4
Sequence number for ID4	2	Sequence number of message to be acknowledged; 0-3. Should be omitted if there is no destination ID4
Number of bits	72-168	

Parameter	Number of bits	Description		
Message ID	6	Identifier for Message 8; always 8		
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; default = 0; 3 = do not repeat any more		
Source ID	30	MMSI number of source station		
Spare	2	Not used. Should be set to zero. Reserved for future use		
Binary data	Maximum 968	Application identifier	16 bits	Should be as described in § 2.1, Annex 5
		Application data	Maximum 952 bits	Application specific data
Maximum number of bits	Maximum 1 008	Occupies 1 to 5 slots For Class B mobile AIS stations the length of the message should not exceed 2 slots.		

Figura 1.26: struttura dei messaggi 6, 7 e 8.

Da notare che la struttura del messaggio 7 è la stessa che viene anche utilizzata per il messaggio 13, quest'ultimo ha anch'esso ruolo di *aknowledge* ma come risposta ad un messaggio di *safety* di tipo indirizzato (messaggio 12).

I messaggi appena analizzati hanno un estensione variabile e, a seconda della quantità di dati che trasportano, vanno ad occupare un numero di *timeslots* compreso tra 1 e 5.

Un altro messaggio di *position report* è il numero 9 (da trasmettere ad intervalli di 10 secondi), dedicato però all'identificazione di velivoli impegnati in operazioni di ricerca e salvataggio.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 9; always 9
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
User ID	30	MMSI number
Altitude (GNSS)	12	Altitude (derived from GNSS or barometric (see altitude sensor parameter below)) (m) (0-4 094 m) 4 095 = not available, 4 094 = 4 094 m or higher
SOG	10	Speed over ground in knot steps (0-1 022 knots) 1 023 = not available, 1 022 = 1 022 knots or higher
Position accuracy	1	1 = high (> 10 m) 0 = low (< 10 m); 0 = default The PA flag should be determined in accordance with table.
Longitude	28	Longitude in 1/10 000 min (±180°, East = positive (as per 2's complement), West = negative (as per 2's complement); 181= ($6791AC0_h$) = not available = default)
Latitude	27	Latitude in 1/10 000 min (±90°, North = positive (as per 2's complement), South = negative (as per 2's complement); 91= (3412140_h) = not available = default)
COG	12	Course over ground in 1/10 = (0-3 599). 3 600 ($E10_h$) = not available = default; 3 601-4 095 should not be used
Time stamp	6	UTC second when the report was generated by the EPFS (0-59 or 60 if time stamp is not available, which should also be the default value or 61 if positioning system is in manual input mode or 62 if electronic position fixing system operates in estimated (dead reckoning) mode or 63 if the positioning system is inoperative)
Altitude sensor	1	0 = GNSS 1 = barometric source
Spare	7	Not used. Should be set to zero. Reserved for future use
DTE	1	Data terminal ready (0 = available 1 = not available = default) (see § 3.3.1)
Spare	3	Not used. Should be set to zero. Reserved for future use
Assigned mode flag	1	0 = Station operating in autonomous and continuous mode = default 1 = Station operating in assigned mode
RAIM-flag	1	RAIM flag of electronic position fixing device; 0 = RAIM not in use = default; 1 = RAIM in use see Table 47
Communication state selector flag	1	0 = SOTDMA communication state follows 1 = ITDMA communication state follows
Communication state	19	SOTDMA communication state (see § 3.3.7.2.1, Annex 2), if communication state selector flag is set to 0, or ITDMA communication state (see § 3.3.7.3.2, Annex 2), if communication state selector flag is set to 1
Number of bits	168	

Figura 1.27: messaggio 9 per i velivoli SAR.

Quando una stazione ha necessità di richiedere l'UTC ad un'altra stazione utilizza un semplice "messaggio di servizio", il numero 10.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 10; always 10
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI number of station which inquires UTC
Spare	2	Not used. Should be set to zero. Reserved for future use
Destination ID	30	MMSI number of station which is inquired
Spare	2	Not used. Should be set to zero. Reserved for future use
Number of bits	72	

Figura 1.28: struttura del messaggio 10.

Dei messaggi 11 e 13 già abbiamo dato un accenno in precedenza, essendo dei messaggi di *ack* che presentano la stessa struttura, rispettivamente, dei messaggi 4 e 7.

Concentriamoci quindi sul numero 12, ovvero un messaggio di *safety* di tipo indirizzato, che ha una lunghezza variabile a seconda della quantità di informazioni che contiene, avendo nella sua struttura un campo dedicato al testo libero destinato a contenere testo codificato sino ad un massimo di 936 bit .

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 12; always 12
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI number of station which is the source of the message.
Sequence number	2	0-3; see § 5.3.1, Annex 2
Destination ID	30	MMSI number of station which is the destination of the message
Retransmit flag	1	Retransmit flag should be set upon retransmission: 0 = no retransmission = default; 1 = retransmitted
Spare	1	Not used. Should be zero. Reserved for future use
Safety related text	Maximum 936	6-bit ASCII as defined in Table 44
Maximum number of bits	Maximum 1 008	Occupies 1 to 5 slots subject to the length of text For Class B mobile AIS stations the length of the message should not exceed 2 slots

Figura 1.29: struttura del messaggio 12.

Il messaggio 14 è l'omologo del 12, ma viene utilizzato per le trasmissioni di tipo *broadcast*.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 14; always 14.
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI number of source station of message
Spare	2	Not used. Should be set to zero. Reserved for future use
Safety related text	Maximum 968	6-bit ASCII as defined in Table 44
Maximum number of bits	Maximum 1 008	Occupies 1 to 5 slots subject to the length of text. For Class B mobile AIS stations the length of the message should not exceed 2 slots

Figura 1.30: struttura del messaggio 14.

Il messaggio 15 viene utilizzato per effettuare interrogazioni in TDMA (e non in DSC) sul data link VHF, la risposta a tela messaggio deve essere trasmessa sullo stesso canale sul quale è stata ricevuta.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 15; always set to 15
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI number of interrogating station
Spare	2	Not used. Should be set to zero. Reserved for future use
Destination ID1	30	MMSI number of first interrogated station
Message ID1.1	6	First requested message type from first interrogated station
Slot offset 1.1	12	Response slot offset for first requested message from first interrogated station
Spare	2	Not used. Should be set to zero. Reserved for future use
Message ID1.2	6	Second requested message type from first interrogated station
Slot offset 1.2	12	Response slot offset for second requested message from first interrogated station
Spare	2	Not used. Should be set to zero. Reserved for future use
Destination ID 2	30	MMSI number of second interrogated station
Message ID 2.1	6	Requested message type from second interrogated station
Slot offset 2.1	12	Response slot offset for requested message from second interrogated station
Spare	2	Not used. Should be set to zero. Reserved for future use
Number of bits	88-160	Total number of bits depends upon number of messages requested

Figura 1.31: struttura del messaggio 15.

Il messaggio 16 viene utilizzato per trasmettere i comandi di assegnazione degli slot da parte delle *base stations* ad eventuali entità controllate, in maniera da regolare e schedulare la trasmissione.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 16. Always 16
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI of assigning station
Spare	2	Spare. Should be set to zero. Reserved for future use
Destination ID A	30	MMSI number. Destination identifier A
Offset A	12	Offset from current slot to first assigned slot(1)
Increment A	10	Increment to next assigned slot(1)
Destination ID B	30	MMSI number. Destination identifier B. Should be omitted if there is assignment to station A, only
Offset B	12	Offset from current slot to first assigned slot. Should be omitted if there is assignment to station A, only(1)
Increment B	10	Increment to next assigned slot(1). Should be omitted, if there is assignment to station A, only
Spare	Maximum 4	Spare. Not used. Should be set to zero. The number of spare bits, which should be 0 or 4, should be adjusted in order to observe byte boundaries. Reserved for future use
Number of bits	96 or 144	Should be 96 or 144 bits

Figura 1.32: struttura del messaggio 16.

Nel caso una stazione risulti connessa ad una sorgente DGNSS, viene utilizzato un messaggio binario, il numero 17, allo scopo di apportare eventuali correzioni mediante tecniche differenziali.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 17; always 17
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source ID	30	MMSI of the base station
Spare	2	Spare. Should be set to zero. Reserved for future use
Longitude	18	Surveyed longitude of DGNSS reference station in 1/10 min (±180°, East = positive, West = negative). If interrogated and differential correction service not available, the longitude should be set to 181°
Latitude	17	Surveyed latitude of DGNSS reference station in 1/10 min (±90°, North = positive, South = negative). If interrogated and differential correction service not available, the latitude should be set to 91°
Spare	5	Not used. Should be set to zero. Reserved for future use
Data	0-736	Differential correction data (see below). If interrogated and differential correction service not available, the data field should remain empty (zero bits). This should be interpreted by the recipient as DGNSS data words set to zero
Number of bits	80-816	80 bits: assumes N = 0; 816 bits: assumes N = 29 (maximum value); see Table 66

Figura 1.33: struttura del messaggio 17.

I messaggi 18 e 19 sono dedicati al *position report* dei trasmettitori di classe B, hanno quindi un'analoga struttura dei *position report* già analizzati ad inizio paragrafo.

Merita un'attenzione particolare il messaggio 20, utilizzato da *base stations* vicine per preannunciare l'allocazione degli slots che effettueranno, secondo un piano di ripartizione FATDMA prestabilito. Grazie a questa tecnica di gestione del *data link* VHF è possibile garantire un elevato livello di efficienza, infatti mediante un'assegnazione fissa che riserva gli *slot* da utilizzare si azzerano le possibili collisioni in fase di invio, anche in regioni di spazio con molte *base stations* adiacenti.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 20; always 20
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Source station ID	30	MMSI number of base station
Spare	2	Not used. Should be set to zero. Reserved for future use
Offset number 1	12	Reserved offset number; 0 = not available(1)
Number of slots 1	4	Number of reserved consecutive slots: 1-15; 0 = not available(1)
Time-out 1	3	Time-out value in minutes; 0 = not available(1)
Increment 1	11	Increment to repeat reservation block 1; 0 = one reservation block per frame(1)
Offset number 2	12	Reserved offset number (optional)
Number of slots 2	4	Number of reserved consecutive slots: 1-15; optional
Time-out 2	3	Time-out value in minutes (optional)
Increment 2	11	Increment to repeat reservation block 2 (optional)
Offset number 3	12	Reserved offset number (optional)
Number of slots 3	4	Number of reserved consecutive slots: 1-15; optional
Time-out 3	3	Time-out value in minutes (optional)
Increment 3	11	Increment to repeat reservation block 3 (optional)
Offset number 4	12	Reserved offset number (optional)
Number of slots 4	4	Number of reserved consecutive slots: 1-15; optional
Time-out 4	3	Time-out value in minutes (optional)
Increment 4	11	Increment to repeat reservation block 4 (optional)
Spare	Maximum 6	Not used. Should be set to zero. The number of spare bits which may be 0, 2, 4 or 6 should be adjusted in order to observe byte boundaries. Reserved for future use
Number of bits	72-160	

Figura 1.34: struttura del messaggio 20.

Il messaggio 21 risulta molto importante come strumento per la diffusione delle informazioni relative agli ausili per la navigazione (AtoN), ovvero fari, boe et .

Un AtoN può essere reale o virtuale, la struttura del messaggio è la stessa e cambierà solo il valore relativo al campo *type of AtoN*, nel primo caso sarà esso stesso a trasmettere le informazioni, ciò presuppone che sia dotato di un proprio trasmettitore AIS. Nel caso di AtoN virtuale, invece, la trasmissione delle informazioni di ausilio è demandata a *base stations* nelle vicinanze in grado di coprire la zona in cui è di interesse far ricevere i dati relativi all'AtoN.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 21
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
ID	30	MMSI number, (see Article 19 of the RR and Recommendation ITU-R M.585)
Type of aids-to-navigation	5	0 = not available = default; refer to appropriate definition set up by IALA; see Table 71
Name of Aids-to-Navigation	120	Maximum 20 characters 6-bit ASCII, as defined in Table 44 "@@@@@@@@@@@@@@@@@@@@" = not available = default. The name of the AtoN may be extended by the parameter "Name of Aid-to-Navigation Extension" below
Position accuracy	1	1 = high (> 10 m) 0 = low (< 10 m) 0 = default. The PA flag should be determined in accordance with Table 47
Longitude	28	Longitude in 1/10 000 min of position of an AtoN (±180°, East = positive, West = negative 181 = ($6791AC0_h$) = not available = default)
Latitude	27	Latitude in 1/10 000 min of an AtoN (±90°, North = positive, South = negative 91 = (3412140_h) = not available = default)
Dimension/ reference for position	30	Reference point for reported position; also indicates the dimension of an AtoN (m) (see Fig. 42 and § 4.1), if relevant(1)
Type of electronic position fixing device	4	0 = Undefined (default) 1 = GPS 2 = GLONASS 3 = Combined GPS/GLONASS 4 = Loran-C 5 = Chayka 6 = Integrated Navigation System 7 = surveyed. For fixed AtoN and virtual AtoN, the charted position should be used. The accurate position enhances its function as a radar reference target 8 = Galileo 9-15 = not used
Time stamp	6	UTC second when the report was generated by the EPFS (0-59 or 60) if time stamp is not available, which should also be the default value or 61 if positioning system is in manual input mode or 62 if electronic position fixing system operates in estimated (dead reckoning) mode or 63 if the positioning system is inoperative)
Off-position indicator	1	For floating AtoN, only: 0 = on position; 1 = off position. NOTE 1 – This flag should only be considered valid by receiving station, if the AtoN is a floating aid, and if time stamp is equal to or below 59. For floating AtoN the guard zone parameters should be set on installation
AtoN status	8	Reserved for the indication of the AtoN status 00000000 = default
RAIM-flag	1	RAIM (Receiver autonomous integrity monitoring) flag of electronic position fixing device; 0 = RAIM not in use = default; 1 = RAIM in use see Table 47
Virtual AtoN flag	1	0 = default = real AtoN at indicated position; 1 = virtual AtoN, does not physically exist(2).
Assigned mode flag	1	0 = Station operating in autonomous and continuous mode = default 1 = Station operating in assigned mode
Spare	1	Spare. Not used. Should be set to zero. Reserved for future use
Name of Aid-to-Navigation Extension	0, 6, 12, 18, 24, 30, 36, ... 84	This parameter of up to 14 additional 6-bit-ASCII characters for a 2-slot message may be combined with the parameter "Name of Aid-to-Navigation" at the end of that parameter, when more than 20 characters are needed for the name of the AtoN. This parameter should be omitted when no more than 20 characters for the name of the A-to-N are needed in total. Only the required number of characters should be transmitted, i.e. no @-character should be used
Spare	0, 2, 4, or 6	Spare. Used only when parameter "Name of Aid-to-Navigation Extension" is used. Should be set to zero. The number of spare bits should be adjusted in order to observe byte boundaries
Number of bits	272-360	Occupies two slots

Figura 1.35: struttura del messaggio 21 *type of AtoN*.

La codifica per il campo *type of AtoN*, che identifica il tipo di AtoN associato alle informazioni in trasmissione meditante il messaggio 21, segue la seguente tabella.

	Code	Definition
	0	Default, Type of AtoN not specified
	1	Reference point
	2	RACON
	3	Fixed structures off-shore, such as oil platforms, wind farms. (NOTE 1 – This code should identify an obstruction that is fitted with an AtoN AIS station)
	4	Spare, Reserved for future use
Fixed AtoN	5	Light, without sectors
	6	Light, with sectors
	7	Leading Light Front
	8	Leading Light Rear
	9	Beacon, Cardinal N
	10	Beacon, Cardinal E
	11	Beacon, Cardinal S
	12	Beacon, Cardinal W
	13	Beacon, Port hand
	14	Beacon, Starboard hand
	15	Beacon, Preferred Channel port hand
	16	Beacon, Preferred Channel starboard hand
	17	Beacon, Isolated danger
	18	Beacon, Safe water
	19	Beacon, Special mark
Floating AtoN	20	Cardinal Mark N
	21	Cardinal Mark E
	22	Cardinal Mark S
	23	Cardinal Mark W
	24	Port hand Mark
	25	Starboard hand Mark
	26	Preferred Channel Port hand
	27	Preferred Channel Starboard hand
	28	Isolated danger
	29	Safe Water
	30	Special Mark
	31	Light Vessel/LANBY/Rigs

Figura 1.36: valori del campo *type of AtoN*.

L’ultimo messaggio analizzato è il messaggio numero 22, la cui struttura è molto semplice e con un impegno di solo 168 bit per il *payload*, quindi per la sua trasmissione risulta sufficiente un unico *timeslot*. Tale messaggio viene utilizzato per la trasmissione in *broadcast* di comandi per la gestione del canale di comunicazione, secondo un’area geografica prestabilita per ogni singola *base station*.

Parameter	Number of bits	Description
Message ID	6	Identifier for Message 22; always 22
Repeat indicator	2	Used by the repeater to indicate how many times a message has been repeated. See § 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more
Station ID	30	MMSI number of Base station
Spare	2	Not used. Should be set to zero. Reserved for future use
Channel A	12	Channel number according to Recommendation ITU-R M.1084, Annex 4
Channel B	12	Channel number according to Recommendation ITU-R M.1084, Annex 4
Tx/Rx mode	4	0 = Tx A/Tx B, Rx A/Rx B (default) 1 = Tx A, Rx A/Rx B 2 = Tx B, Rx A/Rx B 3-15: not used When the dual channel transmission is suspended by Tx/Rx mode command 1 or 2, the required reporting interval should be maintained using the remaining transmission channel
Power	1	0 = high (default), 1 = low
Longitude 1, (or 18 most signifi-cant bits (MSBs) of addressed station ID 1)	18	Longitude of area to which the assignment applies; upper right corner (North-East); in 1/10 min, or 18 MSBs of addressed station ID 1 (±180°, East = positive, West = negative) 181 = not available
Latitude 1, (or 12 least signifi-cant bits (LSBs) of addressed station ID 1)	17	Latitude of area to which the assignment applies; upper right corner (North-East); in 1/10 min, or 12 LSBs of addressed station ID 1, followed by 5 zero bits (±90°, North = positive, South = negative) 91° = not available
Longitude 2, (or 18 MSBs of addressed station ID 2)	18	Longitude of area to which the assignment applies; lower left corner (South-West); in 1/10 min, or 18 MSBs of addressed station ID 2 (±180°, East = positive, West = negative)
Latitude 2, (or 12 LSBs of addressed station ID 2)	17	Latitude of area to which the assignment applies; lower left corner (South-West); in 1/10 min, or 12 LSBs of addressed station ID 2, followed by 5 zero bits (±90°, North = positive, South = negative)
Addressed or broadcast message indicator	1	0 = broadcast geographical area message = default; 1 = addressed message (to individual station(s))
Channel A bandwidth	1	0 = default (as specified by channel number); 1 = spare (formerly 12.5 kHz bandwidth in Recommendation ITU-R M.1371-1)
Channel B bandwidth	1	0 = default (as specified by channel number); 1 = spare (formerly 12.5 kHz bandwidth in Recommendation ITU-R M.1371-1)
Transitional zone size	3	The transitional zone size in nautical miles should be calculated by adding 1 to this parameter value. The default parameter value should be 4, which translates to 5 nautical miles; see § 4.1.5, Annex 2
Spare	23	Not used. Should be set to zero. Reserved for future use
Number of bits	168	

Figura 1.37: struttura del messaggio numero 22.

1.5 Standard NMEA.

Una volta costruito i messaggi AIS da trasmettere, rispettando la suddivisione dei bit così come indicata nelle strutture messaggi riportate nel paragrafo precedente e codificati secondo la tabella ASCII proposta dall'ITU, occorre incapsulare il messaggio AIS secondo uno standard comune alle stazioni riceventi e trasmittenti. La scelta è ricaduta sulla codifica NMEA, che opera tramite un semplice algoritmo che sfrutta una tabella ASCII a 6 bit.

NMEA 0183 (o più comunemente NMEA) è uno standard di comunicazione di dati utilizzato soprattutto in nautica e nella comunicazione di dati satellitari GPS. L'ente che gestisce e sviluppa il protocollo è la *National Marine Electronics Association*. Questo protocollo si basa sul principio che la fonte, detta *talker*, può soltanto inviare i dati e la ricevente, detta *listener*, può soltanto riceverli.

I dati da trasmettere vengono sintetizzanti in *sentences*, frasi che presentano una struttura del tipo:

$PREFISSO,dato1,dato2 ... datoN, *CHECKSUM

La frase inizia sempre con $ e termina sempre con CR LF (*Carriage Return* e *Line Feed*) ed, al massimo, risulta lunga 80 caratteri.

Il prefisso è la prima parte della stringa, che serve a specificare di che tipo è il *talker*, ad esempio, autopilota, dispositivo GPS, controllo velocità, controllo direzione, ecc.

Nel caso dell'AIS, i dati codificati per viaggiare nell'etere secondo lo standard ITU (figura n. 1.38), vengono poi incapsulati in stringhe NMEA.

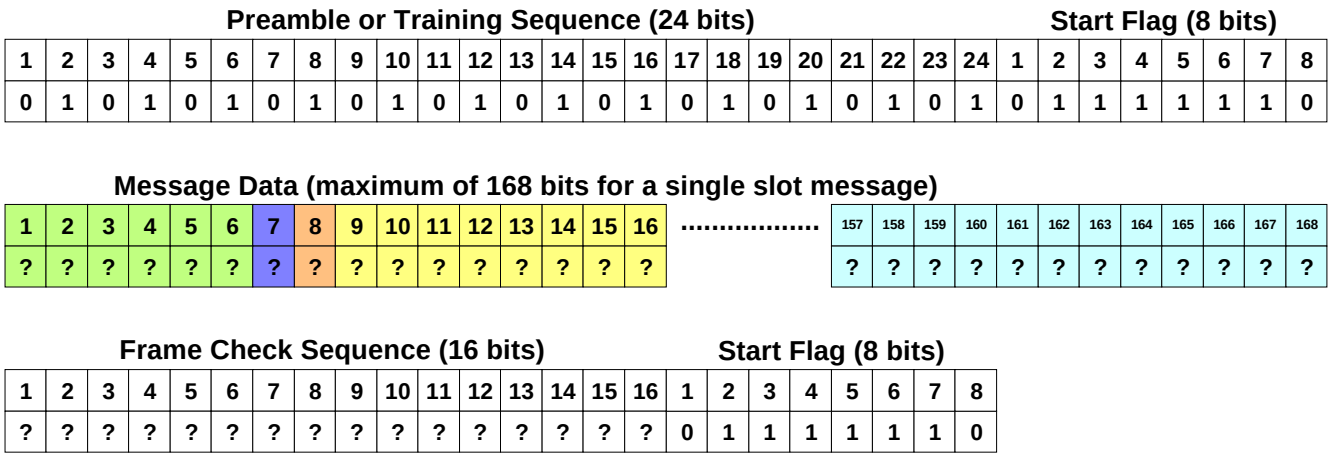


Figura 1.38: pacchetto dati AIS.

La stringa NMEA, che ha un prefisso AI seguito dal tipo della frase, identificato con 3 lettere (es: VDM,VDO et c.), risulta essere una sequenza di caratteri ricavati dalla seguente tabella di *lookup*.

Carattere	**Campo in binario**	**Carattere**	**Campo in binario**
0	000000	P	100000
1	000001	Q	100001
2	000010	R	100010
3	000011	S	100011
4	000100	T	100100
5	000101	U	100101
6	000110	V	100110
7	000111	W	100111
8	001000	'	101000
9	001001	a	101001
:	001010	b	101010
;	001011	c	101011
<	001100	d	101100
=	001101	e	101101
>	001110	f	101110
?	001111	g	101111
@	010000	h	110000
A	010001	i	110001
B	010010	j	110010
C	010011	k	110011
D	010100	l	110100
E	010101	m	110101
F	010110	n	110110
G	010111	o	110111
H	011000	p	111000
I	011001	q	111001
J	011010	r	111010
K	011011	s	111011
L	011100	t	111100
M	011101	u	111101
N	011110	v	111110
O	011111	w	111111

La costruzione della stringa NMEA, sulla base della succitata tabella di lookup, può essere sintetizzato secondo l’algoritmo così come riportato in figura 1.39.

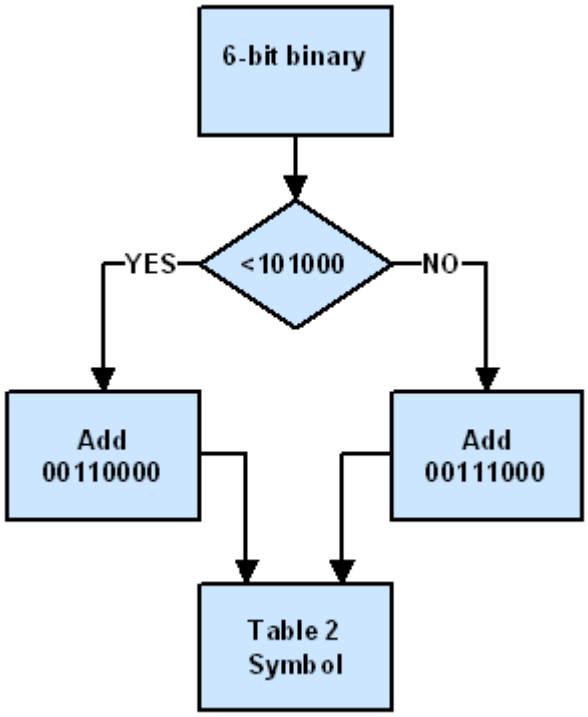


Figura 1.39: algoritmo di costruzione della stringa NMEA..

Viceversa la decodifica di un carattere NMEA in una stringa di bit ASCII a 6 bit segue la sequenza iterativa indicata nella figura 1.40.

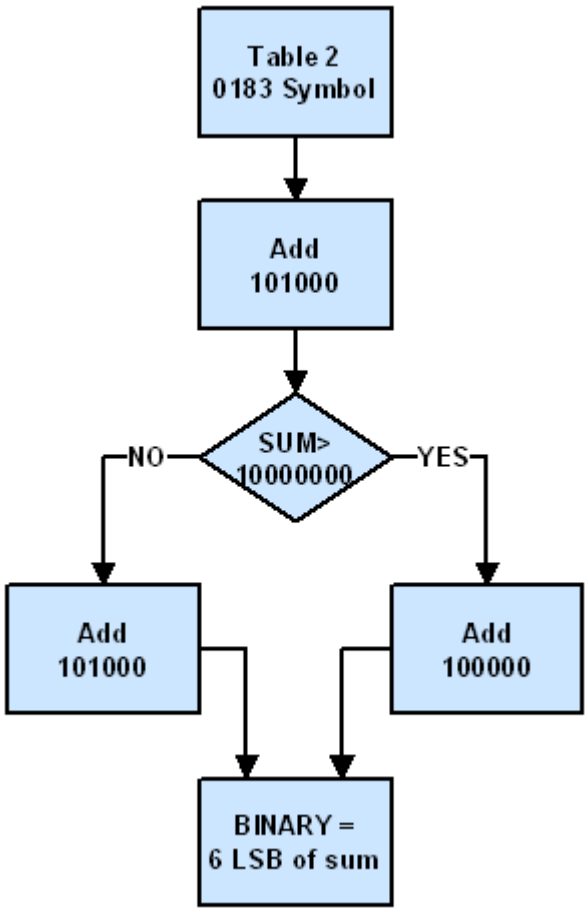


Figura 1.40: algoritmo di decodifica di una stringa NMEA.

Ogni frase presenta un prefisso, un carattere di inizio stringa ed un carattere di terminazione; la *checksum* viene calcolato escludendo il carattere di inizio stringa ed il carattere finale (l'asterisco "*"), utilizzando come algoritmo la ***OR esclusiva*** e componendo il risultato in 2 lettere o numeri.

Una tipica stringa NMEA per codificare un messaggio AIS ha la seguente forma:

!AIVDM,1,1,,A,14eG;o@034o8sd<L9i:a;WF>062D,0*7D

```
!AIVDM:         identifica il tipo di messaggio
1               numero di stringhe
1               numero della stringa (nel caso di più stringhe)
A               Il canale AIS da utilizzare (A o B)
14eG;...        i dati AIS codificati
0*              termine della stringa
7D              NMEA Checksum
```

CAPITOLO 2

L'acquisizione dei dati AIS si basa sulla trasmissione continua e reciproca tra le navi (*ship to ship*) e fra le navi e le stazioni di base a terra (*ship to shore*), tramite due canali VHF dedicati. In questo secondo capitolo sarà dedicata attenzione alla modalità "*ship to shore*" illustrando come costituire una rete AIS costiera, prendendo in esame la rete nazionale italiana dedicata alla ricezione ed elaborazione dei dati AIS, realizzata seguendo le indicazioni dell'Organizzazione Marittima Internazionale, che, in forza della ITU-R M.1371-1, ha formalizzato uno standard tecnico per l'AIS. Grazie alle informazioni scambiate nell'area di copertura della rete AIS è possibile ricostruire la situazione dell'area interessata.

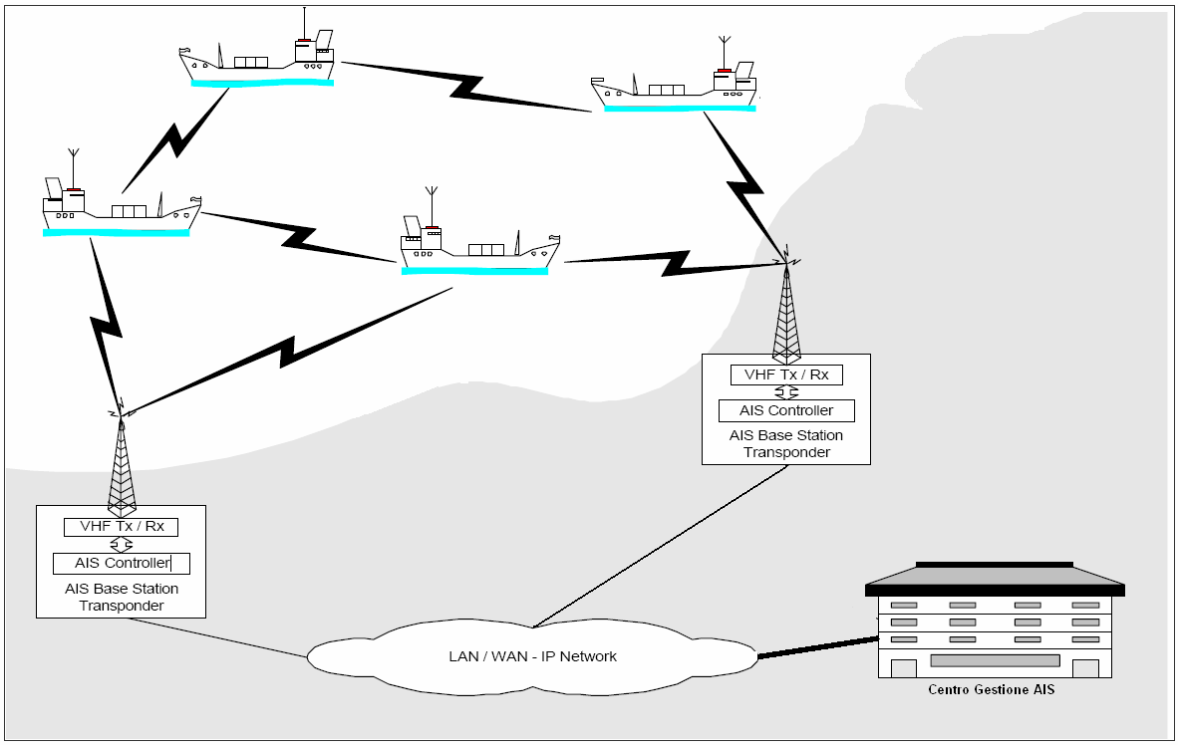


Figura 2.1: schema di scambio dati AIS tra navi e con stazioni AIS costiere.

Nella figura è illustrato quanto si andrà di seguito ad analizzare in dettaglio, ovvero come una rete di *base stations* AIS, opportunamente installate lungo le coste, possano ricevere e distribuire informazioni, tramite il coordinamento di un centro di gestione.

2.1 Architettura di una rete AIS

Il sistema AIS mette a disposizione un servizio di informazione per il monitoraggio e gli schemi di gestione del traffico, fornisce un sistema di *reporting* delle navi e altri servizi di sicurezza costieri, processando le informazioni provenienti dai vari sensori (*base stations* costiere) in modo da presentare all'utente l'immagine del traffico.

Il servizio AIS di una autorità competente comprende una serie di requisiti che possono essere definite in termini funzionali come un set di *Basic AIS Services* (BAS) che incapsulino il dettaglio tecnico sia della tecnologia AIS che del layout e della configurazione locale degli AIS costieri.

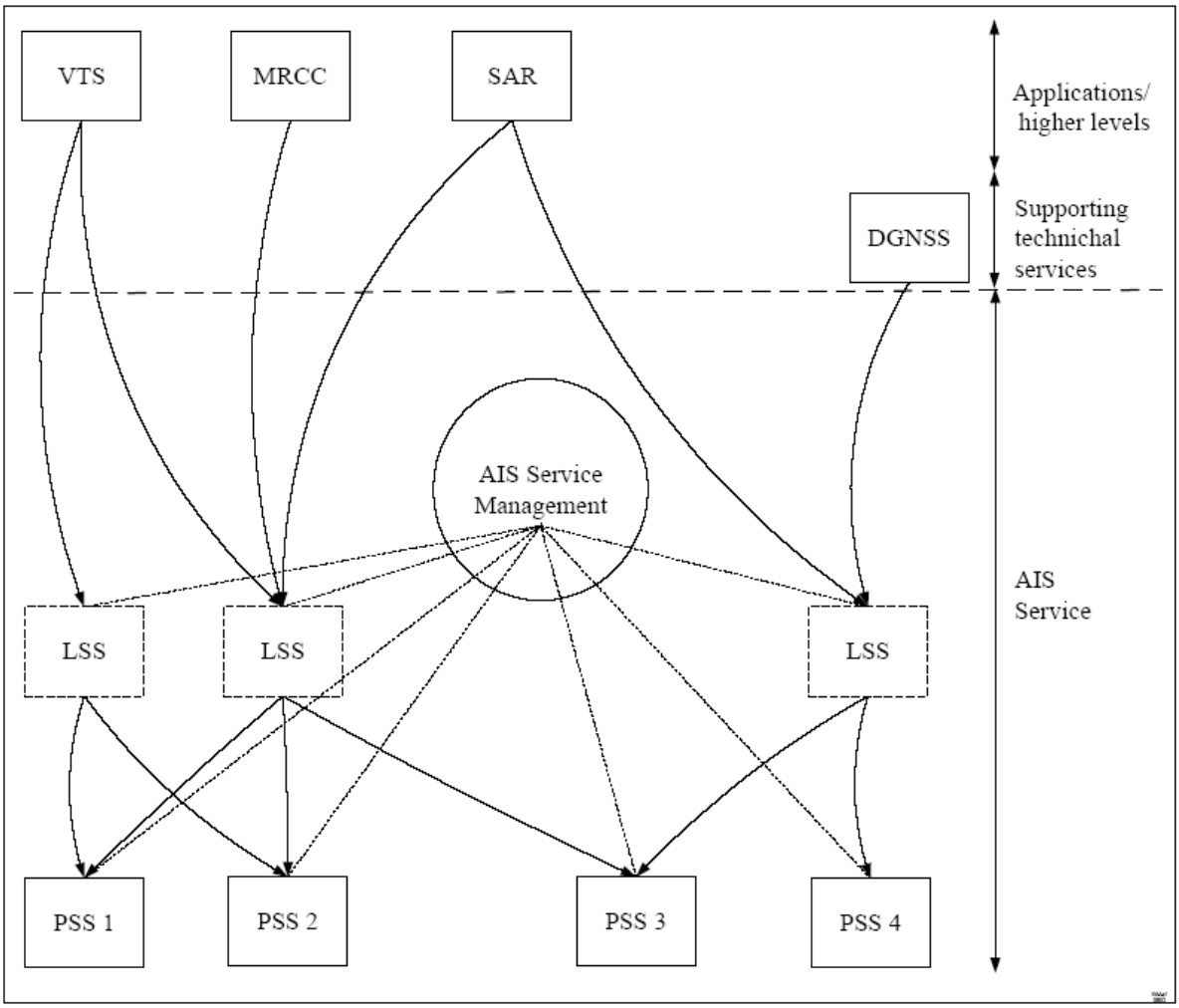


Figura 2.2: esempio di un AIS Service di terra, con clients e providers di dati.

In un sistema AIS possono essere identificati almeno cinque strati funzionali, con differenti obiettivi:

- Il livello *Logical Shore Station* (LSS) elabora i dati derivati dalle differenti stazioni costiere fisiche AIS e fornisce i BAS alle applicazioni durante la sua esecuzione.

- Il livello stazione costiera AIS: una stazione costiera AIS può ospitare una o più stazioni AIS fisse.

- Il livello stazione fissa AIS (stazioni ripetitrici AIS sono localizzate anche a questo livello).

- Il livello dell'apparecchiatura del dominio VHF/RF, la quale comprende l'antenna (o le antenne) e gli altri componenti dell'infrastruttura.

- Il livello AIS *VHF Data Link* (VDL) al quale si può accedere solo attraverso l'apparecchiatura VHF/RF.

Ognuno di questi livelli, di solito, consiste di differenti entità le quali devono espletare la completa funzionalità del livello.

In aggiunta è richiesto un livello superiore il quale ha la responsabilità di gestire l'intero servizio AIS. Questo livello logico più alto del servizio AIS è denominato *AIS Service Management* (ASM).

2.1.1 *Physical AIS Shore Station* (PSS).

Una PSS è basata su di una apparecchiatura transponder con le seguenti caratteristiche:

- Range di frequenza: 156.025 – 162.025 MHz
- Larghezza di banda del canale: 12.5/25 kHz
- Separazione del canale: 12.5 kHz
- Potenza di uscita: 2/12.5 W nominale (selezionabile)
- Modulazione: GMSK (AIS TDMA)/GFSK (DSC)

- Velocità di trasferimento dati: 9600 bps
- Ricevitore GNSS

La ridondanza è implementata per mezzo di due transponder autonomi gestiti da un avanzato controller in grado di passare da una unità all'altra in caso di malfunzionamento.

I transponder saranno posizionati in specifiche posizioni ed altezze in maniera tale da consentire la completa copertura delle acque territoriali.

2.1.2 *Logical AIS Shore Station* (LSS).

Il livello della *Logical AIS Shore Station* è il livello immediatamente superiore alla PSS il quale pre-processa i dati da e verso le stazioni fisse AIS.

Il livello superiore di questo stack fornisce una funzionalità molto importante per le applicazioni: trasforma il servizio AIS collegato alla stazione costiera AIS fisica in un servizio AIS collegato ad una stazione AIS logica. I benefici di questa trasformazione per le applicazioni che utilizzano i servizi AIS sono i seguenti:

- Ogni stazione AIS logica può essere associata ad un'area geografica, la quale è completamente definita da considerazioni operative in contrapposizione ad una qualsiasi stazione AIS fisica, la cui area di copertura è determinata esclusivamente dalla propagazione del segnale radio. In aggiunta, l'area geografica di una stazione AIS può avere qualsiasi forma a differenza dell'area di copertura di una stazione AIS fisica, la cui area è determinata dalla pianificazione dell'antenna e dalla propagazione radio. Il concetto di stazione costiera AIS logica fornisce una interfaccia tra la progettazione operativa e fisica di una infrastruttura AIS.
- Mentre i precedenti benefici sono validi già quando viene presa in considerazione una sola tipologia di applicazione AIS, i benefici aumentano quando vengono presi in considerazione diverse applicazioni AIS differenti. Ognuna di queste applicazioni operative può definire una o più stazioni AIS logiche le quali dipendono esclusivamente dalle loro necessità operative. Tutte queste stazioni AIS logiche saranno processi di trasformazioni i quali acquisiscono i dati rilevanti derivati da AIS da una o più stazioni fisiche AIS. I benefici per le differenti applicazioni derivano dal fatto che si possono definire insiemi di disposizioni operative per le aree geografiche associate le quali possono essere parzialmente o completamente sovrapposte. Quindi il concetto di stazione

AIS logica consente a differenti applicazioni di accedere alle stesse risorse attraverso una interfaccia standardizzata.

- Il concetto di stazione AIS logica delocalizza l'interfaccia tra il servizio AIS e le applicazioni. Di conseguenza non esistono ragioni tecniche per cui le applicazioni che utilizzano il servizio AIS debbano risiedere nelle vicinanze della stazione fisica AIS (potrebbe, tuttavia, sussistere delle motivazioni operative).

2.1.3 *AIS Service Management* (ASM).

Un servizio AIS può includere un numero diverso di Stazioni AIS fisiche (PSS) e di stazioni AIS logiche (LSS). Al fine di operare da remoto e di configurare le PSS e le LSS, viene introdotto un livello più alto. Questo livello più alto è denominato *AIS Service Management* (ASM). L'ASM agisce come un ente di controllo per l'intero servizio AIS.

Quindi un LSS è una rappresentazione logica di uno o più PSS, che offre i servizi di base AIS (BAS) ai vari client, ovvero gli utilizzatori delle informazioni AIS acquisite mediante le stazioni fisiche.

L'ASM "possiede" tutte le stazioni logiche e fisiche, per esempio:

- ASM invoca, inizializza, configura e termina i processi software delle stazioni fisiche e logiche.
- ASM determina le relazioni di comunicazione, tra le stazioni fisiche e le loro associate stazioni logiche, per il loro utilizzo.
- ASM determina le relazioni di comunicazione tra la stazioni logiche e le applicazioni a loro associate.

2.1.4 Linee guida per l'AIS.

Le linee guida del servizio AIS in Italia sono attualmente contenute in un provvedimento adottato di recente dal Comando Generale del corpo delle Capitanerie di porto che, in aderenza alla direttiva 2002/59/CE del 27 Giugno 2002, ha realizzato, per la ricezione delle informazioni AIS, una rete che allo stato attuale conta 45 stazioni di base (*Ground Base Station*), ubicate in siti di competenza degli uffici periferici del Corpo.

Il sistema AIS è di tipo aperto, ciò vuol dire che i dati trasmessi sono acquisibili da chiunque sia dotato di apparecchi ricevitori idonei, quindi vi è un rischio di un utilizzo

improprio da parte di soggetti (ad es. identificazione di una nave per finalità terroristiche). In questa prospettiva, un parziale accorgimento è stato preso obbligando l'interruzione della trasmissione dei dati da parte delle navi quando si verificano situazioni che lasciano trasparire rischi per la sicurezza e quando queste sono ferme in porto in modo da proteggere la sicurezza delle infrastrutture portuali.

Occorre puntualizzare che la presenza del transponder AIS a bordo non è obbligatorio per tutte le navi, ai sensi della SOLAS esso è obbligatorio solo per navi al di sopra di una certa stazza lorda[10].

Ovviamente l'efficienza del sistema è condizionata alla generalizzazione dell'installazione degli apparati in questione su tutte le navi presenti nell'area interessata, che siano in navigazione o alla fonda, nonché sul mantenimento degli stessi in stato di efficienza e funzionamento: un'unità che non sia dotata o non mantenga in stato di efficienza il sistema AIS a bordo, comporta una lacuna nell'assunzione dei dati utili alla valutazione della situazione di traffico e dei fattori connessi alla sicurezza della navigazione nell'area.

2.2 La rete AIS in Italia.

In Italia, come per le altre nazioni costiere dotati di infrastruttura AIS, l'*Automatic Identification System* prevede lo scambio di informazioni tra navi e stazioni di terra attraverso l'utilizzo di un canale dati radio condiviso, operando in banda VHF sui canali AIS 1 161.975 MHz e AIS 2 162.025 MHz.

L'architettura della rete italiana si basa quindi, come da modello appena descritto, su una serie di stazioni costiere PSS, che assicurano l'acquisizione dei dati continuamente trasmessi dalle navi dotate di apposito transponder AIS.

La copertura radio di ogni singola stazione AIS è stata progettata, utilizzando consolidati modelli matematici, in modo tale da garantire il *radio link* in qualsiasi condizioni di propagazione troposferica.

[10] La stazza rappresenta la misura con cui si valuta il volume di una nave. La stazza lorda comprende tutti i volumi interni della nave, compresi gli spazi della sala macchine, dei serbatoi di carburante, le zone riservate all'equipaggio. Si misura partendo dalla superficie esterna delle paratie

Come detto, ogni PSS viene controllata da una stazione logica periferica LSS, essenzialmente costituita da un server dotato di un processo software che ha il compito di elaborare i dati raccolti dalle PSS fungendo da interfaccia da e verso le applicazioni.

Ogni LSS, che può controllare uno o più unità fisiche, oltre alla raccolta, elaborazione e memorizzazione dei dati AIS, si occupa anche delle attività di *networking*, connessione e trasferimento dati verso il server centrale, un "Server Nazionale AIS" con appropriate dotazioni hardware e software che si trova presso il Comando Generale del Corpo delle Capitanerie di Porto, l'ente preposto alla sorveglianza del traffico marittimo e che, quindi, riveste il ruolo di *AIS Service Management*.

2.3 *AIS Ground Base Station.*

Ogni singola stazione base AIS costiera (*AIS ground base station*) è strutturata e può essere funzionalmente suddivisa in tre segmenti (con riferimento alla figura proposta di seguito):

- Sezione remota, che comprende, seguendo la nomenclatura degli standard internazionali, la parte fisica della stazione AIS costiera [PSS];

- Sezione di gestione e controllo con server dati, che comprende la parte logica della stazione AIS costiera [LSS];

- Sezione di comunicazione tra le due sezioni precedenti (Data Link in Ponte Radio)

Nella figura 2.3 è schematizzata la struttura di un'intera stazione di terra AIS suddivisa in due parti. Da un lato la sezione dedicata alla trasmissione/ricezione vera e propria, installata presso siti elevati che garantiscano un'adeguata copertura VHF. Dall'altro lato troviamo il server dedicato alla raccolta, memorizzazione e visualizzazione dei dati, che gli pervengono dal sito remoto mediante collegamento in ponte radio a 900 Mhz.

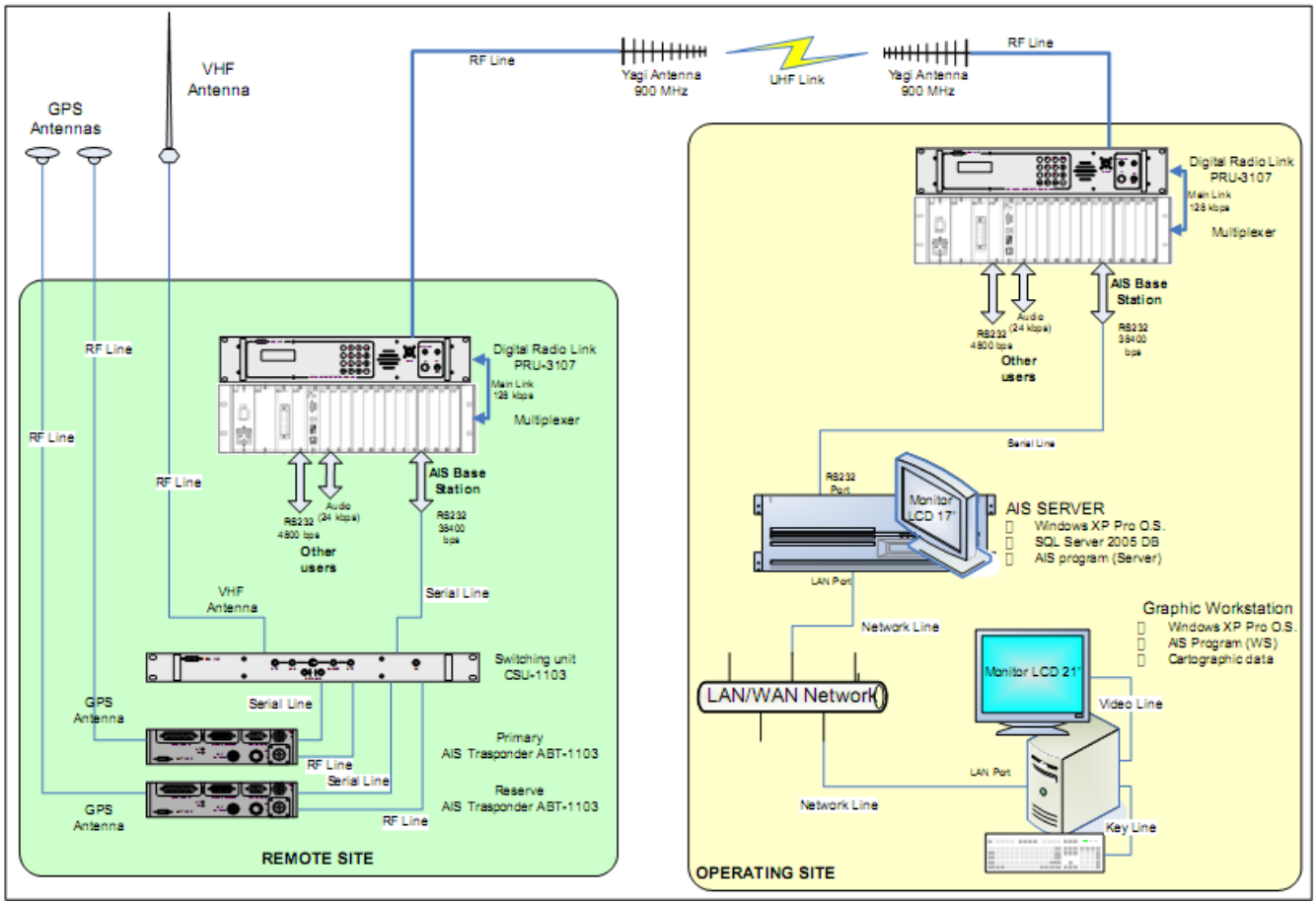


Figura 2.3 stazione di terra AIS.

La sezione remota comprende i seguenti dispositivi:

- Transponder AIS di terra (primario) mod. Elman ABT-1103
- Transponder AIS di terra (riserva) mod. Elman ABT-1103
- Antenne GPS (n. 2)
- Antenna VHF
- Unità di commutazione primario-riserva mod. Elman CSU-1103
- Multiplexer e ponte radio UHF (mod. Elman PRU-3107) per la connessione verso il sito di controllo.

La sezione remota è generalmente installata presso siti montani (o comunque ad elevate altitudini) per ottenere una copertura a radiofrequenza che sia la più ampia possibile. Naturalmente il sito remoto, oltre a presentare caratteristica di altitudine

elevata, deve essere in visibilità radio con il rispettivo sito di gestione e controllo, in modo da potergli trasmettere i dati ricevuti.

La sezione di gestione e controllo è generalmente installata presso gli uffici periferici degli enti responsabili delle attività di sorveglianza del traffico marittimo, essa comprende i seguenti dispositivi:

- Computer server con la funzione di:
  1. acquisizione e memorizzazione dei dati AIS;
  2. gestione amministrativa della *AIS Base Station* (Controllo dei transponder AIS, eventuale loro upgrade, controllo dei ponti radio e dei multiplexer, ecc.);
  3. connessione con eventuale rete LAN/WAN; tale connessione permette una molteplicità di funzioni, prima fra tutte la gestione centralizzata del sistema AIS da un unico server nazionale e la condivisione delle informazioni con altri utenti connessi in rete (es. VTS).

- Computer Workstation cartografica, con la funzione di rappresentazione delle informazioni AIS a favore degli operatori al servizio di sorveglianza marittima.

- Multiplexer e ponte radio UHF per la connessione verso il sito remoto.

I componenti delle sezioni remote e di controllo possono essere rivisti attraverso una diversa visione, ovvero una suddivisione logica che si ottiene raggruppando i singoli elementi nei "contenitori logici" *Phisical Shore Station* e *Logical Shore Station,* così come riportato nella figura 2.4.

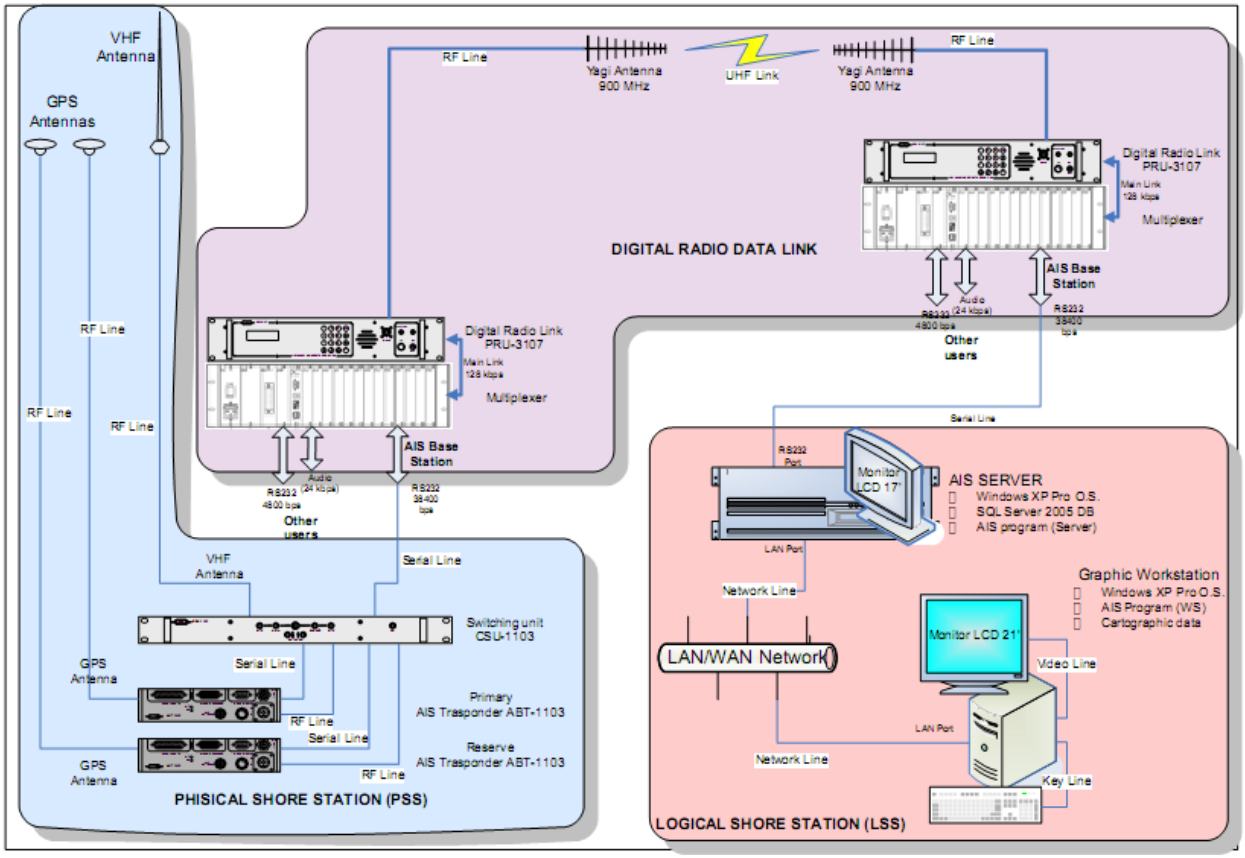


Figura 2.4 suddivisione logica per una stazione AIS.

L'insieme delle PSSs e LSSs vanno a costituire quindi la rete AIS, nel caso italiano la stazione logica viene realizzata mediante dei server periferici che fanno capo ad un server centrale, ubicato presso il Comando Generale della Guardia Costiera, che svolge la funzione di ASM.

Mentre il collegamento tra PSS e LSS viene effettuato mediante trasmissioni in ponte radio, le singole LSS inviato i dati all'ASM mediante una WAN della pubblica amministrazione (nello specifico la SPC, "servizio di pubblica connettività"), che connette tutti gli uffici periferici del Corpo delle Capitanerie di Porto.

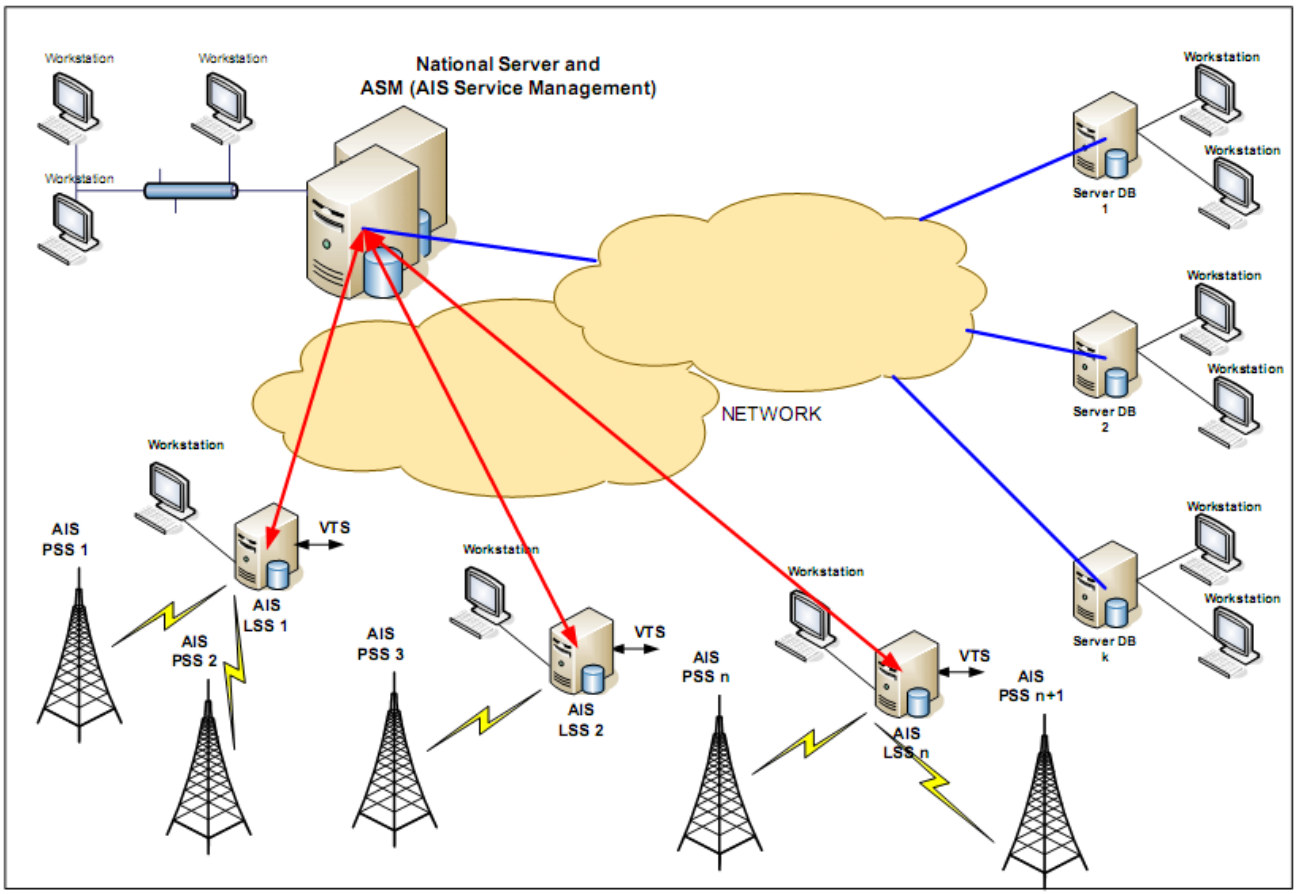


Figura 2.5: rete di PSS e LSS coordinate dall'ASM.

Gli obiettivi principali del server AIS nazionale (e in parte dei server AIS periferici) sono di:

- raccogliere,
- visualizzare,
- memorizzare

i dati AIS relativi alle stazioni costiere connesse. Quindi la struttura generale può essere schematizzata come segue:

Raccolta di dati

- Raccolta dei dati e filtraggio
- Distribuzione in tempo reale dei dati AIS (verso altri utilizzatori)

Memorizzazione

- Memorizzazione dei dati in database

Recupero e analisi

- Lettura dei dati memorizzati
- Analisi e presentazione dei dati memorizzati in rapporti predefiniti, presentazioni grafiche e correlate con informazioni cartografiche e di posizionamento (GIS)
- Accesso ai dati attraverso browser web
- Autenticazione e accesso per gli utenti autorizzati ad usufruire dei dati AIS.

2.3.1 Raccolta dati.

I dati in ingresso (flusso dati AIS) dalle singole stazioni costiere sono raccolti continuamente online attraverso connessioni di rete instaurate mediante protocollo TCP-IP.

I *servers* AIS (nazionale o periferici) ricevono i flussi di dati, rimuove i dati duplicati ed è in grado di distribuire i dati AIS globali in tempo reale ad ogni utente connesso.

Come illustrato nel precedente capitolo il formato per lo scambio dei dati è stabilito nella IEC 61993-2 e si basa sull'incapsulamento in stringhe NMEA dei messaggi radio specificati dalla ITU-R M.1371.

L'eventuale filtraggio dei dati è effettuato secondo le esigenze operative e solo gli utenti autenticati (*username* e *password*) possono avere accesso ai dati visualizzabili su apposita workstation.

.

2.3.2 Memorizzazione dei dati.

La memorizzazione dei dati AIS ricevuti è effettuata ad intervalli prestabiliti e per un tempo indefinito dipendente esclusivamente dalle risorse hardware (spazio di memoria disponibile) prima di procedere ad un attività di backup, effettuata periodicamente.

Il DBMS utilizzato nei server periferici è ***SQL Server 2005 Express Edition***, una versione gratuita, facile da usare, leggera e dotata di potenti funzionalità, come *l'SQL Server Management Studio Express*, che permette una completa gestione del database. Unica limitazione consiste nella quantità di record memorizzabili, che non possono superare una capienza pari a 4 Gb di informazioni.

Per il server nazionale centrale, che deve assicurare quanti meno limiti possibili, si è invece utilizzata la versione professionale del medesimo DBMS appena descritto.

2.3.3 Visualizzazione e analisi dei dati.

La visualizzazione dei dati AIS è possibile in tempo reale, ma anche a posteriori attraverso interrogazione del database e visualizzazione in funzione replay. Tale operazione è comunque effettuabile solo dall'amministratore del sistema che, su specifiche richieste da parte degli operatori, può generare rapporti tabellari specifici dei dati AIS memorizzati mediante apposite query di selezione sul database.

I dati AIS memorizzati ed analizzati possono essere mostrati graficamente in diagrammi e sinottici secondo diverse stratificazioni statistiche (filtri temporali, tipologia di carico, aree geografiche determinate, ecc.) ed è possibile mostrarli in contesti geografici (correlati con i dati di posizione) all'interno di sistemi GIS.

2.4 Protocollo di comunicazione tra server AIS e sistemi esterni.

Il transponder AIS, come tutti i dispositivi e sistemi marittimi di navigazione e radiocomunicazione, dispone di interfacce di comunicazione digitali rispondenti alla normativa IEC-61162 (-1, -2, ecc.), tale normativa definisce in modo completo la tipologia di interfaccia digitale da utilizzare (es. porta seriale protocolli RS232 o RS422 con velocità di 4800 b/sec o 38400 b/sec) e il formato di scambio delle informazioni.

I server AIS (sia il server nazionale che i periferici) utilizzano in pieno le modalità di interfaccia standard definite dalla normativa citata e rendono la connessione (informazioni sia in input che in output) disponibile attraverso rete LAN – WAN (rete fast-ethernet 10/100 TX o secondo le disponibilità della SPC), incapsulando l'informazione all'interno di un pacchetto TCP/IP (o un datagramma UDP se in rete locale ad alta affidabilità) senza alcuna trasformazione o manipolazione delle stringhe AIS. Tale funzione è ottenuta da un processo software denominato "HELCOM Server", che si basa su un protocollo di connessione denominato "Helcom protocol".

Il protocollo di connessione "HELCOM" prevede la possibilità di stabilire una connessione di tipo TCP/IP con autenticazione. I parametri di connessione sono da definire unitamente agli amministratori di sistema, e sono i seguenti:

- Indirizzo IP (IP Address)
- Numero della porta di connessione TCP (*TCP port*)

- Intervallo di invio dell'informazione (*Send Interval*, espresso in secondi)
- Identificativo dell'utente esterno (*User Name*)
- Password

Il protocollo prevede due fasi distinte:

- Instaurazione della connessione TCP/IP.
- Invio dei parametri di autenticazione (User name e password) secondo un formato prefissato.

Ad autenticazione favorevole sussegue l'inoltro dell'informazione AIS (*AIS raw data*) incapsulata verso l'utente connesso con la periodicità programmata e secondo diverse stratificazioni statistiche (filtri temporali, tipologia di carico, aree geografiche determinate, ecc.).

CAPITOLO 3

Nei precedenti capitoli si è illustrato il funzionamento di transponder AIS, che basa il suo funzionamento sulla trasmissione in VHF. Si è poi analizzata l'infrastruttura di una rete AIS costiera nazionale per riceve i dati AIS trasmessi dalle unità navali, al fine di realizzare un sistema di monitoraggio che consenta di visualizzare la situazione del traffico navale.

Partendo da tali presupposti, costruiti mediante l'utilizzo di tecnologie semplici e consolidate (GPS, VHF, rete WAN), è possibile pensare alla progettazione di una rete di gerarchia superiore che raggruppi le reti AIS nazionali degli Stati del Mediterraneo.

Tralasciando di dettagliare la realizzazione degli applicativi software, risulta di maggiore interesse porre l'attenzione sulla modalità di trasmissione dati ed i relativi aspetti di sicurezza e nei successivi paragrafi seguirà l'analisi effettuata su IPSec., anche se, come vedremo successivamente la scelta progettuale è poi stata SSL/TLS, di cui si parlerà nel capitolo seguente.

3.1 Sicurezza IP.

Il protocollo internet (IP) fornisce la funzionalità di interconnessione di sistemi posti su reti differenti, a tale scopo i dati di alto livello da scambiare vengono incapsulati in unità dati PDU[11] (*Protocol Data Unit*).

La sicurezza IP riguarda tre aree funzionali: l'autenticazione, la segretezza e la gestione delle chiavi. Il meccanismo di autenticazione garantisce che un pacchetto ricevuto sia stato realmente trasmesso dalla sorgente indicata nell' intestazione del pacchetto stesso. Inoltre, questo meccanismo garantisce che il pacchetto non sia stato alterato durante il transito. La funzionalità di segretezza consente ai nodi di comunicazione di crittografare i messaggi per impedire intercettazioni da parte di estranei. Il sistema di gestione della chiave si occupa dello scambio sicuro delle chiavi.

La sicurezza IP (IPSec. - *IP Security*) offre la possibilità di rendere sicure le comunicazioni all'interno di una rete locale, fra reti geografiche private e pubbliche e in Internet. Ecco alcuni esempi:

[11] Il PDU in IPv4 ha una dimensione pari a 160 bit, Il PDU in IPv4 ha una dimensione pari a 320 bit per garantire meccanismi maggiormente funzionali ed offrire un maggior numero di indirizzi.

- Connettività sicura delle sedi locali via Internet: un'organizzazione può realizzare una rete privata virtuale sicura attraverso Internet o una rete geografica pubblica. Questo consente all'organizzazione di sfruttare Internet per ridurre la necessità di reti private, risparmiando in costi e attività di gestione della rete.
- Attivazione della connettività extranet e intranet con partner commerciali: IPSec può essere utilizzato per rendere sicure le comunicazioni con le altre aziende, garantendo i servizi di autenticazione e segretezza e fornendo un meccanismo di scambio delle chiavi.
- Miglioramento della sicurezza del commercio elettronico: anche se alcune applicazioni Web e di commercio elettronico sono dotate di protocolli interni per la sicurezza, l'impiego di IPSec migliora questo livello di sicurezza.

La caratteristica principale di IPSec che gli consente di supportare queste differenti applicazioni è il fatto che può crittografare e/o autenticare tutto il traffico a livello IP. Pertanto possono essere rese sicure tutte le applicazioni distribuite, fra cui i login remoti, le comunicazioni client/server, la posta elettronica, il trasferimento di file, l'accesso al Web.

La Figura 3.1 mostra un tipico impiego del protocollo IPSec. Un'azienda è dotata di varie reti locali in filiali remote. All'interno di ciascuna rete locale il traffico IP non è sicuro. Per il traffico che esce dai limiti della sede, sfruttando una rete geografica privata o pubblica, vengono utilizzati i protocolli IPSec. Questi protocolli operano direttamente nei dispositivi di rete, come i router o i firewall che connettono ciascuna rete locale al mondo esterno. In genere il dispositivo di rete IPSec esegue la crittografia e la compressione di tutto il traffico in ingresso nella rete geografica e la decrittografia e la decompressione del traffico proveniente dalla rete geografica. Queste operazioni sono trasparenti per le workstation e i server della rete locale. È anche possibile attivare connessioni sicure con singoli utenti che si collegano alla rete geografica. Per garantire la sicurezza, le workstation di questi utenti devono implementare i protocolli IPSec.

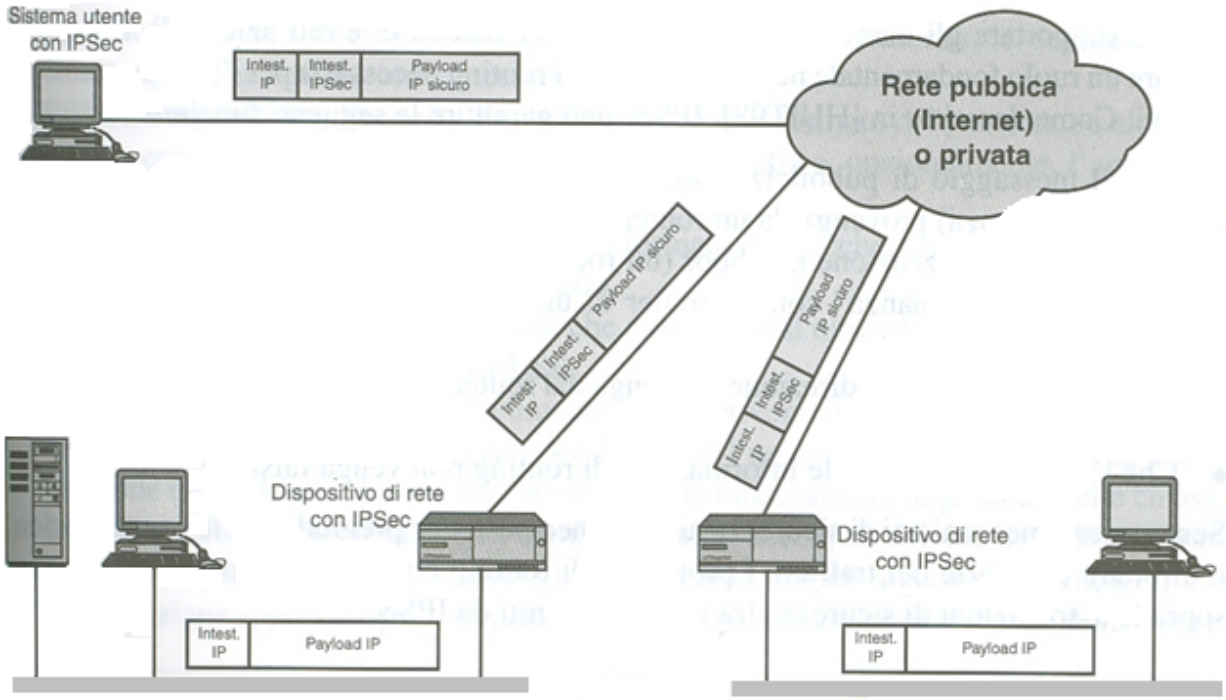


Figura 3.1: esempio di impiego del protocollo IPSec.

I vantaggi di IpSec.

- Quando IPSec viene implementato in un firewall o in un router, fornisce un livello di sicurezza elevato che può essere applicato a tutto il traffico che attraversa il perimetro. Il traffico interno dell' azienda o del gruppo di lavoro non subisce il sovraccarico rappresentato dall' elaborazione dedicata alla sicurezza.
- L'installazione di IPSec in un firewall risulterà difficile da eludere se tutto il traffico proveniente dall'esterno deve utilizzare il protocollo IP e il firewall è l'unico mezzo di ingresso da Internet verso l' azienda.
- IPSec si colloca sotto il livello di trasporto (TCP, UDP) e pertanto risulta trasparente alle applicazioni. Non vi sarà alcuna necessità di cambiare il software sul sistema dell'utente o sul server a causa dell'implementazione di IPSec nel firewall o nel router. Anche se IPSec viene implementato nei sistemi finali, il software di livello superiore, comprese le applicazioni, non verrà interessato dalla modifica.
- IPSec può essere trasparente agli utenti finali. Non vi è alcuna necessità di addestrare gli utenti sui meccanismi di sicurezza, di rilasciare informazioni

sulle chiavi per gli utenti o di revocare tali informazioni quando gli utenti lasciano l' azienda.

- IPSec può garantire anche la sicurezza dei singoli utenti. Questa soluzione è utile per coloro che lavorano fuori sede e per configurare una sottorete virtuale sicura all'interno di un' azienda per le applicazioni riservate.

3.2 Architettura IPSec.

Le specifiche di IPSec sono piuttosto complesse, per avere un'idea globale dell''architettura, si inizierà a indicare i documenti che definiscono IPSec. Poi si affronteranno i servizi di IPSec e si introdurrà il concetto di associazione di sicurezza.

Le specifiche di IPSec sono costituite da numerosi documenti. I più importanti di questi, emessi nel novembre del 1998 sono i documenti RFC 240l, 2402, 2406 e 2408.

- RFC 240 l: Panoramica dell'architettura di sicurezza.
- RFC 2402: Descrizione dell'estensione per l'autenticazione dei pacchetti in IPv4 e inIPv6.
- RFC 2406: Descrizione dell'estensione per la crittografia dei pacchetti in IPv4 e in IPv6.
- RFC 2408: Specifiche delle funzionalità di gestione delle chiavi.

L'*extension header* per l'autenticazione è chiamata ***Authentication Header***; quella per la crittografia è chiamata ESP (***Encapsulating Security Payload***).

Oltre a questi quattro documenti RFC, il gruppo IP Security Protocol Working Group di IETF ha pubblicato molte altre bozze di documenti, suddivisi in sette gruppi:

1. Architettura: descrive i concetti generali. i requisiti di sicurezza. le definizioni ed i meccanismi che definiscono la tecnologia IPSec.
2. *Encapsulating Security Payload*: descrive il formato del pacchetto e altri elementi generali relativi all'uso di ESP per la crittografia e opzionalmente,l'autenticazione.
3. *Authentication Header*: descrive il formato del pacchetto e gli elementi generali relativi all' autenticazione del pacchetto.

4. Algoritmi di crittografia: documenti che descrivono il modo in cui i vari algoritmi di crittografia vengono utilizzati per ESP.
5. Algoritmi di autenticazione: documenti che descrivono il modo in cui i vari algoritmi di autenticazione vengono utilizzati per AH e per l'opzione di autenticazione di ESP.
6. Gestione delle chiavi: documenti che descrivono i meccanismi di gestione delle chiavi.
7. DOI (*Domain of Interpretation*): contiene i valori che fanno riferimento alle relazioni fra i documenti. Fra questi vi sono gli identificatori per gli algoritmi di crittografia e autenticazione approvati, più vari parametri operativi, per esempio la durata delle chiavi.

IPSec fornisce i servizi di sicurezza a livello IP consentendo a un sistema di selezionare protocolli dì sicurezza richiesti, di determinare gli algoritmi da utilizzare per i servizi e di gestire le chiavi crittografiche necessarie per fornire i servizi richiesti. Per garantire la sicurezza vengono impiegati due protocolli: un protocollo di autenticazione stabilito dall'intestazione del protocollo, AH (***Authentication Header***), è un protocollo combinato di crittografia/autenticazione stabilito dal formato del pacchetto, ESP (***Encapsulating Security Payload***). Il supporto di queste funzionalità è obbligatorio in IPv6 e opzionale in IPv4. In entrambi i casi, le funzionalità di sicurezza sono implementate come *extension header* che seguono l'intestazione principale IP.

I servizi offerti sono i seguenti:

- controllo degli accessi;
- integrità dei dati con protocolli senza connessione;
- autenticazione dell'origine dei dati;
- segretezza (crittografia);
- segretezza parziale del flusso di traffico.

	AH	ESP (solo crittografia)	ESP (crittografia e autenticazione)
Controllo degli accessi	✔	✔	✔
Integrità senza connessione	✔		✔
Autenticazione dell'origine dei dati	✔		✔
Rifiuto dei pacchetti a replay	✔	✔	✔
Segretezza		✔	✔
Segretezza parziale del flusso di traffico		✔	✔

Figura 3.2: servizi offerti dai protocolli AH e ESP.

3.3 Associazioni di sicurezza.

Un concetto chiave nei meccanismi di autenticazione e segretezza IP è quello di associazione di sicurezza (SA - *Security Association*). Un'associazione è una relazione monodirezionale, fra un mittente ed un destinatario, che riguarda i servizi di sicurezza relativi al traffico trasportato.

Se è richiesto uno scambio sicuro bidirezionale sono necessarie due associazioni di sicurezza. I servizi di sicurezza sono assegnati ad un'associazione di sicurezza per l'impiego da parte dei protocolli AH o ESP, ma non di entrambi.

Un'associazione di sicurezza è identificata univocamente da tre parametri.

- SPI (*Security Parameters Index*): stringa di bit assegnata all'associazione di sicurezza e con significato esclusivamente locale. Il valore SPI è trasportato nelle intestazioni AH e ESP per consentire al sistema di destinazione di selezionare l'associazione di sicurezza in base alla quale elaborare il pacchetto ricevuto;
- Indirizzo IP di destinazione: attualmente è consentito solo l'impiego di indirizzi unicast, l'indirizzo della destinazione finale dell'associazione di sicurezza ovvero il sistema di un utente finale o un sistema di rete come un firewall o un router;
- Identificatore del protocollo di sicurezza: indica se l'associazione di sicurezza è di tipo AH o ESP.

Pertanto in ogni pacchetto IP, l'associazione di sicurezza è identificata univocamente dall''indirizzo di destinazione nell'intestazione IPv4 o IPv6 e dal parametro SPI nell'intestazione di estensione (AH o ESP).

In ogni implementazione di IPSec esiste un *Security Association Database* che definisce i parametri relativi a ciascuna associazione di sicurezza, che risulta definita dai seguenti parametri:

- *Sequence Number Counter*: valore a 32 bit utilizzato per generare il campo Sequence Number nell'intestazione AH o ESP (obbligatorio per tutte le implementazioni);
- *Sequence Counter Overflow*: flag che indica se un overflow del campo Sequence Number Counter deve generare un evento di audit e impedire l'ulteriore trasmissione di pacchetti su questa associazione di sicurezza (obbligatorio per tutte le implementazioni);
- *Anti-Replay Window*: utilizzato per determinare se un pacchetto AH o ESP in ingresso è un replay (obbligatorio per tutte le implementazioni);
- *AH Information*: algoritmo di autenticazione, chiavi, durata delle chiavi e altri parametri correlati utilizzati con AH (obbligatorio per le implementazioni AH);
- *ESP Information*: algoritmo di crittografia e autenticazione, chiavi, valori di inizializzazione, durata delle chiavi e parametri relativi utilizzati con ESP (obbligatorio per le implementazioni ESP);
- *Lifetime or This Security Association*: intervallo di tempo, o conteggio in byte, trascorso il quale un' associazione di sicurezza deve essere sostituita da una nuova associazione di sicurezza (un nuovo SPI) o chiusa, più l'indicazione dell'azione da eseguire alla scadenza (obbligatorio per tutte le implementazioni);
- *IPSec Protocol Mode*: tunnel, trasporto o *wildcard* (obbligatorio per tutte le implementazioni);
- *Path MTU*: massima unità di trasmissione sul percorso (dimensioni massime del pacchetto che può essere trasmesso senza frammentazione) e variabili relativi alla durata di validità (obbligatorio per tutte le implementazioni)..

IPSec fornisce all'utente una notevole flessibilità sul modo in cui i servizi IPSec vengono applicati al traffico IP che viene associato alle SA tramite il SPD (*Security Policy Database*).

Il traffico che può aggirare IPSec non viene associato ad alcuna SA. Nella sua forma più semplice, un database SPD contiene delle voci, ognuna delle quali definisce un sottoinsieme del traffico IP e punta a un'associazione di sicurezza per tale traffico. In ambienti più complessi, vi possono essere più voci potenzialmente correlate alla medesima associazione di sicurezza o più associazioni di sicurezza relative a una sola voce del database.

Ogni voce del database SPD è definita da un insieme di campi IP e di livello superiore chiamati selettori. In pratica i selettori consentono di filtrare il traffico in uscita per associarlo a una determinata associazione di sicurezza. L'elaborazione in uscita di ciascun pacchetto IP procede secondo la seguente sequenza generale:

1. Confronta i valori dei campi appropriati del pacchetto (i campi del selettore) con quelli del database SPD per trovare la voce corrispondente che punta a zero o più associazioni di sicurezza.
2. Determina l'eventuale associazione di sicurezza relativa a questo pacchetto e il corrispondente SPI.
3. Svolge l'elaborazione IPSec richiesta (ovvero l'elaborazione AH o ESP).

Una voce SPD è determinata dai seguenti selettori.

- *Destination IP Address*: può trattarsi di un singolo indirizzo IP, di un elenco o di un intervallo di indirizzi o di un insieme di indirizzi specificati con una maschera. Questi ultimi due sono necessari per supportare le situazioni in cui più sistemi di destinazione condividono la stessa associazione di sicurezza (per esempio quando i sistemi si trovano dietro un firewall).
- *Source IP Address*: può trattarsi di un unico indirizzo IP, di un elenco o di un intervallo di indirizzi o di un insieme di indirizzi specificati con una maschera. Le ultime due forme di indirizzamento sono necessarie per supportare le situazioni in cui più sistemi di origine condividono la stessa associazione di sicurezza (per esempio dietro un firewall).
- *User ID*: identificatore dell'utente ottenuto dal sistema operativo. Non si tratta di un campo IP o di livelli superiori ma è disponibile se IPSec utilizza lo stesso sistema operativo dell'utente.

- *Data Sensitivity Level*: usato per i sistemi che gestiscono la sicurezza del flusso di informazioni (per esempio Secret o Unclassified).
- *Transport Layer Protocol*: ottenuto dai campi lPv4 Protocol o IPv6 Next Header.Può essere un singolo numero di protocollo, un elenco di numeri di protocolli o un intervallo di numeri di protocolli.
- *Source Ports* e *Destination Ports*: può trattarsi di singole porte TCP o UDP, di un elenco di porte o di un'indicazione a maschera delle porte.

3.4 Le modalità *transport* e *tunnel*.

AH e ESP supportano due modalità di funzionamento: *transport* e *tunnel*. La modalità *transport* fornisce fondamentalmente la protezione dei protocolli di livello superiore. In altre parole la protezione della modalità *transport* riguarda il *payload* del pacchetto IP.

Fra gli esempi vi sono i segmenti TCP o UDP o i pacchetti ICMP che operano direttamente sopra il livello IP nello stack di protocolli dell'host. In genere la modalità transport viene utilizzata per le comunicazioni end-to-end fra due host (per esempio un client e un server o due workstation). Quando un host utilizza AH o ESP su IPv4, il payload è rappresentato dai dati che normalmente seguono l'intestazione IP. Per IPv6 il payload è rappresentato dai dati che normalmente seguono sia l'intestazione IP che eventuali intestazioni di estensione IPv6 presenti, con la possibile eccezione dell'intestazione delle opzioni della destinazione, che può essere inclusa nella protezione.

ESP in modalità *transport* esegue la crittografia e, opzionalmente, autentica il payload IP ma non l'intestazione IP, AH in modalità *transport* autentica il *payload* IP e determinate parti dell'intestazione IP.

La modalità *tunnel* fornisce la protezione dell'intero pacchetto IP. Per ottenere ciò, dopo che al pacchetto IP sono stati aggiunti i campi AH o ESP, l'intero pacchetto e i campi di sicurezza vengono trattati come payload del nuovo pacchetto IP "esterno" con una nuova intestazione IP esterna. L'intero pacchetto originario, quello interno, viaggia in una sorta di "*tunnel*" da un punto all'altro della rete IP; nessun router lungo il percorso sarà in grado di esaminare l'intestazione IP interna. Poiché il pacchetto originale è incapsulato, il nuovo pacchetto esterno può avere indirizzi di destinazione e di origine completamente differenti, in modo da migliorare ulteriormente la sicurezza. La modalità

tunnel viene utilizzata quando una o entrambe le estremità di un'associazione di sicurezza sono costituite da gateway di sicurezza. per esempio un firewall o un router che implementano IPSec. Con la modalità *tunnel*, gli host posti su reti protette da firewall possono intrattenere comunicazioni sicure senza implementare IPSec. I pacchetti non protetti generati da questi host vengono convogliati in un "*tunnel*" attraverso le reti esterne grazie alle associazioni di sicurezza in modalità *tunnel*. configurate dal software IPSec nel firewall o nel router di sicurezza che si trova al confine della rete locale.

Ecco un esempio di funzionamento della modalità *tunnel* di IPSec. L'host A di una rete genera un pacchetto IP con l'indirizzo di destinazione dell'host B che si trova su un'altra rete. Questo pacchetto viene indirizzato dall'host di origine a un firewall o a un router di sicurezza posto ai confini della rete di A.

Il firewall filtra tutti i pacchetti in uscita per determinare la necessità di elaborazioni IPSec. Se questo pacchetto da A a B richiede IPSec, il firewall svolge l'elaborazione IPSec e incapsula il pacchetto in un'intestazione IP esterna. L'indirizzo IP di origine di questo pacchetto IP esterno è questo firewall e l'indirizzo di destinazione deve essere il firewall posto sul confine della rete locale di B. Questo pacchetto viene poi instradato al firewall di B tramite i router intermedi, che esaminano solo l'intestazione IP esterna. Giunto al firewall di B, l'intestazione IP esterna del pacchetto viene eliminata e a B viene consegnato il pacchetto interno.

ESP in modalità tunnel esegue la crittografia e, opzionalmente, l'autenticazione, dell''intero pacchetto IP interno, compresa l'intestazione IP interna.

AH in modalità tunnel autentica l'intero pacchetto IP interno e determinate porzioni dell'intestazione IP esterna.

	Modalità transport	Modalità tunnel
AH	Autentica il payload IP e determinate parti dell'intestazione IP e delle intestazioni di estensione IPv6.	Autentica l'intero pacchetto IP interno (intestazione interna più payload IP) più determinate parti dell'intestazione IP esterna e delle intestazioni di estensione IPv6 esterna.
ESP	Esegue la crittografia del payload IP e di ogni intestazione di estensione IPv6 che segue l'intestazione ESP.	Esegue la crittografia del pacchetto IP interno.
ESP con autenti-cazione	Esegue la crittografia del payload IP e di ogni intestazione di estensione IPv6 che segue l'intestazione ESP. Autentica il payload IP ma non l'intestazione IP.	Esegue la crittografia del pacchetto IP interno. Autentica il pacchetto IP interno.

Figura 3.3: funzionalità modalità tunnel e transport.

3.5 Authentication Header.

Authentication Header (AH) fornisce il supporto per l'integrità dei dati e per l'autenticazione dei pacchetti IP. La funzionalità di integrità dei dati garantisce che non sia possibile modificare il contenuto del pacchetto in transito senza che tale operazione venga rilevata. La funzionalità di autenticazione consente a un sistema terminale o a un dispositivo di rete, di autenticare l'utente o l'applicazione e di filtrare il traffico. Inoltre impedisce gli attacchi a indirizzo IP fasullo (*IP spoofing*), attualmente molto diffusi in Internet.

L'autenticazione si basa su un codice MAC, pertanto le due parti devono condividere una chiave segreta. L'intestazione *Authentication Header* è costituita dai seguenti campi:

- *Next Header* (8 bit): identifica il tipo di intestazione che segue immediatamente questa intestazione.
- *Payload Length* (8 bit): lunghezza di *Authentication Header* in word di 32 bit, meno 2. Per esempio, la lunghezza standard del campo dati di autenticazione è di 96 bit ovvero tre word di 32 bit. Con un'intestazione fissa di 3 word, nell'intestazione vi saranno in totale 6 word e il campo *Payload Length* conterrà il valore 4.
- *Reserved* (16 bit): riservato per utilizzi futuri.
- *Security.Parameters Index* (32 bit): identifica un'associazione di sicurezza.
- *Sequence Number* (32 bit): un contatore incrementale monotonico.
- *Authentication Data* (variabile): un campo di lunghezza variabile (deve essere però un multiplo intero di word di 32 bit) che contiene il valore ICV (*lntegrity Check Value*) o MAC di questo pacchetto.

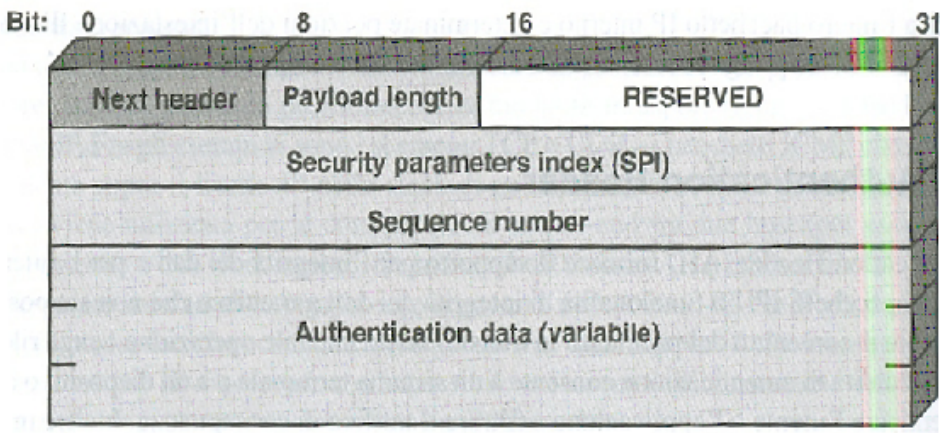


Figura 3.4: *Authentication Header* di IPSec.

Come detto, il campo *Authentication Data* contiene un valore chiamato *Integrity Check Value*, un codice di autenticazione del messaggio o una versione troncata di un codice prodotto da un algoritmo MAC. Le specifiche correnti stabiliscono che un'implementazione compatibile debba supportare:

- HMAC-MD5-96;
- HMAC-SHA-1-96.

Entrambi usano l'algoritmo HMAC, il primo con codice hash MD5 e il secondo con codice hash SHA-l. In entrambi i casi viene calcolato il valore HMAC completo ma questo valore viene poi troncato utilizzando i primi 96 bit che rappresentano la lunghezza standard del campo *Authentication Data*.

La Figura 3.5 mostra due modi in cui può essere utilizzato il servizio di autenticazione di IPSec. In un caso l'autenticazione viene fornita direttamente fra il server e le workstation cIient. La workstation può essere sulla stessa rete del server o su una rete esterna. Se la workstation e il server condividono una chiave segreta protetta, il processo di autenticazione è sicuro. Questo caso utilizza un'associazione di sicurezza in modalità transport. Nell'altro caso, una workstation remota si autentica presso il firewall aziendale per accedere all'intera rete interna o perché il server richiesto non supporta la funzionalità di autenticazione. Questo caso usa l'associazione di sicurezza in modalità tunnel.

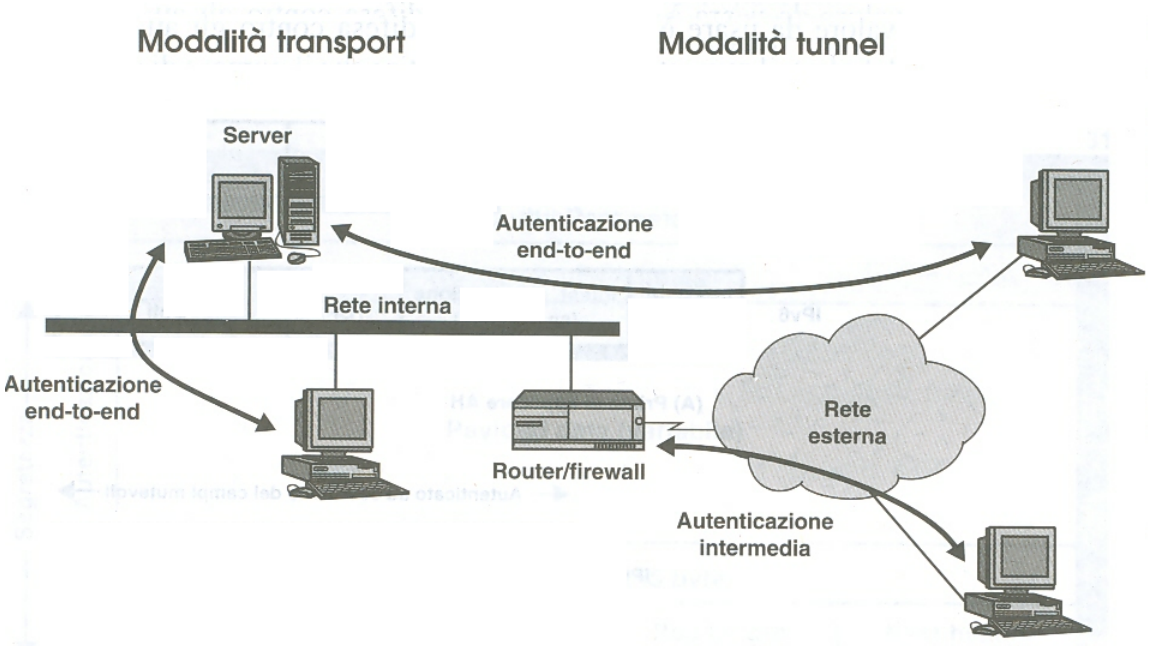


Figura 3.5: autenticazione in IPSec.

3.6 L'*Encapsulating Security Payload* (ESP).

Encapsulating Security Payload (ESP) fornisce servizi di segretezza, fra cui la segretezza del contenuto del messaggio e una parziale segretezza del flusso di traffico. Opzionalmente ESP può fornire un servizio di autenticazione.

Il formato di un pacchetto ESP consta dei seguenti campi:

- *Security Parameters Index* (32 bit): identifica un'associazione di sicurezza.
- *Sequence Number* (32 bit): contatore incrementale monotonico
- *Payload Data* (variabile): il segmento di trasporto dei dati (modalità transport) o il pacchetto IP (modalità tunnel) protetto dalla crittografia.
- Padding (0-255 byte): con lo scopo di riempimento.
- *Pad Length* (8 bit): indica il numero di byte di riempimento che precedono questo campo.
- *Next Header* (8 bit): identifica il tipo di dati contenuti nel campo *Payload Data* identificando la prima intestazione contenuta (per esempio un'extension header IPv6 o un protocollo di livello superiore come TCP).
- *Authentication Data* (variabile): campo di lunghezza variabile (ma costituito da un numero intero di word di 32 bit) che contiene il valore *Integrity Check Value* calcolato sul pacchetto ESP meno il campo *Authentication Data.*

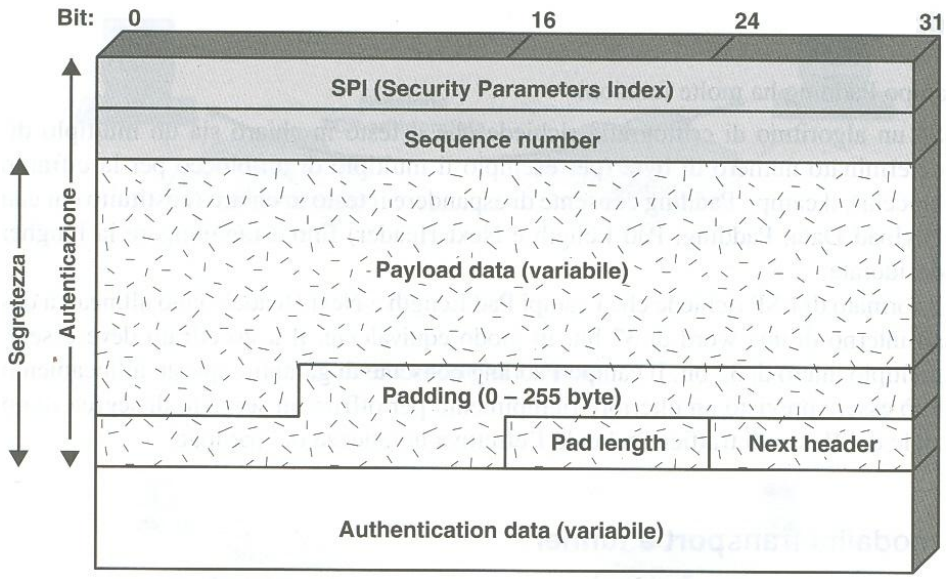


Figura 3.6: il formato di un pacchetto ESP.

I campi *Payload Data*, *Padding*, *Pad Length* e *Next Header* sono crittografati dal servizio ESP. Le specifiche correnti dicono che un'implementazione compatibile deve supportare l'algoritmo DES in modalità CBC (*Cipher Block Chaining*), nel documento DOI sono stati assegnati anche gli identificatori di alcuni altri algoritmi che possono essere utilizzati per la crittografia, fra cui:

- Triple DES a tre chiavi;
- RC5;
- CAST;
- *Blowfish*.

Anche ESP (come AH) supporta l'uso di un codice MAC con una lunghezza standard di 96 bit. Sempre come AH, le specifiche correnti dicono che un'implementazione compatibile deve supportare HMAC-MD5-96 e HMAC-SHA-1-96.

In proposito del campo *Padding*, illustriamo le sue molteplici funzioni:

- Se un algoritmo di crittografia richiede che il testo in chiaro sia un multiplo di un determinato numero di byte (per esempio il multiplo di un blocco per la cifratura a blocchi). Il campo *Padding* consente di espandere il testo in chiaro (costituito dai campi *Payload Data*, *Padding*, *Pad Length* e *Next Header*) fino a raggiungere la lunghezza desiderata.
- Il formato di ESP richiede che i campi *Pad Length* e *Next Header* siano allineati a destra all'interno di una word di 32 bit. In modo equivalente, il testo cifrato deve essere un multiplo intero di 32 bit. Il campo Padding consente di garantire questo allineamento.
- Può essere previsto un ulteriore riempimento per offrire un servizio di segretezza parziale del flusso di traffico, celando l'effettiva lunghezza del payload.

3.6.1 ESP, modalità di impiego: *transport* e *tunnel*.

La Figura 3.7 mostra due modalità di impiego del servizio ESP di IPSec. Nella parte superiore della figura. la crittografia (e opzionalmente l’autenticazione) è impiegata direttamente fra i due host. Nell’altra immagine mostra come la modalità tunnel possa essere utilizzata per creare una rete privata virtuale. In questo esempio un'azienda ha quattro reti private interconnesse tramite Internet. Gli hosts delle reti interne usano

Internet per trasportare i dati ma senza interagire con altri host in Internet. Facendo arrivare i tunnel nei gateway di sicurezza di ciascuna rete interna, questa configurazione consente di evitare di implementare negli host le funzionalità di sicurezza. La prima tecnica è supportata da un'associazione di sicurezza in modalità transport mentre l'ultima tecnica utilizza la modalità tunnel.

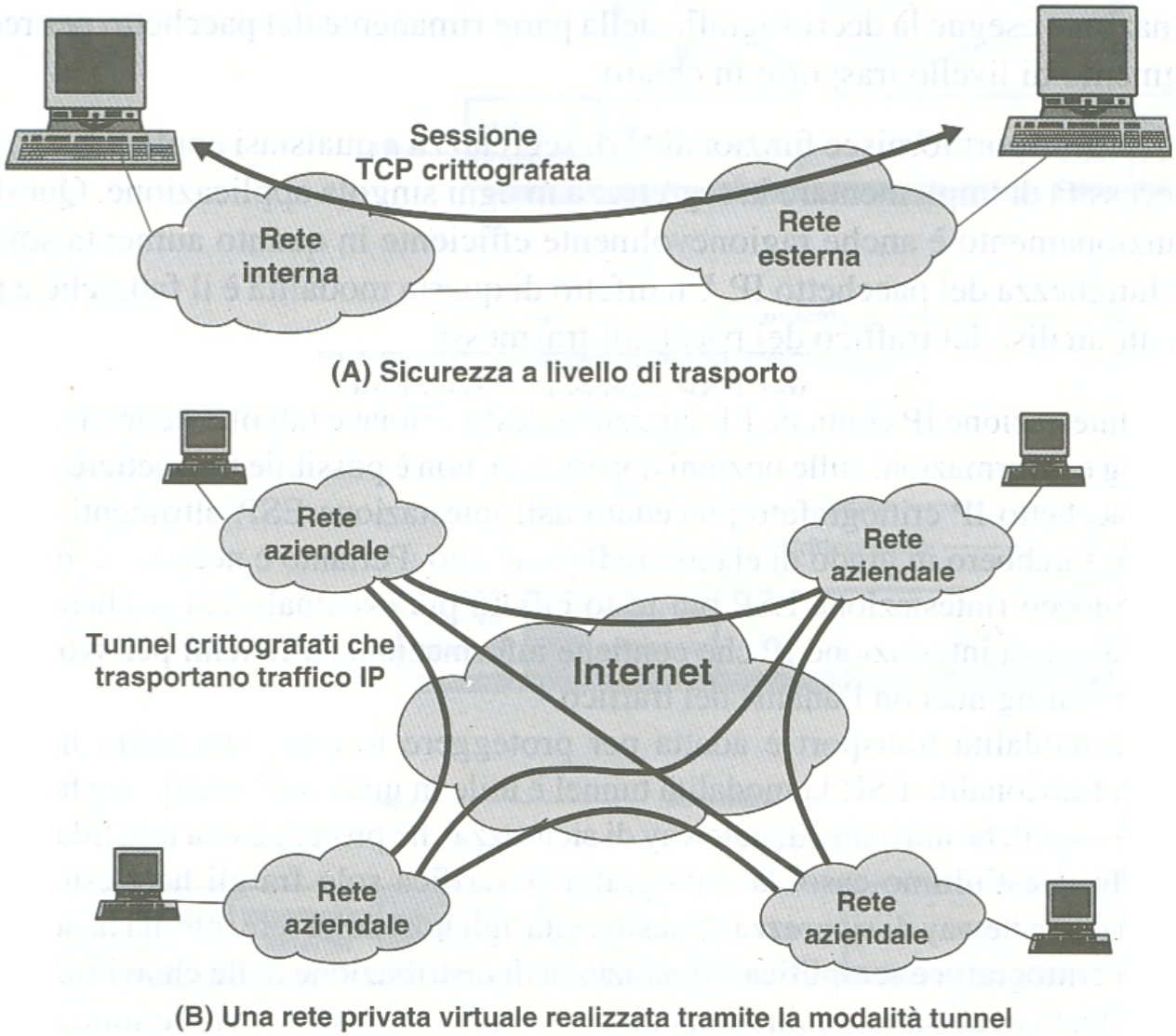


Figura 3.7: modalità trasport (A) e tunnel (B) in ESP.

La modalità transport di ESP consente di crittografare e opzionalmente autenticare i dati trasportati da IP (per esempio un segmento TCP). Per questa modalità in IPv4, l'intestazione ESP viene inserita nel pacchetto IP immediatamente prima dell'intestazione del livello di trasporto (per esempio TCP, UDP, ICMP) e dopo il pacchetto IP viene

inserita una coda ESP (i campi *Padding*, *Pad Length* e *Next Header*); se è richiesta anche l'autenticazione. Dopo la coda ESP viene aggiunto il campo ESP *Authentication Data*. Vengono crittografati l'intero segmento del livello di trasporto più la coda ESP. L'autenticazione copre tutto il testo cifrato più l'intestazione ESP.

Nel contesto di IPv6, ESP è considerato come un *payload end-to-end*, ovvero non viene né esaminato né elaborato dai router intermedi. Pertanto l'intestazione ESP si presenta dopo l'intestazione IPv6 e le *extension headers* relative al funzionamento hop-by-hop, al routing e alla frammentazione. L'*extension headers* delle opzioni di destinazione può trovarsi prima o dopo l'intestazione ESP, a seconda della semantica desiderata. In IPv6, la crittografia copre l'intero segmento di livello trasporto, più la coda ESP, più l'*extension header* delle opzioni di destinazione nel caso in cui si trovi dopo l'intestazione ESP. Anche in questo caso l'autenticazione copre il testo cifrato più l'intestazione ESP.

Il funzionamento della modalità transport può essere riepilogato nel modo seguente.

1. Alla sorgente, viene crittografato il blocco di dati costituito dalla coda ESP più l'intero segmento di livello trasporto; il testo in chiaro di questo blocco viene sostituito dal testo cifrato per formare il pacchetto IP per la trasmissione. Opzionalmente può essere aggiunta l'autenticazione.
2. Il pacchetto viene instradato alla destinazione. Ogni router intermedio deve esaminare ed elaborare l'intestazione IP più eventuali estensioni IP in chiaro ma non ha necessità di esaminare il testo cifrato.
3. Il nodo di destinazione esamina ed elabora l'intestazione IP più ogni intestazione di estensione IP in chiaro. Poi, sulla base del campo SPI dell'intestazione ESP, il nodo di destinazione esegue la decrittografia della parte rimanente del pacchetto per recuperare il segmento di livello trasporto in chiaro.

La modalità *transport* fornisce funzionalità di segretezza a qualsiasi applicazione, evitando così la necessità di implementare la segretezza in ogni singola applicazione. Questa modalità di funzionamento è anche ragionevolmente efficiente in quanto aumenta solo leggermente la lunghezza del pacchetto IP. Un difetto di questa modalità è il fatto che è possibile svolgere un' analisi del traffico dei pacchetti trasmessi.

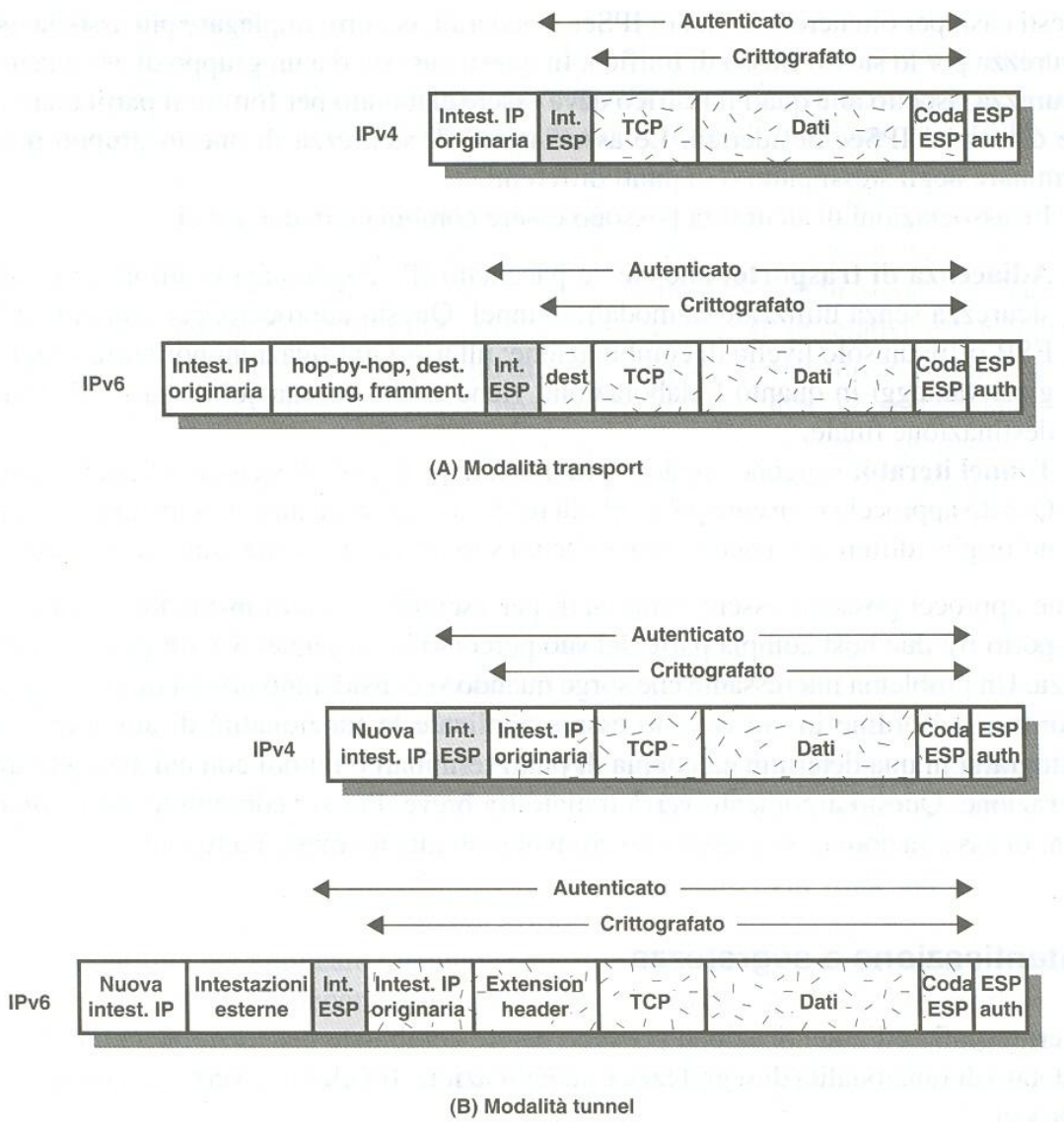


Figura 3.8: campi nella crittografia ed autenticazione ESP.

Per quanto concerne ESP in modalità tunnel, invece, si utilizza per crittografare l'intero pacchetto IP. Con questa modalità, l'intestazione ESP precede il pacchetto e vengono crittografati il pacchetto più la coda ESP.

Poiché !'intestazione IP contiene l'indirizzo di destinazione e talvolta delle direttive di *source routing* e informazioni sulle opzioni hop by hop, non è possibile trasmettere semplicemente il pacchetto IP crittografato preceduto dall'intestazione ESP, altrimenti i router intermedi non sarebbero in grado di elaborare il pacchetto. Pertanto è necessarie incapsulare l'intero blocco (intestazione ESP più testo cifrato più eventuali dati *Authentication Data*) con una nuova intestazione IP che contiene informazioni sufficienti per svolgere le operazioni di routing ma non l'analisi del traffico.

Mentre la modalità transport è adatta per proteggere le connessioni fra host che supportano la funzionalità ESP, la modalità tunnel è utile in una configurazione che include un firewall o qualche altro tipo di gateway di sicurezza che protegge una rete fidata dalle reti esterne. In quest'ultimo caso, la crittografia si verifica solo fra gli host esterni e il gateway o fra due gateway di sicurezza. Questo evita agli host della rete interna la necessità di svolgere la crittografia e semplifica l'operazione di distribuzione delle chiavi riducendo il numero di chiavi necessarie. Inoltre impedisce le analisi del traffico basate sulla destinazione finale.

Si consideri il caso in cui un host esterno voglia comunicare con un host che si trova su una rete interna, protetto da un firewall. Si supponga che ESP sia implementato nell'host esterno e nei firewall.

Per trasferire un segmento di livello trasporto dall'host esterno all'interno occorre:

1. La sorgente prepara un pacchetto IP (interno) con l'indirizzo di destinazione dell'host interno. Questo pacchetto viene fatto precedere da un'intestazione ESP, quindi il pacchetto e la coda ESP vengono crittografati con l'eventuale aggiunta di dati *Authentication Data*. Il blocco prodotto viene incapsulato in una nuova intestazione IP (l'intestazione base più le estensioni opzionali come per esempio le opzioni di routing in IPv6) il cui indirizzo di destinazione è il firewall; si forma così il pacchetto IP esterno.

2. Il pacchetto esterno viene instradato al firewall di destinazione. Ciascun router intermedio deve esaminare ed elaborare l'intestazione IP esterna più le eventuali *extension header IP* ma non ha necessità di esaminare il testo cifrato.

3. Il firewall di destinazione esamina ed elabora l'intestazione IP esterna più le eventuali extension header IP esterne. Poi, sulla base del campo SPI contenuto nell'intestazione ESP, il nodo di destinazione esegue la decrittografia della parte rimanente del pacchetto per recuperare il pacchetto IP interno; questo pacchetto viene poi trasmesso nella rete interna.

4. Il pacchetto interno viene inoltrato, attraverso uno o più router nella rete interna, fino a raggiungere l'host di destinazione.

3.7 Combinazione di più associazioni di sicurezza.

Una singola associazione di sicurezza può specificare il protocollo AH o ESP, ma non entrambi. Talvolta invece un determinato flusso di traffico richiede sia servizi forniti

da AH che da ESP. Inoltre un determinato flusso di traffico può richiedere i servizi IPSec fra gli host e, per lo stesso flusso, servizi distinti fra i gateway di sicurezza come i firewall. In tutti questi casi, per ottenere i servizi di IPSec desiderati occorre impiegare più associazioni di sicurezza per lo stesso flusso dj traffico. In questi casi si crea un gruppo di associazioni di sicurezza rispetto alle quali il traffico deve essere elaborato per fornire il particolare insieme di servizi IPSec desiderato. Le associazioni di sicurezza di questo gruppo possono terminare negli stessi punti o in punti differenti.

Le associazioni di sicurezza possono essere combinate in due modi:

Adiacenza di trasporto: allo stesso pacchetto IP vengono applicati più protocolli di sicurezza senza utilizzare la modalità tunnel. Questo approccio, per combinare AH e ESP. offre un solo livello di combinazione; ulteriori nidificazioni non forniscono maggiori vantaggi in quanto l'elaborazione viene svolta su una sola istanza di IPSec: la destinazione finale.

Tunnel iterato: vengono applicati più livelli di protocolli di sicurezza tramite tunnel IP. Questo approccio consente più livelli di nidificazione in quanto ciascun tunnel può avere un'origine differente o può terminare in un sito IPSec differente lungo il percorso.

I due approcci possono essere combinati, per esempio facendo in modo che una SA di trasporto fra due host compia parte del suo percorso in un tunnel SA fra gateway di sicurezza. Un problema interessante che sorge quando si considerano gruppi di associazioni di sicurezza è l'ordine in cui devono essere applicate le funzionalità di autenticazione e crittografia in una determinata coppia di punti terminali e i modi con cui svolgere questa operazione.

Segue una panoramica dei vari approcci di combinazione tra crittografia e autenticazione adoperate per trasmettere un pacchetto IP dotato di funzionalità di segretezza e autenticazione fra gli host.

- ESP con opzione di autenticazione

In questo approccio, illustrato nella Figura 3.8, l'utente applica innanzi tutto ESP ai dati che devono essere protetti e quindi aggiunge il campo dei dati di autenticazione. Esistono due possibilità.

1. ESP in modalità transport: l'autenticazione e la crittografia si applicano al payload IP consegnato all'host mentre l'intestazione IP non viene protetta.

2. ESP in modalità tunnel: l'autenticazione si applica all'intero pacchetto IP consegnato all'indirizzo di destinazione IP esterno (per esempio un firewall) dove viene effettuata l'autenticazione. L'intero pacchetto IP interno è protetto dal meccanismo di privacy per la consegna alla destinazione IP

In entrambi i casi, l'autenticazione si applica al testo cifrato e non al testo in chiaro.

- Adiacenza di trasporto.

Un altro modo per applicare "autenticazione dopo la crittografia consiste nell'utilizzare due associazioni di sicurezza di trasporto, dove quella interna è una associazione di sicurezza ESP e quella esterna è un'associazione di sicurezza AH. In questo caso ESP viene utilizzato senza la sua opzione di autenticazione. Poiché l'associazione di sicurezza interna è in modalità transport, la crittografia viene applicata al payload lP. Il pacchetto risultante consiste in un'intestazione IP (ed eventualmente delle intestazioni di estensione IPv6) seguita da un ESP. AH viene quindi applicato in modalità transport in modo che l'autenticazione copra l'intestazione ESP più l'intestazione originale IP (con le relative estensioni), ad eccezione dei campi mutevoli. Il vantaggio di questo approccio rispetto al semplice utilizzo dell'associazione di sicurezza ESP con opzione di autenticazione, consiste nel fatto che l'autenticazione copre più campi, compresi gli indirizzi IP di sorgente e di destinazione. Lo svantaggio è il sovraccarico dovuto al fatto di impiegare due associazioni di sicurezza invece di una.

- Raggruppamento transport-tunnel.

L'uso dell'autenticazione prima della crittografia può essere preferibile per vari motivi. Innanzitutto, poiché i dati di autenticazione sono protetti dalla crittografia, sarà impossibile intercettare il messaggio e modificare i dati di autenticazione senza che venga rilevato. In secondo luogo può essere conveniente poter memorizzare le informazioni di autenticazione insieme al messaggio per eventuali riferimenti futuri. E' più comodo svolgere questa operazione se le informazioni di autenticazione si applicano al messaggio non crittografato: in caso contrario il messaggio dovrebbe essere ricrittografato ogni volta che fosse necessario verificare le informazioni di autenticazione.

Una possibilità per applicare l'autenticazione prima della crittografia fra due host prevede l'impiego di un raggruppamento costituito da un'associazione di sicurezza di trasporto AH interna e un'associazione di sicurezza a tunnel ESP esterna. In questo caso

l'autenticazione viene applicata al payload IP e all'intestazione IP (e relative estensioni) ad esclusione dei campi mutevoli. Il pacchetto IP risultante viene poi elaborato in modalità tunnel da ESP: il risultato è che l'intero pacchetto interno autenticato viene crittografato e viene aggiunta una nuova intestazione IP esterna con le relative estensioni.

- Combinazioni di base delle associazioni di sicurezza.

Il documento *IPSec Architecture* elenca quattro esempi di combinazioni di associazioni di sicurezza che devono essere supportate dagli host compatibili IPSec (per esempio workstation e server) o dai gateway di sicurezza (per esempio firewall e router). Questi esempi sono illustrati nella figura che segue. La parte inferiore di ciascun caso della figura rappresenta la connettività fisica degli elementi; la parte superiore rappresenta la connettività logica attraverso una o più associazioni di sicurezza nidificate. Ciascuna associazione di sicurezza può essere AH o ESP. Per le associazioni di sicurezza fra host. la modalità può essere di trasporto o tunnel: altrimenti si tratta della modalità tunnel.

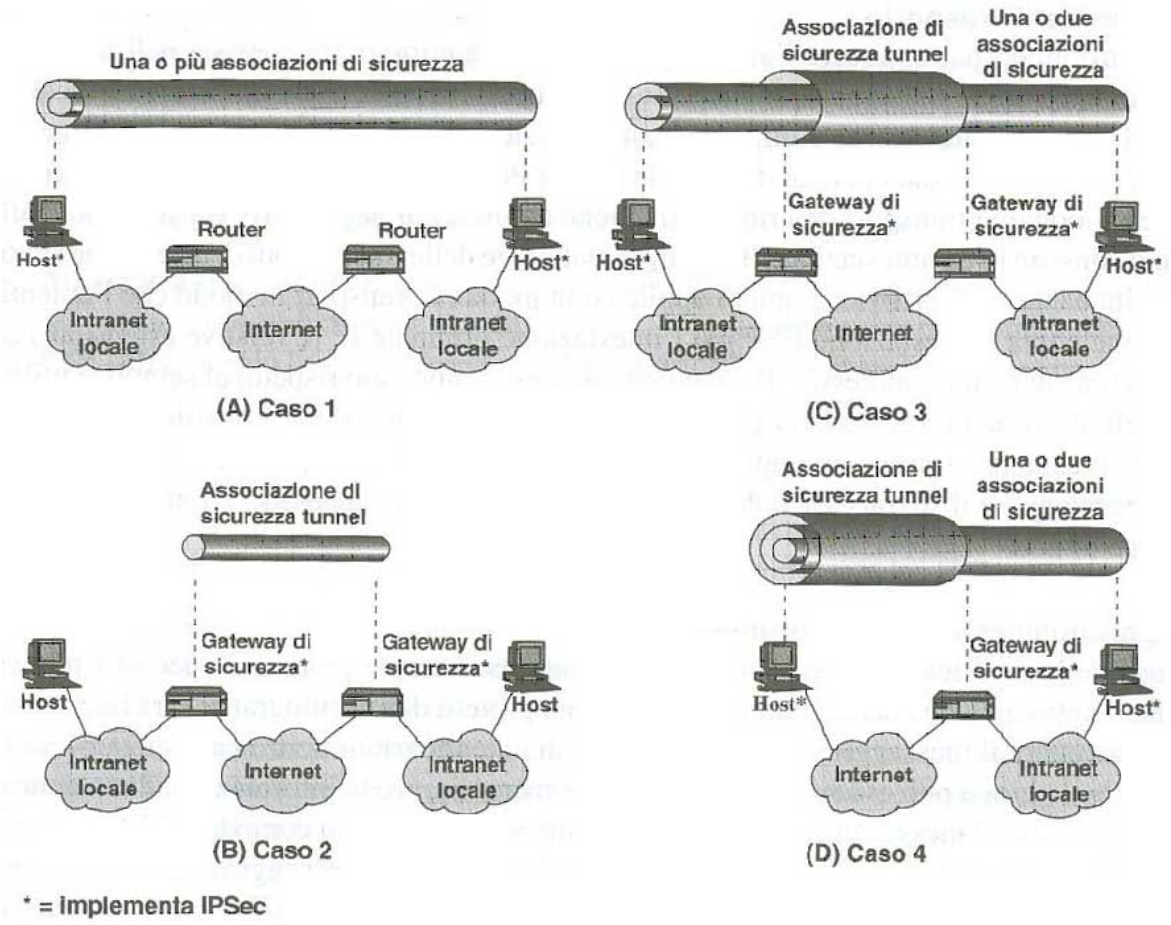


Figura 3.9: combinazioni in base alle associazioni di sicurezza.

Nel Caso 1 la sicurezza viene fornita dai sistemi terminali che implementano IPSec. Perché due sistemi terminali possano comunicare tramite un'associazione di sicurezza, devono condividere le chiavi segrete appropriate. Ecco alcune combinazioni possibili.

A. AH in modalità transport
B. ESP in modalità transport
C. AH seguito da ESP in modalità transport (un'associazione di sicurezza ESP all'interno di un'associazione di sicurezza AH)
D. A, B o C all'interno di AH o ESP in modalità tunnel.

Si è già analizzato come queste combinazioni possano essere utilizzate per supportare l'autenticazione, la crittografia, l'autenticazione prima della crittografia e l'autenticazione dopo la crittografia.

Nel Caso 2 la sicurezza viene fornita solo fra gateway (router, firewall e così via) mentre nessun host implementa IPSec. Questo caso illustra il supporto di una semplice rete privata virtuale. Il documento di architettura della sicurezza specifica che in questo caso è necessaria un'unica associazione di sicurezza a tunnel. Il tunnel può supportare AH, ESP o ESP con opzione di autenticazione. Non è necessario utilizzare tunnel nidificati, in quanto i servizi in IPSec si applicano all'intero pacchetto interno.

Il Caso 3 estende il Caso 2 aggiungendo la sicurezza end-to-end. Sono consentite le stesse combinazioni trattate per i Casi l e 2. Il tunnel gateway-gateway fornisce l'autenticazione, la segretezza o entrambe le funzionalità per tutto il traffico fra i sistemi terminali. Quando il tunnel gateway-gateway è di tipo ESP, fornisce anche una forma parziale di segretezza del traffico. I singoli host possono implementare eventuali servizi aggiuntivi IPSec necessari per determinate applicazioni o per determinati utenti, tramite associazioni di sicurezza end-to-end.

Il Caso 4 fornisce il supporto per un host remoto che utilizza Internet per raggiungere il firewall di un'azienda per poi accedere a qualche server o workstation protetto dal firewall. Fra l'host remoto e il firewall è necessaria solo la modalità tunnel. Come nel Caso I, fra l'host remoto e l'host locale possono essere utilizzate una o due associazioni di sicurezza.

La gestione delle chiavi in IPSec comporta la scelta e la distribuzione delle chiavi segrete. In genere sono richieste quattro chiavi per le comunicazioni fra due applicazioni:

due coppie di chiavi di trasmissione e ricezione sia per AH che per ESP. Il documento IPSec Architecture richiede obbligatoriamente il supporto di due tipi di gestione delle chiavi.

- Manuale: l'amministratore di sistema configura manualmente ciascun sistema con le proprie chiavi e con le chiavi degli altri sistemi. Si tratta di una soluzione pratica per ambienti piccoli e relativamente statici.
- Automatica: un sistema automatico consente la creazione su richiesta delle chiavi per le associazioni di sicurezza e ne facilita l'uso in sistemi distribuiti di grande estensione con configurazione dinamica.

Il protocollo standard per la gestione automatizzata delle chiavi per IPSec è chiamato ISAKMP/Oakley ed è costituito dai seguenti elementi:

- **Oakley Key Determination Protocol**: Oakley è un protocollo per lo scambio delle chiavi basato sull'algoritmo Diffie-Hellman, che però fornisce una sicurezza più elevata. Oakley è un protocollo generico, per il fatto che non stabilisce alcun formato specifico.
- **ISAKMP** (*Internet Security Association and Key Management Protocol*): ISAKMP fornisce una struttura per la gestione delle chiavi in Internet e il supporto, fra cui i formati, per la negoziazione degli attributi di sicurezza.

ISAKMP non impone un algoritmo specifico per lo scambio delle chiavi ma è piuttosto un insieme di tipi di messaggi che consentono di utilizzare vari algoritmi per lo scambio delle chiavi. Oakley era l'algoritmo per lo scambio delle chiavi obbligatorio nella versione iniziale di ISAKMP. Oakley è un raffinamento dell'algoritmo per lo scambio delle chiavi Diffie-Hellman, il quale prevede le seguenti interazioni fra gli utenti A e B: vi è un accordo iniziale sui parametri globali: q, un numero primo esteso e • , una radice primitiva di q. A seleziona un intero casuale X_A come chiave privata e trasmette a B la sua chiave pubblica $Y_a = \bullet^{xA} \bmod q$. Analogamente, B seleziona un intero casuale X_B come propria chiave privata e trasmette ad A la propria chiave pubblica $Y_B = \bullet^{Xb}$. Ognuno dei due lati può quindi calcolare la chiave segreta di sessione:

$$K = (Y_B)^{Xa} \bmod q = (Y_A)^{Xb} \bmod q = \bullet^{Xa\ Xb} \bmod q$$

L'algoritmo Diffie-Hellman ha due caratteristjche interessanti.

Innanzitutto luogo le chiavi segrete vengono create solo quando necessario, non vi è alcuna necessità di memorizzare le chiavi segrete per un lungo periodo di tempo, cosa che le renderebbe più vulnerabili. In secondo luogo lo scambio non richiede alcuna infrastruttura preesistente, tranne un accordo globale sui parametri.

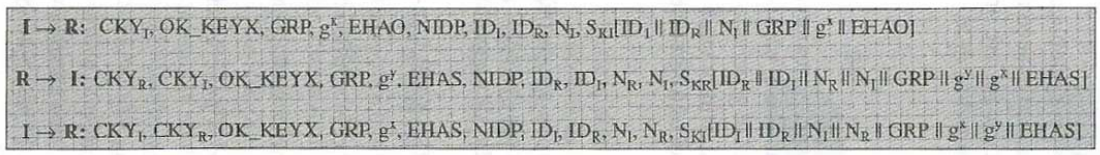
Tuttavia, l'algoritmo Diffie-Hellman presenta anche vari punti deboli:

- Non fornisce informazioni sull'identità delle parti.
- È soggetto all'attacco *man-in-the-middle*, in cui un estraneo C si finge B mentre comunica con A e si finge A mentre comunica con B. Sia A che B si trovano a negoziare una chiave con C che può quindi ascoltare e inoltrare il traffico. L'attacco *man-in-the-middle* si svolge nel seguente modo.
  - B invia la sua chiave pubblica Y_B in un messaggio indirizzato ad A.
  - L'estraneo (E) intercetta questo messaggio. E salva la chiave pubblica di B e invia un messaggio ad A che ha il codice utente di B ma la chiave pubblica di E, Y_E. Questo messaggio viene inviato in modo che sembri provenire dal sistema host di B. A riceve il messaggio di E e memorizza la chiave pubblica di E con il codice utente di B. Analogamente, E invia un messaggio a B con la propria chiave pubblica, facendo in modo che sembri provenire da A.
  - B calcola una chiave segreta K_1 basata sulla propria chiave privata e su Y_E. A calcola una chiave segreta K_2 basata sulla propria chiave privata e su Y_E. E calcola K_1 utilizzando la propria chiave segreta X_E e Y_B e calcola K_2 utilizzando la propria chiave segreta X_E e Y_A.
  - D'ora in poi E è in grado di rinviare i messaggi fra A e B e fra B e A, cambiando in modo appropriato la cifratura lungo il percorso in modo che né A né B possano rendersi conto che stanno condividendo la comunicazione con E.
- E’ computazionalmente intensivo, ed è quindi vulnerabile a un attacco *clogging*, in cui un estraneo richiede un elevato numero di chiavi. La vittima dedica una notevole quantità di risorse di calcolo a svolgere inutili calcoli esponenziali modulari.

Oakley è progettato per mantenere i vantaggi di Diffie-Hellman evitando nel contempo i suoi punti deboli, risulta caratterizzato da cinque importanti caratteristiche:

1. Impiega un meccanismo di *cookie* per evitare gli attacchi *clogging*.
2. Consente alle due parti di negoziare i parametri globali dello scambio delle chiavi Diftie-Hellman.
3. Consente lo scambio delle chiavi pubbliche Diffie-Hellman.
4. Autentica lo scambio Diffie-Hellman per sventare l'attacco *man-in-the-middle*.

Le specifiche Oakley includono vari esempi di scambio delle chiavi consentiti dal protocollo. Per dare un'idea del funzionamento di Oakley, si presenterà un esempio che nelle specifiche è chiamato scambio aggressivo delle chiavi poiché vengono scambiati solo tre messaggi.



Notazione:

I	=	iniziatore
R	=	risponditore
CKY_I, CKY_R	=	cookie dell'iniziatore e del risponditore.
OK_KEYX	=	tipo di messaggio per lo scambio della chiave.
GRP	=	nome del gruppo Diffie-Hellman per questo scambio.
g^x, g^y	=	chiave pubblica dell'iniziatore e del risponditore. g^{xy} = chiave di sessione per questo scambio.
EHAO, EHAS	=	algoritmi di crittografia (Encryption), Hash e Autenticazione offerti e selezionati.
NIDP	=	indica che la crittografia non verrà utilizzata per la parte rimanente del messaggio.
ID_I, ID_R	=	identificatore dell'iniziatore e del risponditore.
N_I, N_R	=	codice nonce casuale fornito dall'iniziatore e dal risponditore per questo scambio.
$S_{KI}[X]$, $S_{KR}[X]$	=	indica la firma su X utilizzando la chiave privata (chiave di firma) dell'iniziatore e risponditore.

Figura 3.10: Esempio di scambio delle chiavi Oakley aggressivo.

La figura mostra il protocollo di scambio aggressivo delle chiavi. Nel primo passo, l'iniziatore (I) trasmette un *cookie*, il gruppo da utilizzare e la propria chiave pubblica Diffie-Hellman per questo scambio. I indica anche gli algoritmi di crittografia a chiave pubblica, di calcolo hash e di autenticazione offerti da utilizzare in questo scambio. Sempre nel messaggio sono inclusi gli identificatori di I e del risponditore R, più il valore *nonce* di I per questo scambio. Infine, I aggiunge una firma utilizzando la propria chiave privata che firma i due identificatori, il *nonce*, il gruppo, la chiave pubblica Diffie-Hellman e gli algoritmi offerti.

Quando R riceve il messaggio verifica la firma utilizzando la chiave pubblica di I. R conferma la ricezione del messaggio rimandando a I il *cookie*, l'identificatore e il nonce

insieme al gruppo. R include nel messaggio anche un *cookie*, la propria chiave pubblica Diffie-Hellman, gli algoritmi selezionati (che devono essere fra gli algoritmi offerti), il proprio identificatore e il proprio valore *nonce* per questo scambio. Infine R aggiunge una firma utilizzando la propria chiave privata che firma i due identificatori, i due *nonce*, il gruppo, le due chiavi pubbliche Diffie-Hellman e gli algoritmi selezionati.

Quando I riceve il secondo messaggio, verifica la firma utilizzando la chiave pubblica di R. I valori *nonce* contenuti nel messaggio garantiscono che questo non sia un attacco a replay che impiega un vecchio messaggio. Per completare lo scambio, I deve inviare un messaggio a R per consentirgli di verificare che I abbia ricevuto la chiave pubblica di R.

ISAKMP

ISAKMP definisce le procedure e i formati dei pacchetti necessari per attivare, negoziare, modificare e cancellare le associazioni di sicurezza. Nell'ambito dell'attivazione di un 'associazione di sicurezza, ISAKMP definisce il payload per lo scambio dei dati di generazione e autenticazione delle chiavi. Questi formati di payload costituiscono una struttura indipendente dallo specifico protocollo di scambio delle chiavi, dall'algoritmo di crittografia e dal meccanismo di autenticazione.

Un messaggio ISAKMP è costituito da un'intestazione ISAKMP seguita da uno o più carichi utili. Tutto ciò viene trasportato in un protocollo di trasporto. Le specifiche stabiliscono che le implementazioni debbano supportare l'impiego del protocollo di trasporto UDP.

L'intestazione di un messaggio ISAKMP è costituita dai seguenti campi.

- *Initiator Cookie* (64 bit): cookie dell'entità che ha iniziato l'attivazione, la notifica o la cancellazione del!' associazione di sicurezza.
- *Responder Cookie* (64 bit): cookie dell'entità rispondente; nullo nel primo messaggio dell'iniziatore.
- *Next Payload* (8 bit): indica il tipo del primo *payload* del messaggio;
- *Major Version* (4 bit): indica la versione (*major*) di ISAKMP utilizzata.
- *Minor Version* (4 bit): indica la versione (*minor*) di ISAKMP utilizzata.
- *Exchange Type* (8 bit): indica il tipo di scambio; se ne parlerà più avanti.

- *Flags* (8 bit): indica le opzioni impostate per questo scambio ISAKMP. I due bit attualmente definiti sono: il bit *Encryption*, impostato se tutti i carichi utili che seguono l'intestazione sono crittografati utilizzando l'algoritmo di crittografia di questa associazione di sicurezza, ed il bit *Commit*, che viene usato per garantire che il materiale crittografato non venga ricevuto prima del completamento dell'attivazione dell'associazione di sicurezza.
- *Message ID* (32 bit): codice univoco di questo messaggio.
- *Length* (32 bit): lunghezza totale del messaggio (intestazione più tutti i carichi utili) misurata in ottetti.

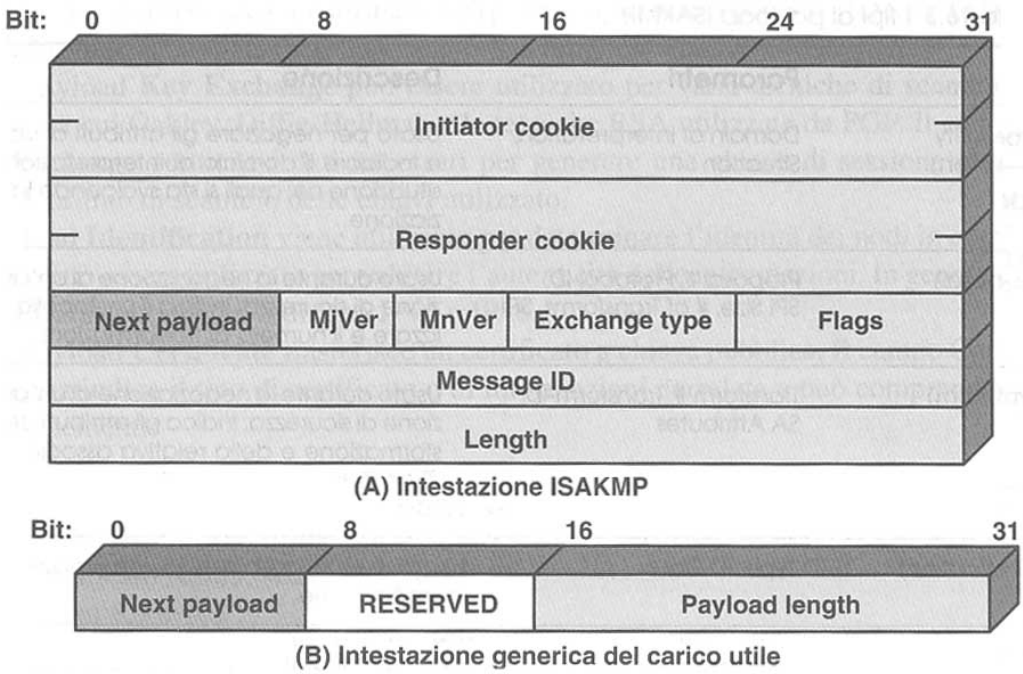


Figura 3.10: Formato dell'intestazione di un messaggio ISAKMP

Tutti i carichi utili ISAKMP iniziano con la stessa intestazione generica rappresentata nella parte B della figura, il campo *Next Payload* ha il valore 0 se questo è l'ultimo *payload* del messaggio altrimenti il suo valore è il tipo del *payload* successivo. Il campo *Payload Length* indica la lunghezza in ottetti del *payload*, compresa l'intestazione generica.

ISAKMP definisce vari tipi di payload, nella tabella che segue sono elencati i campi o parametri per ciascun tipo.

Tipo	Parametri	Descrizione
SA (Security Association)	Domain of Interpretation, Situation	Usato per negoziare gli attributi di sicurezza e indicare il dominio di interpretazione e la situazione nei quali si sta svolgendo la negoziazione.
P (Proposal)	Proposal #, Protocol-ID, SPI Size, # of Transforms, SPI	Usato durante la negoziazione di un'associazione di sicurezza: indica il protocollo da utilizzare e il numero di trasformazioni.
T (Transform)	Transform #, Transform-ID, SA Attributes	Usato durante la negoziazione di un'associazione di sicurezza: indica gli attributi della trasformazione e della relativa associazione di sicurezza.
KE (Key Exchange)	Key Exchange Data	Supporta varie tecniche di scambio delle chiavi.
ID (Identification)	ID Type, ID Data	Usato per scambiare le informazioni di identificazione.
CERT (Certificate)	Cert Encoding, Certificate Data	Usato per trasportare i certificati e altre informazioni correlate.
CR (Certificate Request)	# Cert Types, Certificate Types, # Cert Auths, Certificate Authorities	Usato per richiedere i certificati: indica i tipi di certificati richiesti e le autorità di certificazione accettate.
HASH (Hash)	Hash Data	Contiene i dati generati da una funzione hash.
SIG (Signature)	Signature Data	Contiene i dati generati da una funzione di firma digitale.
NONCE (nonce)	Nonce Data	Contiene un codice nonce.
N (Notification)	DOI, Protocol-ID, SPI Size, Notify Message Type, SPI, Notification Data	Usato per trasmettere i dati di notifica, come per esempio una condizione d'errore.
D (Delete)	DOI, Protocol-ID, SPI Size, # of SPIs, SPI (uno o più)	Indica che un'associazione di sicurezza non è più valida.

ISAKMP fornisce una struttura per lo scambio dei messaggi, dove i tipi di payload servono come elementi costitutivi. Le specifiche identificano cinque tipi di scambio standard che dovrebbero sempre essere supportati.

Scambio	Nota
A. Scambio Base	
(1) I → R: SA; NONCE	Inizia la negoziazione dell'associazione di sicurezza ISAKMP.
(2) R → I: SA; NONCE	Associazione di sicurezza base concordata.
(3) I → R: KE; ID_I; AUTH	Chiave generata; identità dell'iniziatore verificata dal risponditore.
(4) R → I: KE; ID_R; AUTH	Identità del risponditore verificata dall'iniziatore; chiave generata; associazione di sicurezza attivata.
B. Scambio Identity Protection	
(1) I → R: SA	Inizia la negoziazione dell'associazione di sicurezza ISAKMP.
(2) R → I: SA	Associazione di sicurezza base accordata.
(3) I → R: KE; NONCE	Chiave generata.
(4) R → I: KE; NONCE	Chiave generata.
(5)* I → R: ID_I; AUTH	Identità dell'iniziatore verificata dal risponditore.
(6)* R → I: ID_R; AUTH	Identità del risponditore verificata dall'iniziatore; associazione di sicurezza attivata.
C. Scambio Authentication Only.	
(1) I → R: SA; NONCE	Inizia la negoziazione dell'associazione di sicurezza ISAKMP.
(2) R → I: SA; NONCE; IDR; AUTH	Associazione di sicurezza base accordata; identità del risponditore verificata dall'iniziatore.
(3) I → R: ID_I; AUTH	Identità del risponditore verificata dall'iniziatore; associazione di sicurezza attivata.
D. Scambi Aggressive	
(1) I → R: SA; KE; NONCE; IDI	Inizia la negoziazione dell'associazione di sicurezza ISAKMP e lo scambio delle chiavi.
(2) R → I: SA; KE; NONCE; IDR; AUTH	Identità del risponditore verificata dall'iniziatore; chiave generata; associazione di sicurezza base accordata.
(3)* I → R: AUTH	Identità del risponditore verificata dall'iniziatore; associazione di sicurezza attivata.
E. Scambio Informational	
(1) I → R: N/D;	Cancellazione o notifica di errore o di stato.

Notazione
I = iniziatore
R = risponditore
* = crittografia del payload dopo l'intestazione ISAKMP.
AUTH = meccanismo di autenticazione impiegato.

Lo scambio **Base** consente lo scambio contemporaneo delle chiavi e delle informazioni di autenticazione. Questo riduce il numero di scambi ma presenta il difetto di non proteggere l'identità. I primi due messaggi forniscono i cookie e attivano un'associazione di sicurezza, le trasformazioni e il protocollo concordati; entrambe le parti usano un codice nonce per proteggersi dagli attacchi a replay. Gli ultimi due messaggi scambiano le informazioni delle chiavi e i codici ID utente con un meccanismo

di autenticazione utilizzato per autenticare le chiavi, le identità e i codici nonce dei primi due messaggi.

Lo scambio ***Identity Protection*** espande lo scambio Base per proteggere le identità degli utenti. I primi due messaggi attivano l'associazione di sicurezza. I due messaggi successivi eseguono lo scambio delle chiavi pubbliche.

Lo scambio ***Authentication Only*** viene utilizzato per svolgere la reciproca autenticazione senza scambio di chiavi. I primi due messaggi attivano l'associazione di sicurezza. Inoltre il risponditore utilizza il secondo messaggio per trasferire il proprio codice utente e utilizza l'autenticazione per proteggere il messaggio. L'iniziatore invia il terzo messaggio per trasmettere il proprio codice utente autenticato.

Lo scambio ***Aggressive*** riduce il numero di scambi ma non garantisce la protezione dell'identità. Nel primo messaggio l'iniziatore propone un'associazione di sicurezza offrendo dei protocolli e delle opzioni di trasformazione. L'iniziatore attiva anche lo scambio della chiave e fornisce il proprio codice utente. Nel secondo messaggio, il risponditore indica se ha accettato l'associazione di sicurezza con un determinato protocollo e una determinata trasformazione, completa lo scambio della chiave e autentica le informazioni trasmesse. Nel terzo messaggio l'iniziatore autentica le informazioni precedenti, crittografandole con la chiave di sessione segreta condivisa.

Lo scambio ***Informational*** viene utilizzato per la trasmissione monodirezionale di informazioni per la gestione dell'associazione di sicurezza.

CAPITOLO 4

Nella fase di scambio dati effettuata tramite Internet tra un Proxy ed il Server Regionale, come già affrontato nel capitolo precedente, occorre prevedere un approccio che garantisca adeguati meccanismi di sicurezza.

Il vantaggio di IPSec consiste nel fatto che è trasparente agli utenti finali e alle applicazioni, inoltre include una funzionalità di filtraggio cosicché non tutto il traffico sia sottoposto al sovraccarico dovuto alle elaborazioni IPSec.

Tuttavia la scelta progettuale nel disegnare l'architettura del sistema AIS regionale è ricaduta su SSL/TLS, una soluzione che implementa la sicurezza appena sopra il livello trasporto, pertanto trasparente alle applicazioni, e che può essere incorporato in specifici pacchetti.

Nello scambio di dati mediante Internet si vuol garantire un trasporto efficiente, ma allo stesso tempo si desidera ottenere una trasmissione sicura. Questo significa proteggere da alterazioni o letture non autorizzate le informazioni trasmesse.

SSL è il protocollo che garantisce tutto questo, prevenendo intrusioni e manomissioni dei messaggi scambiati, e permettendo di certificare l' identità degli interlocutori

In questo capitolo si evidenzieranno le caratteristiche del meccanismo di sicurezza SSL/TLS; non si esclude che, dopo un opportuno periodo di valutazione, sulla base di quanto analizzato in questo lavoro di tesi, si possa successivamente valutare di adoperare come soluzione alternativa la sicurezza a livello IP.

4.1 Funzionalità SSL.

Quando si usa il web esistono vari problemi di sicurezza che si devono affrontare. La comunicazione in rete tra due host solitamente non avviene in maniera diretta, ma coinvolge tutta una serie di sistemi di computer che funzionano da intermediari nel trasporto dei dati. SSL si occupa di assicurare che questi sistemi intermedi non interferiscano nella sessione o acquisiscano informazioni confidenziali. A tal fine SSL implementa precise funzionalità di autenticazione, cifratura e verifica dell'integrità dei dati. Più in dettaglio il protocollo fornisce le seguenti caratteristiche:

Autenticazione: È il processo grazie al quale si è in grado di stabilire l'identità dell'interlocutore, garantendo alle parti in causa di essere in comunicazione con entità fidate. I due host si scambiano certificati di identità, la cui validità è sottoscritta da enti considerati attendibili dagli utenti. Questi enti sono chiamati *Certificate Authorities* (CA). In un modello di comunicazione client-server SSL permette l'autenticazione dell'uno e dell'altro.

Autenticazione del server: consente ad un generico utente di confermare l'identità del server a cui si sta connettendo. In un collegamento via SSL è previsto che il server invii il proprio certificato di identità al client che richiede una comunicazione protetta. Per prima cosa l'host dell'utente verifica che il certificato sia autentico, assicurandosi che sia controfirmato da una delle CA appartenenti alla lista delle CA di fiducia. Il client si preoccupa poi di verificare che il certificato ricevuto sia intestato proprio al server richiesto, per scongiurare il pericolo che qualcuno si sia voluto intromettere nella comunicazione inviando il proprio certificato. Il procedimento consente quindi al client di autenticare il server prima di iniziare una comunicazione protetta.

Autenticazione del client: permette ad un server di confermare l'identità di un utente. Il procedimento è analogo a quello previsto per l'autenticazione del server, solo con le parti in gioco scambiate. È infatti l'utente che invia il proprio certificato al server e questo ne verifica l'autenticità. Lo standard SSL prevede come obbligatoria soltanto l'autenticazione del server, quella del client è opzionale, ma risulta fondamentale quando le informazioni confidenziali viaggiano dal server verso l'utente.

Privatezza del collegamento: i dati sensibili scambiati sul canale di comunicazione sono protetti utilizzando algoritmi di crittografia a chiave simmetrica, in cui cioè la stessa chiave (considerata sicura) è utilizzata sia in fase di cifratura dei dati che in fase di decodifica. In questo modo i dati confidenziali possono essere letti soltanto da coloro che conoscono la chiave concordata per la comunicazione. SSL mette a disposizione più algoritmi di cifratura simmetrica (DES, RC4, ecc.), con diversi livelli di sicurezza, permettendo una qualità del servizio adeguata alla sensibilità dei dati.

Integrità delle informazioni: i dati, prima di essere inviati, vengono autenticati mediante un campo *Message Autentication Code* (MAC), generato mediante le funzioni

hash di firma (MD5, SHA, ecc.) che SSL mette a disposizione. Quando i dati vengono ricevuti si può quindi verificare che non siano stati modificati semplicemente controllando il campo MAC.

4.2 Architettura SSL.

SSL (*Secure Socket Layer*) è stato sviluppato da Netscape. La versione 3 del protocollo è stata progettata con un processo di revisione pubblica e sulla base di input dal mondo delle aziende ed è stata pubblicata come bozza Internet. Successivamente. raggiunto un certo consenso per proporre il protocollo come standard Internet. venne costituito il gruppo di lavoro TLS nell'ambito di IETF per sviluppare uno standard comune. Questa prima versione di TLS può essere considerata fondamentalmente come la versione 3.1 di SSL ed è molto simile e compatibile all'indietro con SSL versione 3. L'architettura SSL è stata progettata per impiegare TCP con un servizio affidabile end-to-end. SSL non è un unico protocollo ma è costituito da due livelli di protocolli, come indicato nella seguente figura.

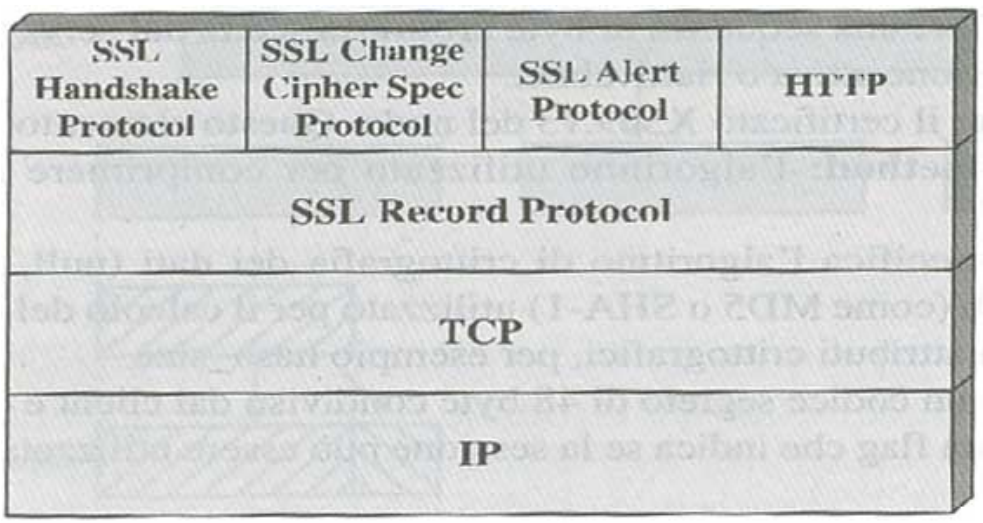


Figura 4.1: stack del protocollo SSL.

Il protocollo *SSL Record Protocol* fornisce i servizi di sicurezza di base per i vari protocolli di livello superiore. In particolare. il protocollo HTTP (*Hyper Terminal Transfer Protocol*). che fornisce il servizio di trasferimento per le interazioni Web client/server. può operare sopra SSL. Nell'ambito di SSL sono definiti tre protocolli di

più alto livello utilizzati nella gestione degli scambi SSL: *Handshake Protocol*, *Change Cipher Spec Protocol* e *Alert Protocol*.

Due concetti importanti in SSL sono la sessione SSL e la connessione SSL che sono definiti nel seguente modo:

-Connessione: una connessione è una fase di trasporto (nella definizione del modello a livelli OSI) che fornisce un determinato tipo di servizio. Per SSL. tali connessioni sono relazioni fra nodi paritari. Le connessioni sono transitorie ed ogni connessione è associata a una sola sessione.

-Sessione: una sessione SSL è un' associazione fra un client e un server. Le sessioni vengono create dal protocollo Handshake e definiscono un insieme di parametri di sicurezza crittografica che possono essere condivisi fra più connessioni. Le sessioni vengono utilizzate per evitare di svolgere la costosa negoziazione di nuovi parametri di sicurezza per ciascuna connessione.

Ogni coppia di parti può intrattenere più connessioni sicure. In teoria vi potrebbero anche essere più sessioni simultanee fra le parti ma in realtà questa funzionalità non viene utilizzata.

A ciascuna sessione vengono associati più stati. Una volta attivata una sessione. vi è uno stato operativo corrente per la lettura e la scrittura (ovvero per la ricezione e l'invio). Inoltre, durante il protocollo di *Handshake*. vengono creati degli stati provvisori di lettura e scrittura. Alla conclusione del protocollo di *Handshake*. gli stati provvisori diventano stati correnti.

Uno stato di sessione è definito dai seguenti parametri (definizioni tratte dalle specifiche SSL).:

- *Session identifier*: una sequenza di byte arbitraria scelta dal server per identificare lo stato di una sessione attiva o riattivabile.
- *Pee*r certificate: il certificato X509.v3 del nodo. Questo elemento può essere nullo.
- *Compression method*: l'algoritmo utilizzato per comprimere i dati prima della crittografia.
- *Cipher spec*: specifica l'algoritmo di crittografia dei dati (null, AES e così via) e l'algoritmo hash (come MD5 o SHA-I) utilizzato per il calcolo del codice MAC. Inoltre definisce gli attributi crittografici
- *Master secret*: un codice segreto di 48 byte condiviso dal client e dal server.

- *Is resumable*: un flag che indica se la sessione può essere utilizzata per iniziare nuove connessioni.

Lo stato di una connessione è definito dai parametri di seguito elencati:

- **Server and client:** sequenze di byte scelte dal server e dal client per ciascuna connessione.

- **Server write MAC secret:** la chiave segreta utilizzata nelle operazioni MAC per i dati inviati dal server.

- **Client write Mac secret:** la chiave segreta utilizzata nelle operazioni MAC sui dati inviati dal client.

- **Server write key:** la chiave di crittografia convenzionale per i dati crittografati dal server e decrittografati dal client.

- **Client write key:** la chiave di crittografia convenzionale per i dati crittografati dal client e decrittografati dal server.

- **Initialization vectors:** quando viene usata una cifratura a blocchi in modalità CBC, per ciascuna chiave viene mantenuto un vettore di inizializzazione (IV). Questo campo viene inizializzato dal protocollo *SSL Handshake*. Successivamente il blocco di testo cifrato finale di ciascun record viene preservato per essere utilizzato come vettore di inizializzazione del record seguente.

- **Sequence numbers:** ciascuna parte gestisce numeri sequenziali distinti per i messaggi trasmessi e ricevuti per ciascuna connessione. Quando una parte invia o riceve un messaggio di *change cipher spec* il numero di sequenza viene impostato a 0. I numeri di sequenza non devono superare il valore 264 - l.

4.3 Il protocollo SSL Record.

Il protocollo *SSL Record* fornisce due servizi per le connessioni SSL:

- Segretezza: il protocollo *Handshake* definisce una chiave segreta condivisa utilizzata per la crittografia convenzionale del *payload SSL*.

-Integrità del messaggio: il protocollo *Handshake* definisce anche una chiave segreta condivisa che viene utilizzata per creare un codice MAC.

Nel suo funzionamento generale, il protocollo SSL Record accetta il messaggio da trasmettere, frammenta i dati in blocchi di dimensioni appropriate, opzionalmente comprime i dati, applica un codice MAC, esegue la crittografia, aggiunge l’intestazione e

trasmette l'unità risultate in un segmento TCP. I dati ricevuti vengono decrittografati. verificati, decompressi, assemblati e quindi consegnati agli utenti di livello più elevato.

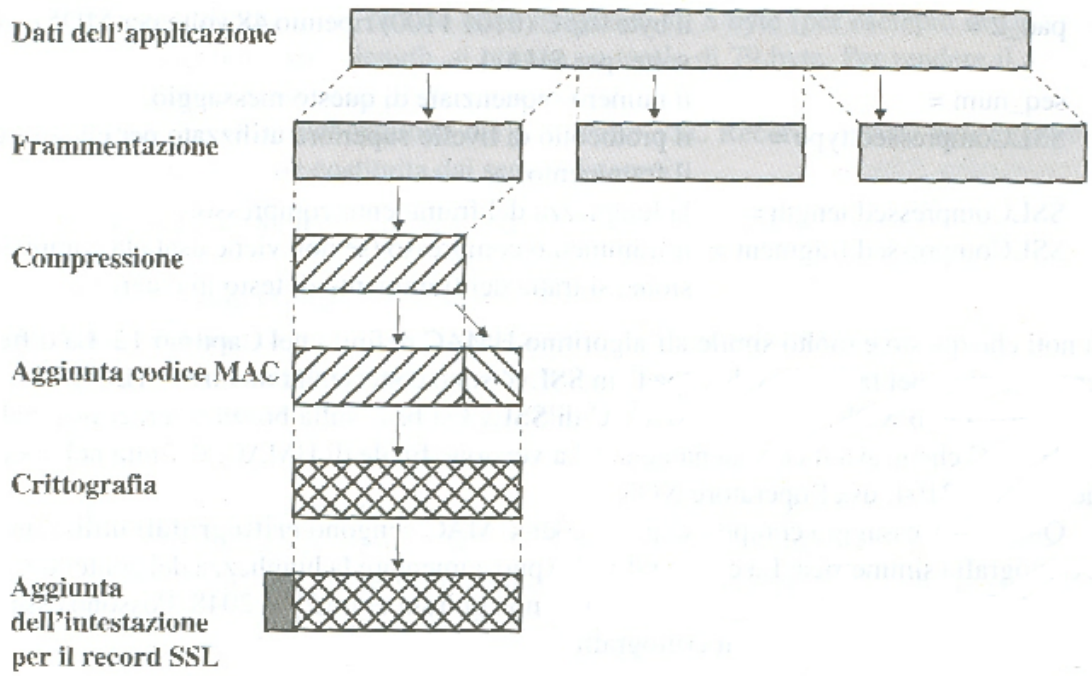


Figura 4.2: funzionamento del protocollo SSL record.

Il primo passo è la frammentazione. Ciascun messaggio dal livello superiore viene frammentato in blocchi di 2^{14} byte (16.384 byte) o meno. Poi viene eventualmente applicata la compressione. La compressione non deve perdere informazioni e non deve aumentare la lunghezza del contenuto di più di 1024 byte. In SSLv3 (e nella versione corrente di TLS) non è specificato alcun algoritmo di compressione e dunque l'algoritmo di compressione predefinito è nullo.

Il passo successivo consiste nel calcolo del codice MAC sui dati compressi, a tale scopo viene utilizzata una chiave segreta. Il calcolo è definito nel seguente modo:

hash(MAC_write_secret || pad_2 || hash(MAC_write_secret || pad_1 || seq_num || SSLCompressed.type || SSLCompressed.length || SSLCompressed.fragment)

in cui la notazione adoperata indica:

II	concatenamento
MAC_write_secret	chiave segreta condivisa
hash	algoritmo crittografico hash; MD5 o SHA -1

pad_1	il byte 0x36 (0011 0110) ripetuto 48 volte (384 bit) per MD5 e 40 volte (320 bit) per SHA-1
pad_2	il byte 0x5C (0101 1100) ripetuto 48 volte per MD5 e 40 volte per SHA-1
seq_num	il numero sequenziale di questo messaggio
SSLCompressed.type	il protocollo di livello superiore utilizzato per elaborare il frammento
SSLCompressed.lenght	la lunghezza del frammento compresso
SSLCompressed.fragment	il frammento compresso (se non viene usata la compressione si tratta del frammento di testo in chiaro)

Quindi il messaggio compresso più il codice MAC vengono crittografati utilizzando la crittografia simmetrica. La crittografia non può aumentare la lunghezza del contenuto di più di 1024 byte e dunque la lunghezza totale non può superare 2^{14} + 2048. Possono essere utilizzati i seguenti algoritmi di crittografia.

Crittografia a blocchi		Crittografia di flussi	
Algoritmo	Dimensione chiave	Algoritmo	Dimensione chiave
AES	128, 256	RC4-40	40
IDEA	128	RC4-128	128
RC2-40	40		
DES-40	40		
DES	56		
3DES	168		
Fortezza	80		

Per la crittografia di flussi vengono crittografati il messaggio compresso più il codice MAC. Si noti che il codice MAC viene calcolato prima della crittografia e poi viene crittografato insieme al testo in chiaro semplice o compresso.

Per la crittografia a blocchi, possono essere aggiunti dei pad dopo il codice MAC prima della crittografia. La parte di riempimento è costituita dal numero di byte di

riempimento seguiti da un byte che indica la lunghezza del riempimento. Il numero di byte di riempimento è il più piccolo valore tale che la dimensione totale dei dati da crittografare (testo in chiaro più codice MAC più riempimento) sia un multiplo della lunghezza del blocco di testo cifrato. Un esempio è un testo in chiaro (o testo compresso se viene utilizzata la compressione) di 58 byte con un codice MAC di 20 byte (utilizzando SHA-l) che viene crittografato utilizzando un blocco della lunghezza di 8 byte (per esempio in DES). Aggiungendo il byte *padding.length*, si ottiene un totale di 79 byte. Per rendere il totale un multiplo intero di 8, viene aggiunto un solo byte.

L'ultimo passo di elaborazione del protocollo SSL Record consiste nell'aggiunta di un'intestazione iniziale costituita dai seguenti campi.

Content Type (8 bit): il protocollo di livello superiore utilizzato per elaborare il frammento incluso.

Major Version (8 bit): la versione major di SSL in uso. Per SSLv3 il valore è 3.

Minor Version (8 bit): la versione minor in uso. Per SSLv3 il valore è 0.

Compressed Length (16 bit): la lunghezza in byte del frammento di testo in chiaro (o del frammento compresso se viene usata la compressione). Il valore massimo è 2^{14} + 2048.

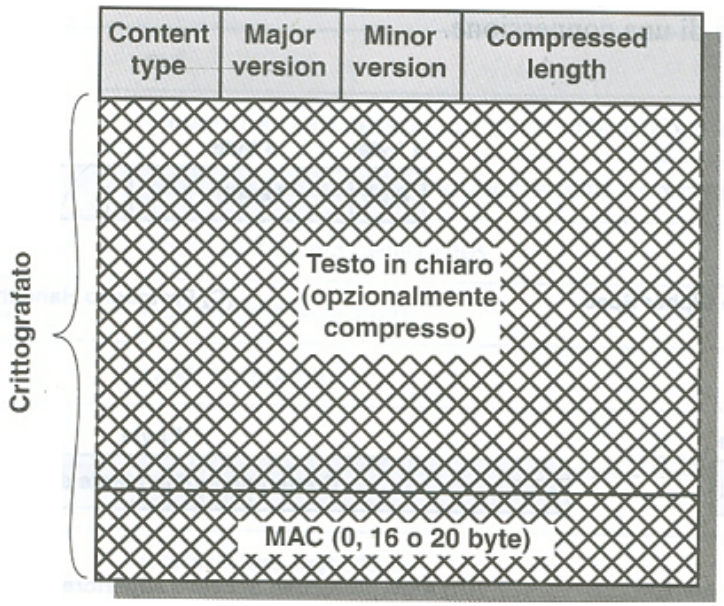


Figura 4.3: il formato del record SSL.

4.3.1 Il protocollo *Change Cipher Spec*.

Il protocollo *Change Cipher Spec* è uno dei tre protocolli specifici di SSL che utilizzano il protocollo *SSL Record*, ed è anche il più semplice. Questo protocollo è

costituito da un unico messaggio costituito da un unico byte contenente il valore l. (vedere la Figura 4.4A)

L'unico scopo di questo messaggio è quello di fare in modo che lo stato provvisorio venga copiato nello stato corrente, aggiornando e quindi attivando la cifratura che verrà utilizzata in questa connessione

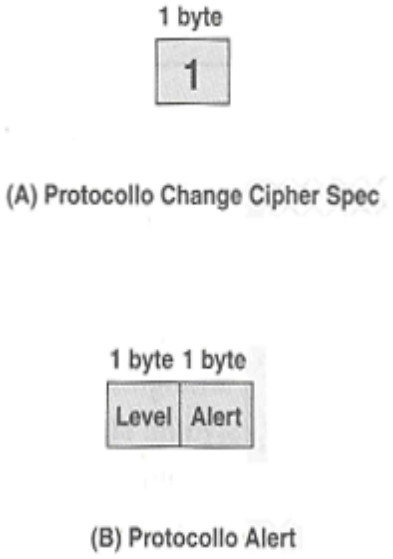


Figura 4.4: messaggi per i protocolli *Change Cipher Spec* (A) e Alert (B).

4.3.2 Il protocollo *Alert.*

Il protocollo *Alert* viene utilizzato per trasmettere allarmi SSL all'entità peer. Come per altre applicazioni che utilizzano SSL. i messaggi alert vengono compressi e crittografati, come specificato dallo stato corrente.

Ciascun messaggio di questo protocollo è costituito da due byte (Figura 4.4B). Il primo assume il valore *warning* (l) o *fatal* (2) per indicare la gravità del messaggio. Se il livello è *fatal*, SSL chiude immediatamente la connessione. Altre connessioni nella stessa sessione possono continuare ma non si possono attivare nuove connessioni in questa sessione. Il secondo byte contiene un codice che indica l' allarme specificato. Innanzitutto si elencano gli allarmi che sono sempre irreversibili (definizioni tratte dalle specifiche SSL).

- **unexpected_messag**e: è stato ricevuto un messaggio inappropriato.

- bad_record_mac: è stato ricevuto un codice MAC errato.

-decompression_failure: la funzione di decompressione ha ricevuto un input errato (per esempio incapacità di comprimere o decomprimere a una dimensione superiore alla massima consentita).

-handshakc_failure: il mittente non è stato in grado di negoziare un insieme di parametri di sicurezza accettabile date le opzioni disponibili.

-ilIegal_parameter: un campo in un messaggio di handshake era oltre i limiti consentiti o era incoerente rispetto agli altri campi.

Ecco gli altri allarmi.

-closc_notify: notifica al destinatario che il mittente non invierà altri messaggi in questa connessione. Ciascuna parte deve inviare un allarme close_notify prima di chiudere il lato di scrittura di una connessione.

. no_certificate: può essere inviato in risposta a una richiesta di certificato nel caso in cui non sia disponibile alcun certificato appropriato.

. **bad_certificate**: un certificato ricevuto era alterato (per esempio conteneva una firma che non poteva essere verificata).

. unsupported_certificate: il tipo del certificato ricevuto non è supportato.

. certificatc_revoked: il certificato è stato revocato dal suo firmatario.

. certificate_cxpired: il certificato è scaduto.

. certificatc_unknown: si è verificato un errore non specificato nell'elaborazione del certificato che lo rende inaccettabile.

4.3.3 Il protocollo *Handshake*.

La parte più complessa di SSL è il protocollo *Handshake*. Questo protocollo consente al server e al client di autentificarsi l'un l'altro e di negoziare un algoritmo di crittografia

e MAC, e le chiavi crittografiche da utilizzare per proteggere i dati inviati in un record SSL. Il protocollo *Handshake* viene utilizzato prima della trasmissione di qualsiasi dato dell'applicazione.

Il protocollo *Handshake* è costituito da una serie di messaggi scambiati dal client e dal server. Questi messaggi hanno il formato rappresentato nella Figura 4.5A. Ogni messaggio contiene tre campi.

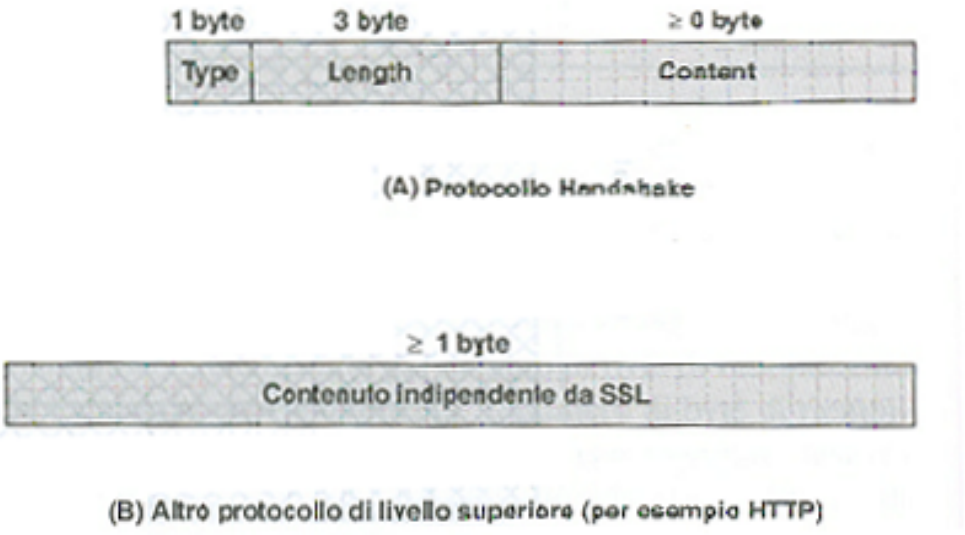


Figura 4.5: messaggi per i protocollo Handshake (A) e di livello superiore (B).

- **Type** (1 byte): indica un messaggio fra i dieci elencati nella seguente tabella:

Tipo messaggio	Parametri
hello_request	null
client_hello	version, random, session id, cipher suite, compression method
server_hello	version, random, session id, cipher suite, compression method
certificate	chain of X.509v3 certificates
server_key_exchange	parameters, signature
certificate_request	type, authorities
server_done	null
certificate_verify	signature
client_key_exchange	parameters, signature
finished	hash value

- **Length** (3 byte): la lunghezza del messaggio in byte.

- Content (>0 byte): i parametri associati a questo messaggio; anch'essi elencati nella tabella sopra riportata.

La Figura 4.6 mostra lo scambio iniziale necessario per stabilire una connessione logica fra il client e il server. Lo scambio si svolge in pratica in quattro fasi.

Fase 1. Inizializzazione delle funzionalità di sicurezza.

Questa fase viene utilizzata per avviare una connessione logica e per stabilire le funzionalità di sicurezza che possono essere impiegate. Lo scambio viene iniziato dal client che invia il messaggio *client_hello* con i seguenti parametri.

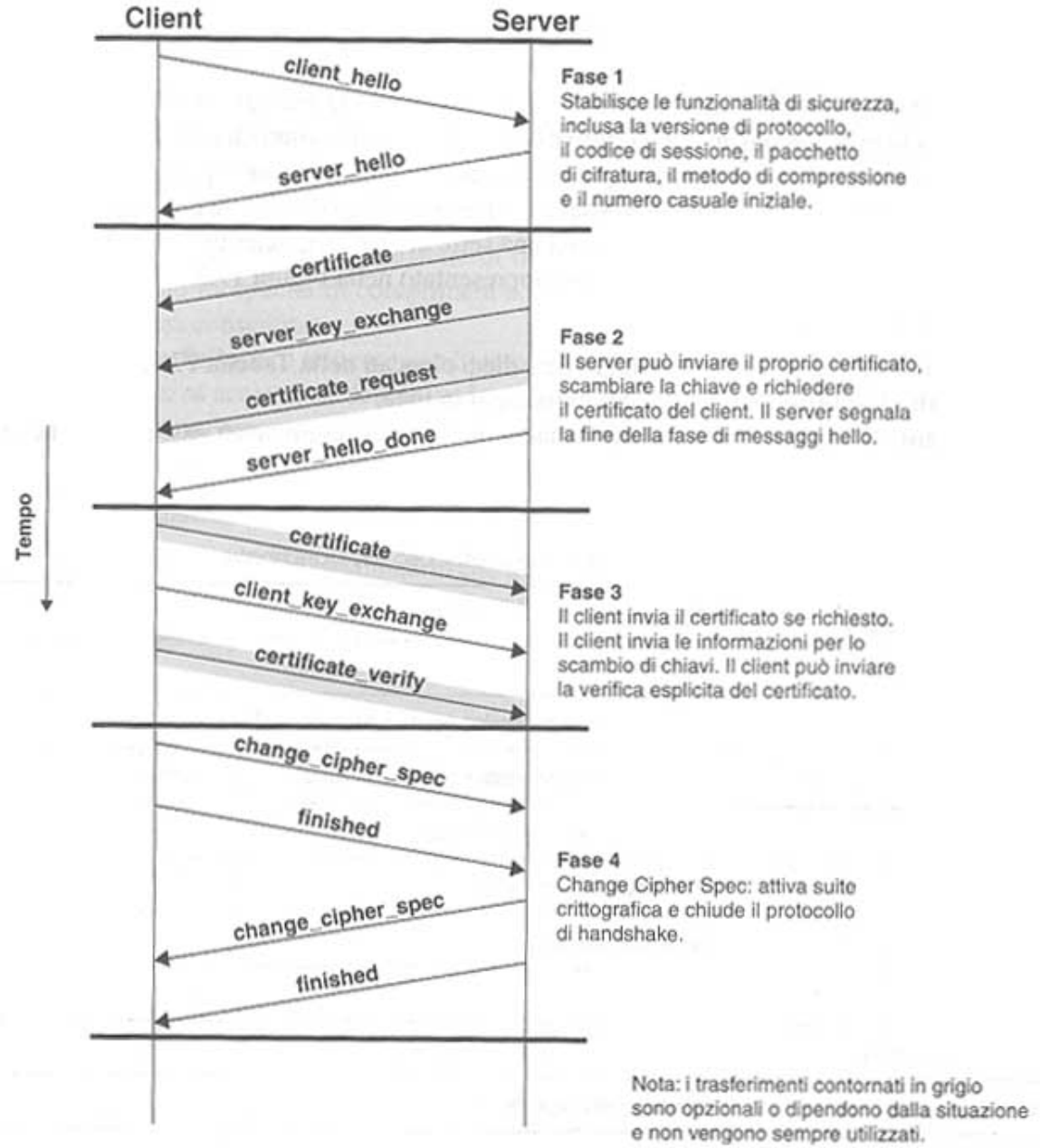


Figura 4.6: funzionamento del protocollo *Handshake.*

- **Version:** versione di SSL più elevata fra quelle utilizzabili dal client.

- **Random:** struttura casuale generata dal client costituita da un timestamp di 32 bit e 28 byte prodotti da un generatore sicuro di numeri casuali. Questi valori fungono da codice *nonce* e vengono utilizzati durante lo scambio delle chiavi per impedire gli attacchi a replay.

- **Session ID**: un identificatore di sessione di lunghezza variabile. Un valore diverso da zero indica che il client vuole aggiornare i parametri di una connessione esistente o creare una nuova connessione in questa sessione. Il valore 0 indica che il client vuole stabilire una nuova connessione in una nuova sessione.

- **CipherSuit**e: elenco che contiene gli algoritmi crittografici supportati dal client. in ordine decrescente di preferenza. Ciascun elemento della lista (ciascuna suite di crittografia) definisce sia un algoritmo di scambio delle chiavi che un *CipherSpec* (di cui si parlerà più avanti).

- **Compression method**: elenco dei metodi di compressione supportati dal client.

Dopo aver inviato il messaggio *client_hello*. il client attende il messaggio *server_hello* che contiene gli stessi parametri del proprio messaggio *client_hello*. Per il messaggio *server_hello* si applicano le seguenti convenzioni. Il campo *Version* contiene la più bassa fra la versione suggerita dal client e la versione più elevata supportata dal server. Il campo *Random* viene generato dal server ed è indipendente dal campo Random del client. Se il campo *Session ID* del client è diverso da zero. il server usa lo stesso valore, altrimenti il campo SessionID del server contiene il valore per una nuova sessione. Il campo *CipherSuite* contiene il pacchetto di cifratura selezionato dal server fra quelli proposti dal client. Il campo *Compression* contiene il metodo di compressione selezionato dal server fra quelli proposti dal client.

Il primo elemento del parametro *Cipher Suite* è il metodo per lo scambio delle chiavi (ovvero il metodo con cui vengono scambiate le chiavi crittografiche per la crittografia convenzionale e il codice MAC). Sono supportati i seguenti metodi per lo scambio delle chiavi.

RSA: la chiave segreta viene crittografata con la chiave pubblica RSA del destinatario. Si deve rendere disponibile un certificato di chiave pubblica per la chiave del destinatario.

Fixed Diffie-Hellman: questo è uno scambio della chiave Diffie-Hellman in cui il certificato del server contiene i parametri pubblici Diffie-Hellman firmati dall'autorità di certificazione. In pratica il certificato di chiave pubblica contiene i parametri della chiave pubblica Diffie-Hellman. Il client fornisce i propri parametri della chiave pubblica Diffie-Hellman in un certificato (se è richiesta l'autenticazione del client) oppure in un messaggio per lo scambio delle chiavi. Questo metodo produce come risultato una chiave segreta fissa fra i due nodi basata sul calcolo Diffie-Hellman utilizzando le chiavi pubbliche fisse.

Ephemeral Diffie.Hellman: questa tecnica viene uti1izzata per creare chiavi segrete effimere (temporanee, monouso). In questo caso. vengono scambiate le chiavi pubbliche Diffie-Hellman. Firmate utilizzando la chiave privata RSA o DSS del mittente. II destinatario può verificare la firma utilizzando la corrispondente chiave pubblica. I certificati vengono utilizzati per autenticare le chiavi pubbliche. Questa sembra la più sicura delle tre opzioni Diffie-Hellman in quanto produce una chiave temporanea autenticata.

Anonymous Diffie-Hellman: viene utilizzato l'algoritmo base Diffie-Hellman. senza autenticazione. Ovvero ciascuna parte invia all'altra parte i propri parametri pubblici Diffie-Hellman senza autenticazione. Questo approccio è vulnerabile ad attacchi *main-the-middle* in cui un estraneo esegue uno scambio anonimo Diffie-Hellman con entrambe le parti.

Dopo la definizione del metodo per lo scambio delle chiavi, si trova *CipherSpec*, che include i seguenti campi.

- **CipherAlgorithm:** uno qualsiasi degli algoritmi menzionati in precedenza: RC4, RC2, DES, 3DES, DES40, IDEA, Fortezza.
- **MACAlgorithm:** MD5 o SHA-I.
- **CipherType:** Stream o Block.
- **IsExportable:** True o False
- **HashSize:** 0,16 (per MD5) o 20 (per SHA-I) byte.
- **Key material:** una sequenza di byte che contiene i dati utilizzati per la generazione delle chiavi di scrittura.

- **IV Size:** le dimensioni del vettore di inizializzazione per la crittografia CBC (*Cipher Block Chaining*).

Fase 2. Autenticazione del server e scambio delle chiavi.

Il server inizia questa fase inviando il proprio certificato, sempre che debba essere autenticato; il messaggio contiene un certificato X.509 oppure una catena di certificati. Il messaggio *certificate* è obbligatorio per ogni metodo concordato per lo scambio di chiavi tranne che per lo scambio Diffie-Hellman anonimo. Si noti che se viene utilizzato lo scambio fiXed Diffie-Hellman, questo messaggio *certificate* funge da messaggio per lo scambio delle chiavi del server poiché contiene i parametri Diffie-Hcllman pubblici del server.

Poi, se necessario, può essere inviato un messaggio *server_key_exchange*. Questo non è necessario in due casi: quando il server ha inviato un certificato con i parametri fiXed Diffie-Hellman o quando deve essere utilizzato lo scambio delle chiavi RSA. Il messaggio *server_key_exchange* è obbligatorio nei seguenti casi.

- Anonymous Diffie-Hellman: il contenuto del messaggio è costituito dai due valori globali Diffie-Hellman (un numero primo e una sua radice primitiva) più la chiave Diffie-Hellman pubblica del server.

-Ephemeral Diflie-Hellman: il contenuto del messaggio include i tre parametri DiffieHcllman necessari per la forma Diffie-Hellman anonima più una firma di questi parametri.

-Scambio delle chiavi RSA in cui il server usa RSA ma ha una chiave RSA di sola firma: in questo caso il client non può semplicemente inviare una chiave segreta crittografata con la chiave pubblica del server. Al contrario il server deve creare una coppia temporanea di chiavi RSA pubblica/privata e utilizzare il messaggio server key_exchange per inviare la chiave pubblica. Il contenuto del messaggio include i due parametri della chiave pubblica temporanea RSA (esponente e modulo), più una firma di questi parametri.

Fortezza: occorre fornire ulteriori dettagli sulle firme. Come di consueto, una firma viene creata prendendo il valore hash di un messaggio e crittografandolo con la chiave privata del mittente. In questo caso il codice hash è definito nel seguente modo:

hash(ClientHello.random || ServerHello.random || ServerParams)

Dunque il codice hash non copre solo i parametri Diffie-Hellman o RSA ma anche i due *nonce* dei messaggi *hello* iniziali. Questo evita gli attacchi a replay. Nel caso di una firma DSS, il calcolo hash viene eseguito utilizzando l'algoritmo SHA-I. Nel caso di una firma RSA, vengono calcolati entrambi i valori hash MD5 e SHA-I e il concatenamento di questi due valori hash (36 byte) viene crittografato utilizzando la chiave privata del server.

Quindi un server non anonimo (un server che non utilizza la forma Diffie-Hellman anonima) può richiedere un certificato al client. Il messaggio *certificate_request* include due parametri: *certificate_type* e *certificate_authorities.* Il *certificate_type* indica l'algoritmo a chiave pubblica e il suo impiego.

RSA, solo firma.

DSS, solo firma.

RSA per fiXed Diffie-Hellman; in questo caso la firma viene utilizzata solo per l'autenticazione inviando un certificato firmato con RSA.

DSS per fiXed Diffie-Hellman; usato anch'esso solo per l'autenticazione.

RSA per Ephemeral Diffie-Hellman.

DSS per Ephemeral Diffie-Hellman.

Fortezza.

Il secondo parametro del messaggio *certificate_request* è un elenco delle autorità di certificazione accettate.

L'ultimo messaggio della Fase 2 è obbligatorio ed è il messaggio *server_done* che viene inviato dal server per indicare la fine dei messaggi di *hello* del server. Dopo aver inviato questo messaggio, il server attende una risposta del client. Questo messaggio non ha alcun parametro.

Fase 3. Autenticazione del client e scambio delle chiavi.

Dopo aver ricevuto il messaggio *server_done,* il client deve verificare che il server abbia fornito un certificato valido (se richiesto) e deve controllare che i parametri di *server_hello* siano accettabili. Se tutto è soddisfacente. il client invia al server uno o più messaggi.

Se il server ha richiesto un certificato, il client inizia questa fase inviando un messaggio certificate. Se non è disponibile alcun certificato adatto, il client invia l'allarme *no*_certificate.

Poi viene il messaggio *client_key_exchange* che deve essere inviato in questa fase. Il contenuto del messaggio dipende dal tipo di scambio delle chiavi.

-RSA: il cliente genera un valore segreto pre-master di 48 byte e ne esegue la crittografia con la chiave pubblica tratta dal certificato del server o con la chiave RSA temporanea tratta da un messaggio *server_key_exchange*. Il suo uso per calcolare il codice segreto master verrà descritto più avanti.

-Ephemeral oAnonymous Diffie-Hellman: vengono inviati i parametri pubblici Diffie-Hellman del client

- **Fixed Diffie-Hellman**: i parametri Diffie-Hellman pubblici del client sono stati precedentemente inviati in un messaggio certificate e dunque il contenuto di questo messaggio è nullo.
- **Fortezza**: vengono inviati i parametri Fortezza del client.

Infine. in questa fase. il client può inviare un messaggio *certificate_verify* per fornire una verifica esplicita di un certificato del client. Questo messaggio viene inviato solo dopo un certificato del client che ha funzionalità di firma (ovvero tutti i certificati tranne quelli contenenti parametri fixed Diffie-Hellman). Questo messaggio firma un codice hash sulla base dei messaggi precedenti, come indicato di seguito:

Certificate Verify.signature.md5_hash

MD5(master_secret || pad_2 || MD5(handshake_messages || master_secret || pad_l);

Certificate.signature.sha_hash

SHA(master_secret || pad_2 || SHA(handshake_messages || master-secret Il pad_l);

dove pad_1 e pad_2 sono i valori definiti in precedenza per il codice MAC, *handshake_messages* fa riferimento a tutti i messaggi del protocollo *Handshake* inviati o ricevuti a partire da *client_hello* (escluso questo messaggio) e *master_secret* è il codice segreto calcolato la cui costruzione verrà descritta più avanti in questa parte del capitolo.

Se la chiave privata dell'utente è DSS, viene utilizzata per crittografare il codice hash SHA-l. Se la chiave privata dell'utente è RSA, viene utilizzata per crittografare il concatenamento dei codici hash MD5 e SHA-I. In entrambi i casi lo scopo è quello di verificare la proprietà della chiave privata da parte del client. Anche se qualcuno dovesse usare fraudolentemente il certificato del client, non sarebbe in grado di inviare questo messaggio.

Fase 4. Fine

Questa fase completa l'impostazione di una connessione sicura. Il client invia un messaggio *change_cipher_spec* e copia il *CipherSpec* temporaneo nel *CipherSpec* corrente. Si noti che questo messaggio non è parte del protocollo *Handshake* ma viene inviato utilizzando il protocollo *Change Cipher Spec*. Il client invia immediatamente il messaggio *finished* con i nuovi algoritmi, chiavi e "segreti". Il messaggio *finished* verifica che i processi di scambio delle chiavi e di autenticazione abbiano avuto successo. Il contenuto del messaggio *finished* è il concatenamento dei due valori hash:

MD5(master_secret || pad2 || MD5(handshake_messages || Sender || master-secret || padl)
SHA(master_secret || pad2 || SHA(handshake_messages || Sender || master_secret || padl)

dove *Sender* è un codice che identifica che il mittente è il client e *handshake_messages* è costituito da tutti i dati di tutti i messaggi di *handshake* fino a questo messaggio escluso.

In risposta a questi due messaggi, il server invia il proprio messaggio change_cipher_spec. trasferisce il *CipherSpec* provvisorio nel *CipherSpec* corrente e invia il proprio messaggio *finished*. A questo punto la procedura di *handshake* è completa e il client e il server possono iniziare a scambiarsi i dati a livello applicazioni.

4.4 Calcoli crittografici.

Altri due argomenti importanti: sono la creazione di un valore segreto master condiviso tramite lo scambio di chiavi e la generazione dei relativi parametri crittografici.

4.4.1 La creazione del valore segreto master.

Il valore segreto master condiviso è un valore monouso di 48 byte (384 bit) generato per una sessione tramite scambio di chiavi sicuro. La creazione si svolge in due fasi. Innanzitutto viene scambiato il valore *pre_master_secret*. Poi viene calcolato il valore *master_secret* da entrambe le parti. Per lo scambio del valore *pre_master_secret* vi sono due possibilità.

- **RSA**: il client genera un valore pre_master_sccret di 48 byte, crittografato con la chiave RSA pubblica del server e la invia al server. Il server esegue la decrittografia del testo cifrato utilizzando la propria chiave privata per recuperare il valore *pre_master_secret*.

- **diffie-Hellman:** sia il client che il server generano una chiave pubblica Diffie-Hellman.

Dopo aver scambiato queste chiavi, ognuno dei due svolge il calcolo Diffie-Hellman per creare il valore *pre_master_secret*.

Entrambi i lati calcolano il valore *master_secret* nel seguente modo:

master_secret = MD5(pre_master_secret || SHA('A' || pre_master_secret ||
ClientHello.random || ServerHello.random) ||
MD5(pre_master_secret || SHA('BB' || pre_master_secret ||
ClientHello.random || ServerHello.random)) ||
MD5(pre_master_secret || SHA('CCC' || pre_master_secret ||
ClientHello.random || ServerHello.random))

dove *ClientHello.random* e *ServerHello.random* sono i due valori *nonce* scambiati nei messaggi *hello* iniziali.

4.4.2 Generazione dei parametri crittografici.

CipherSpecs richiede un “segreto” MAC di scrittura per il client, un “segreto” MAC di scrittura per il server, una chiave di scrittura per il client, una chiave di scrittura per il server, un vettore di inizializzazione di scrittura per il client e un vettore di inizializzazione di scrittura per il server che vengono generati dal valore segreto master

in questo ordine. Questi parametri vengono generati dal valore segreto master tramite calcoli hash che producono parametri di lunghezza appropriata.

La generazione del materiale per la chiave dal valore segreto master utilizza lo stesso formato per la generazione del valore segreto master a partire dal pre-master:

key_block = MD5(master_secret || SHA('A' || master_secret ||
ServerHello.random || ClientHello.random) ||
MD5(master_secret || SHA('BB' || master_secret ||
ServerHello.random || ClientHello.random) ||
MD5(master_secret || SHA('CCC' || master_secret ||
ServerHello.random || ClientHello.random) || . . .

fino a generare un output di dimensioni sufficienti. Il risultato di questa struttura algoritmica è una funzione pseudocasuale. Si può considerare il valore *master_secret* come il seme pseudocasuale per questa funzione. I numeri casuali del client e del server servono per complicare l'analisi crittografica.

4.5 *Transport Layer Security*.

TLS è un'iniziativa di standardizzazione di IETF il cui obiettivo è di produrre una versione standard per Internet di SSL. TLS è stata definita come *Proposed Internet Standard* nel documento RFC 2246 che è molto simile a SSLv3. In questa parte del capitolo si evidenzieranno le differenze.

4.5.1 Numero di versione.

Il valore *TLS Record Format* coincide con il valore *SSL Record Format* (Figura 4.3) e i campi dell'intestazione hanno lo stesso significato. L'unica differenza è nei valori della versione. Per la versione corrente di TLS: *Major Version* è 3 e *Minor Version* è l.

4.5.2 Codice MAC (*Message Authentication Code*).

Vi sono due differenze fra gli schemi MAC di SSLv3 e TLS: l'algoritmo utilizzato e il tipo di calcolo MAC. TLS utilizza l'algoritmo HMAC definito nel documento RFC 2104. HMAC è definito nel seguente modo:

$$\mathbf{HMAC_K(M)=H[(K^+ + opad) \,||\, H[(K^+ + ipad) \,||\, M]]}$$

dove:

H	funzione hash incorporata (per TLS si tratta di MD5 o SHA-I)
M	messaggio di input per HMAC
K^+	chiave segreta riempita di valori "0" sulla sinistra in modo che il risultato sia uguale alla lunghezza di blocco del codice hash (per MD5 e SHA-I, la lunghezza di blocco è di 512 bit)
Ipad	00110110 (36 in esadecimale) ripetuto per 64 volte (512 bit)
Opad	01011100 (5C in esadecimale) ripetuto per 64 volte (512 bit)

SSLv3 utilizza lo stesso algoritmo. tranne per il fatto che i byte di riempimento sono concatenati con la chiave segreta (invece di eseguire uno XOR con la chiave segreta) e riempiti fino a raggiungere la lunghezza del blocco. Il livello di sicurezza dovrebbe essere simile in entrambi i casi.

Per TLS il calcolo MAC prevede i campi indicati nella seguente espressione:

HMAC_hash(MAC_write_secret, seq_num || TLSCompressed.type || TLSCompressed.version || TLSCompressed.length || TLSCompressed.fragment)

Il calcolo MAC copre tutti i campi coperti dal calcolo SSLv3 più il campo *TLSCompressed.version* che è la versione del protocollo impiegato.

4.5.3 Funzione pseudo casuale

TLS utilizza una funzione pseudocasuale chiamata PRF che espande i valori segreti in blocchi di dati per la generazione o la convalida della chiave. L'obiettivo è quello di utilizzare un valore segreto condiviso relativamente compatto per generare lunghi blocchi di dati sicuri da ogni genere di attacco alle funzioni hash e i ai codici MAC. La funzione PRF si basa sulla seguente funzione di espansione dei dati:

P_hash(secret, seed) = HMAC_hash(secret, A(l) || seed) ||
HMAC_hash(secret, A(2) || seed) ||
HMAC_hash(secret, A(3) || seed) ||…

dove A() è definito come:

A(0) seme

A(*i*) HMAC_has(secret,A(*i*-1))

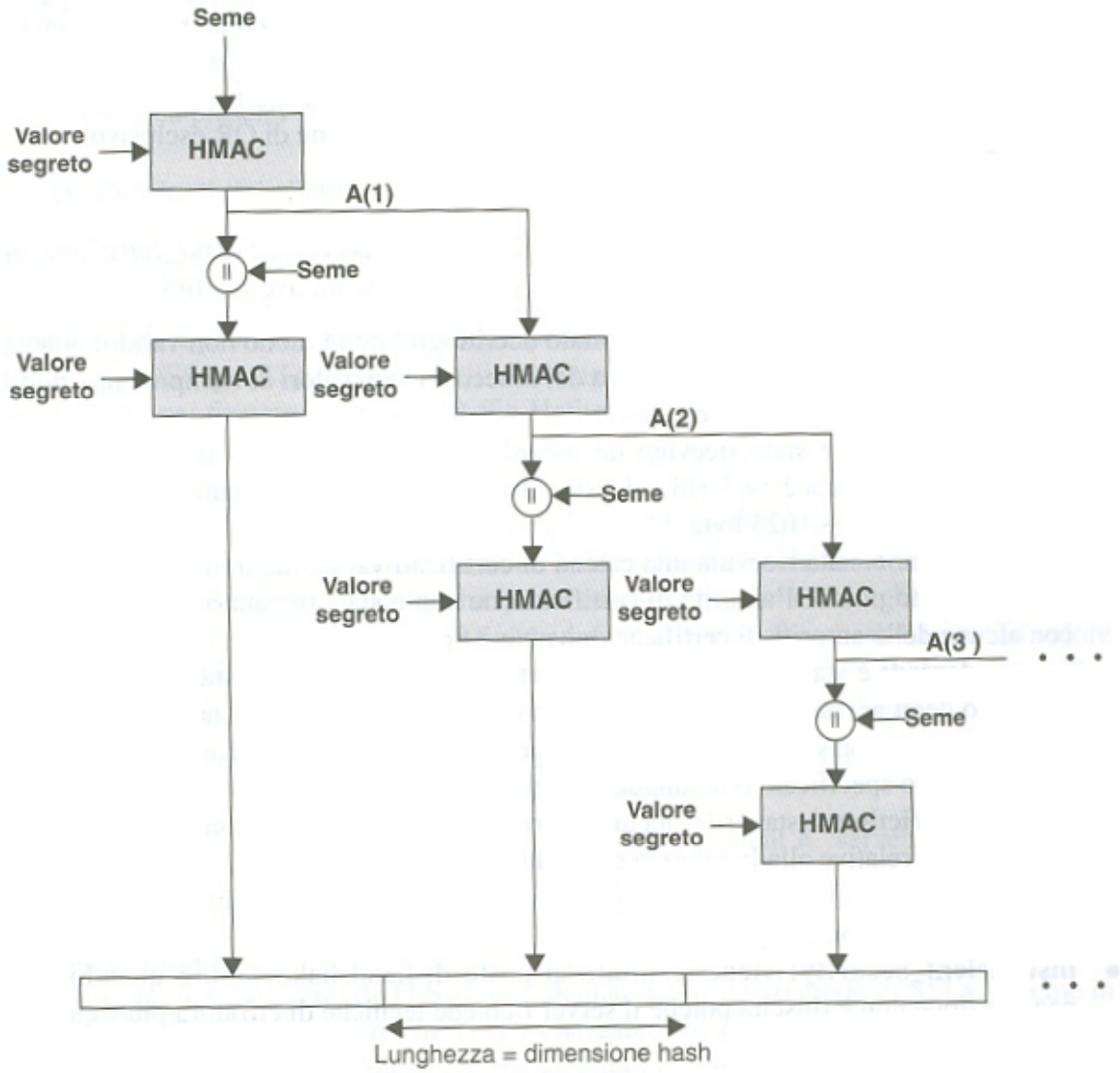


Figura 4.6: La funzione TLS P_hash (valore segreto, seme).

La funzione di espansione dei dati utilizza l'algoritmo HMAC con MD5 o SHA-I come funzione hash di base. Come si può vedere, P_hash può essere iterata tutte le volte che è necessario per produrre la quantità desiderata di dati. Per esempio, se P_SHA-l dovrebbe essere utilizzato per generare 64 byte di dati dovrà essere iterato per quattro volte producendo 80 byte di dati, dei quali ne verranno eliminati 16. In questo caso. anche P_MD5 dovrà essere iterato quattro volte producendo esattamente 64 byte di dati.

Si noti che ciascuna iterazione prevede due esecuzioni di HMAC. ognuna delle quali a sua volta prevede due esecuzioni dell'algoritmo hash sottostante.

Per rendere PRF più sicura possibile. la funzione utilizza due algoritmi hash in modo da garantire la sicurezza, sempre che uno dei due algoritmi rimanga sicuro. PRF è definita come:

PRF(secret, label, seed)= P _MD5(S1. label || seed) + P _SHA-1 (S2, label ||seed)

PRF prende come input un valore segreto. un'etichetta di identificazione e un seme e produce un output di lunghezza arbitraria. L'output viene creato suddividendo il valore segreto in due parti (S1 e S2) ed eseguendo P_hash su ciascuna metà, utilizzando MD5 su una e SHA-1 sull'altra. Ai due risultati viene applicato un OR esclusivo per produrre l'output; per questo scopo. P_MD5 dovrà generalmente essere iterato più volte rispetto a P_SHA-1 per produrre una quantità di dati uguale per l'input della funzione di OR esclusivo.

4.5.4 Codici di allarme.

TLS supporta tutti i codici di allarme definiti in SSLv3 con l'eccezione di *no_certificate*, ai quali ne aggiunge alcuni altri. fra cui i seguenti che sono sempre irreversibili.

decryption_failed: un testo cifrato è stato decrittografato in modo non valido (poteva non essere un multiplo della lunghezza del blocco o i suoi valori di riempimento, dopo essere stati controllati. erano errati).

record_overflow: è stato ricevuto un record TLS con un payload (testo cifrato) di lunghezza superiore a 2^{14} + 2048 o il testo cifrato è stato decrittografato a una lunghezza maggiore di 2^{14}+ 1024 byte.

unknown_ca: è stata ricevuta una catena di certificati valida ma un certificato non è stato accettato poiché l'autorità di certificazione non è stata trovata o non coincideva con alcune delle autorità di certificazione note e fidate.

access_denied: è stato ricevuto un certificato valido ma quando è stato richiesto o il controllo degli accessi. il mittente ha deciso di non procedere con la negoziazione.

decode_error: non si è potuto decodificare un messaggio poiché un campo fuoriusciva.

decrypt_error: un'operazione crittografica di handshake è fallita. Per esempio non è stata in grado di verificare una firma, decrittografare uno scambio di chiavi o convalidare un messaggio terminato.

user_canceled: questo handshake è stato annullato per motivi differenti da un errore di protocollo.

no_renegotiation: inviato da un client in risposta a una richiesta hello o dal server in risposta a un hello del client dopo l'handshake iniziale. Entrambi questi messaggi produrrebbero normalmente una rinegoziazione ma questo allarme indica che il mittente non è stato in grado di rinegoziare. Questo messaggio è sempre un warning. dall'intervallo specificato o la lunghezza del messaggio era errata.

.**export_restriction:** è stata rilevata una negoziazione non coerente con le restrizioni di esportazione relative alla lunghezza delle chiavi.

protocol_version: la versione di protocollo che il client ha tentato di negoziare viene riconosciuta ma non è supportata.

insufficient_security: viene restituito al posto di *handshake_failure* quando una negoziazione non è riuscita poiché il server richiede tecniche di cifratura più sicure di quelle supportate dal client.

internal_error: un errore interno indipendente dal nodo peer oppure dal protocollo rende impossibile continuare.

Ecco i nuovi allarmi rimanenti.

decript_error: un'operazione crittografica di *handshake* è fallita, per esempio non è stata in grado di verificare una firma, decrittografare uno scambio di chiavi o convalidare un messaggio terminato.

user_canceled: questo handshake è stato annullato per motivi differenti da un errore di protocollo.

no_renegotiation: inviato da un client in risposta a una richiesta hello o dal server in risposta a un hello del client dopo l'handshake iniziale. Entrambi questi messaggi produrrebbero normalmente una rinegoziazione ma questo allarme indica che il mittente non è stato in grado di rinegoziare. Questo messaggio è sempre un warning.

Suite crittografiche.

Vi sono alcune differenze fra le suite crittografiche disponibili sotto SSLv3 e TLS.

-Scambio delle chiavi: TLS supporta tutte le tecniche di scambio delle chiavi di SSLv3 ad eccezione di Fortezza.

Algoritmi di crittografia simmetrici: TLS include tutti gli algoritmi di crittografia simmetrica presenti in SSLv3, ad eccezione di Fortezza.

Tipi di certificati del client

TLS definisce i seguenti tipi di certificati che possono essere richiesti in un messaggio certificate_request: rsa_sign, dss_sign, rsa_fixed_dh e dss_fixed_dh. Questi sono definiti anche in SSLv3. Inoltre, SSLv3 include rsa_ephemeral_dh, dss_ephemeral_dh e fortezza_kea. Ephemeral Diffie-Hellman richiede la firma dei parametri Diffie-Hellman con RSA o DSS; per TLS vengono utilizzati i tipi rsa_sign e dss_sign; non è necessario un tipo distinto per firmare i parametri Diffie-Hellman. TLS non include lo schema Fortezza.

Messaggi *certificate_verify* e *finished*

Nel messaggio TLS *certificate_verify*, i codici hash MD5 e SHA-I vengono calcolati solo su *handshake_messages*. In SSLv3, i calcoli hash includono anche il valore segreto master e i bit di riempimento. Si è ritenuto che questi campi aggiuntivi non migliorassero la sicurezza. Come in SSLv3. il messaggio *finished* di TLS è un codice hash basato sul valore *master_secret* condiviso, i messaggi precedenti di *handshake* e un'etichetta che identifica il client o il server. Il calcolo è però leggermente differente, in TLS si ha:

PRF(master_secret, finished_label, MD5(handshake_messages) ||
SHA-1 (handshake_messages))

dove *finished_label* è la stringa "*client finished*" per il client e "*server finished*" per il server.

Calcoli crittografici

Il valore *pre_master_secret* di TLS viene calcolato nello stesso modo di SSLv3.

Come in SSLv3, il valore *master_secret* in TLS viene calcolato come funzione hash del valore *pre_master_secret* e dei due numeri casuali di *hello*. La forma del calcolo TLS è differente da quella di SSLv3 ed è definita nel seguente modo:

mastersecret =

PRF(pre_master_secret, "master secret", ClientHello.random || ServerHello.random)

L'algoritmo viene eseguito finché non vengono prodotti 48 byte di output pseudocasuale.

Il calcolo del *key-block* (chiavi segrete MAC, chiavi di crittografia della sessione e vettori di inizializzazione) è definito come segue:

key _block =

PRF(master_secret, “key expansion",

SecurityParameters.server _random || Security Parameters.client_random)

fino a produrre un output di lunghezza sufficiente. Come in SSLv3, *key_block* è funzione di *master_secret* e dei numeri casuali del client e del server ma in TLS l'algoritmo impiegato è differente.

Riempimento.

In SSL, i byte di riempimento aggiunti prima della crittografia dei dati utente sono la quantità minima necessaria per fare in modo che la dimensione dei dati da crittografare sia multipla della lunghezza del blocco dell'algoritmo di crittografia. In TLS i byte di riempimento possono essere di qualsiasi dimensione che produca un totale multiplo della lunghezza del blocco dell'algoritmo di crittografia, fino a un massimo di 255 byte. Per esempio, se il testo in chiaro (o il testo compresso, nel caso venga impiegata la compressione) più il codice MAC, più il byte *padding.length* fosse di 79 byte, allora i byte di riempimento in byte potrebbero essere l, 9, 17 e così via fino a 249. Questa variabilità viene utilizzata per vanificare gli attacchi basati sull'analisi della lunghezza dei messaggi scambiati.

CAPITOLO 5

Quanto descritto nei precedenti capitoli risulta propedeutico e di basilare importanza per la creazione di una rete AIS di gerarchia superiore che si occupi di raccogliere i dati AIS da tutti gli ASM delle nazioni costiere.

Lo scopo di creare una rete AIS di portata regionale, per avere un quadro del traffico navale in tutto il Mar Mediterraneo, non è stato immediatamente raggiungibile, ma sono servite una serie di valutazioni per capire come poter realizzare una raccolta dati dai diversi ASM nazionali, non tutti standardizzati e molti dei quali con differenti tecnologie di funzionamento.

Lo schema logico da implementare è quello riportato nella figura seguente, per ottenerlo ho portato avanti una approfondita fase di studio sulle struttura delle singole reti AIS degli Stati partecipanti al progetto, ovvero Portogallo, Spagna; Malta, Cipro, Francia, Slovenia, Croazia, Grecia, Bulgaria, Romania ed, ovviamente, Italia con il ruolo di nazione coordinatrice e realizzatrice del sistema in parola.

Per ovvi motivi di segretezza non sarà trattata la descrizione dettagliata dei singoli sistemi studiati, verrà invece presentata la soluzione tecnologica che ho proposto e poi realizzato, che ha distanza di un anno si sta mostrando efficace, efficiente e robusta.

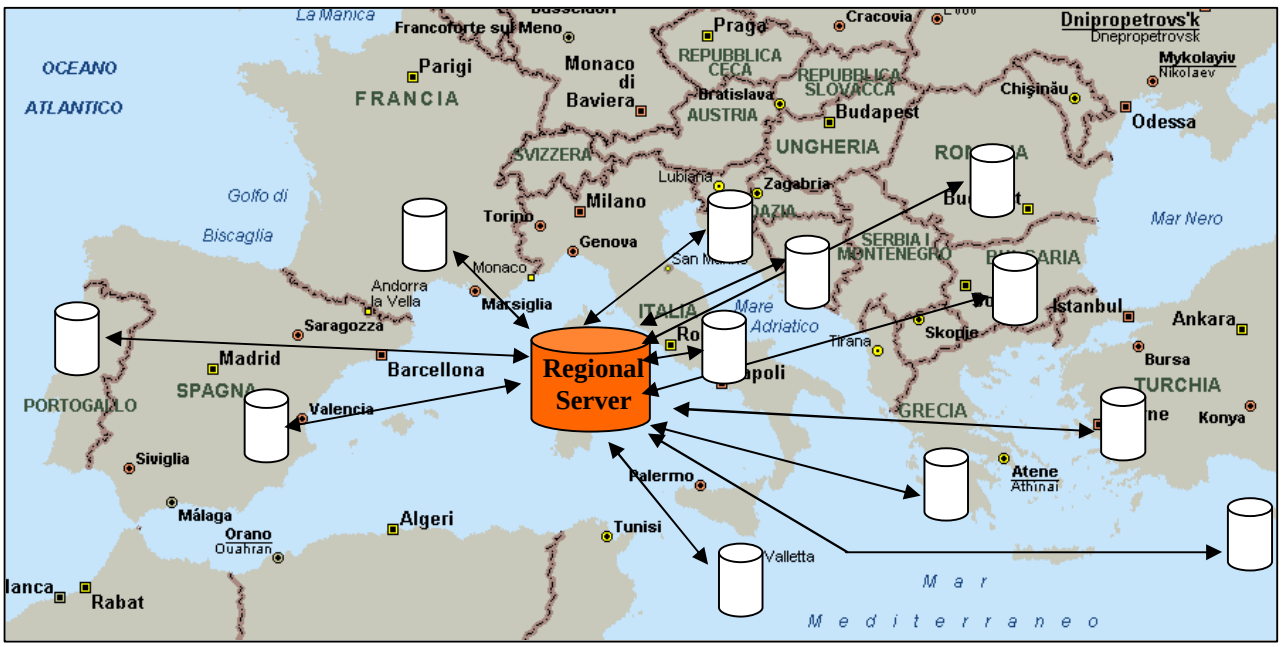


Figura 5.1: schema logico del sistema AIS regionale del Mediterraneo.

Il sistema realizzato, inizialmente conosciuto come “AIS del Mediterraneo”, è stato recentemente battezzato con il nome di **MARES,** acronimo di *Mediterranean AIS Regional Exchange System*, il cui l’ultima lettera viene rappresentata secondo la notazione greca per richiamarne il significato matematico di “sommatoria”, questo per indicare che il risultato ottenuto è stato raggiunto grazie allo sforzo congiunto di tutti partecipanti al progetto.

5.1 Architettura MARES.

L’architettura studiata basa il proprio funzionamento su due moduli distinti. Il primo, denominato **Server Regionale**, comprende moduli software installati su uno o più server posizionati nel centro di raccolta regionale posizionato presso il Comando Generale del Corpo delle capitanerie di porto a Roma. Le funzioni principali che questi deve garantire sono la raccolta e distribuzione dei dati AIS da/verso gli Stati Membri, la memorizzazione dei dati AIS nel database regionale, il recupero ed analisi dei dati memorizzati con presentazione (via WEB) degli stessi, correlate con informazioni cartografiche e di posizionamento (GIS).

Il secondo modulo, denominato ***Proxy***, include tutti i moduli software in grado di abilitare lo scambio dati AIS tra un server nazionale ASM ed il server regionale, gestire le connessioni TCP/IP verso il server nazionale ed SSL verso il server regionale, ed infine eseguire operazioni di filtraggio sui dati provenienti dal sistema AIS nazionale.

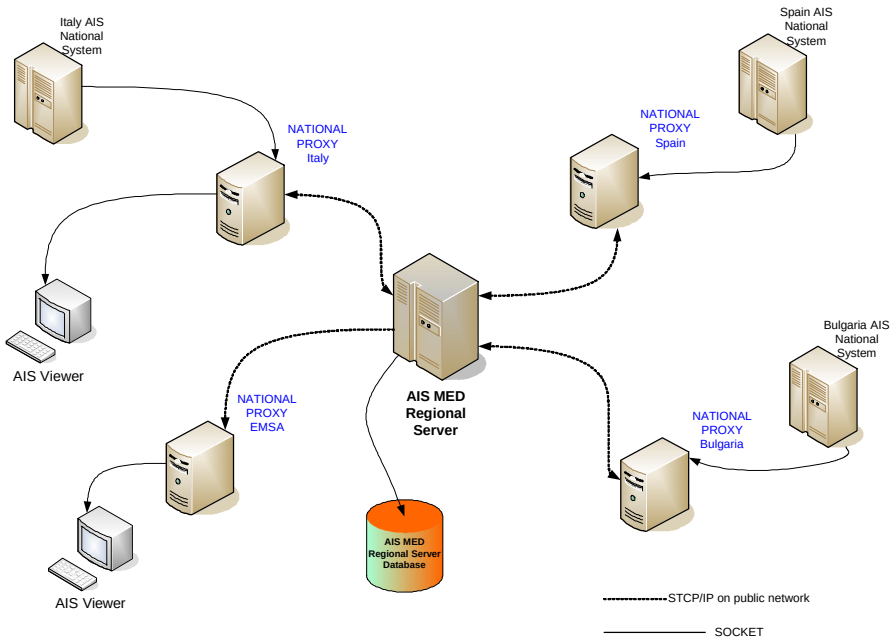


Figura 5.2: architettura del sistema AIS regionale del Mediterraneo.

Come si vede dallo figura 5.2, per realizzare l'architettura di scambio dati esistono due tipi di connessioni, da un lato vi è uno scambio dati tramite ***socket*** (indirizzo IP e porta TCP di collegamento) messe a disposizione dai singoli server nazionali in ascolto delle "chiamate" da parte del proxy.

Dall'altro lato del proxy vi è uno scambio dati messo in piedi mediante connessioni TCP/IP sicure che permettono il collegamento del proxy al Server Regionale.

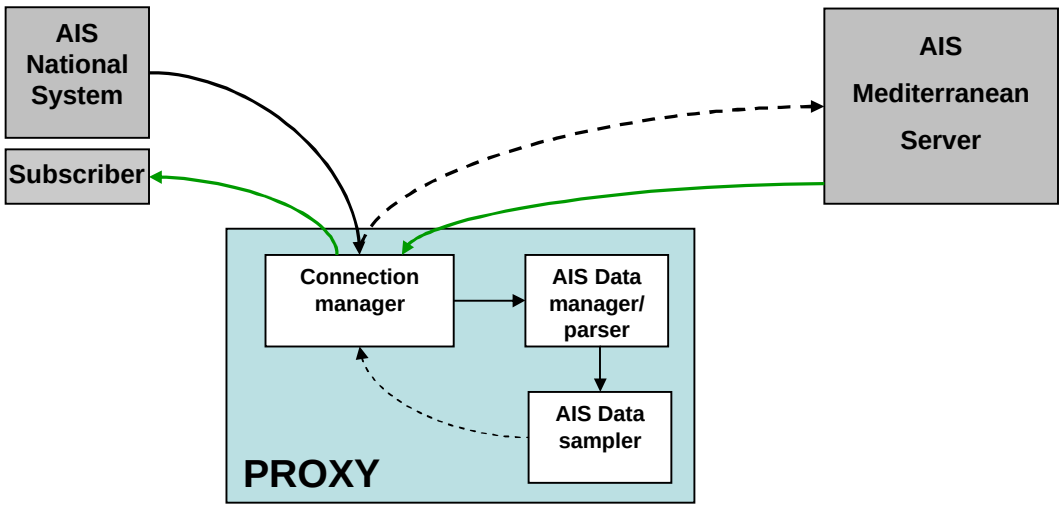


Figura 5.3: schema logico connessioni Proxy.

5.2 *Proxy*.

Il proxy è un programma che si interpone tra un il Server Regionale, che riveste il ruolo client, ed i vari server nazionali. Il client si collega al proxy invece che al server, il *proxy* inoltra le richieste e le risposte dall'uno all'altro. Nello specifico ogni proxy, uno per Stato partecipante, preleva le stringhe AIS grezze dal rispettivo server nazionale, ma contemporaneamente riceve dal Server Regionale i dati AIS che questi riceve anche da tutti gli altri partecipanti. Il proxy mette a disposizione tali informazioni tramite una serie di *subscriber line* a cui è possibile collegare degli AIS *viewer* per visualizzare graficamente i dati AIS scambiati tra tutti i Paesi.

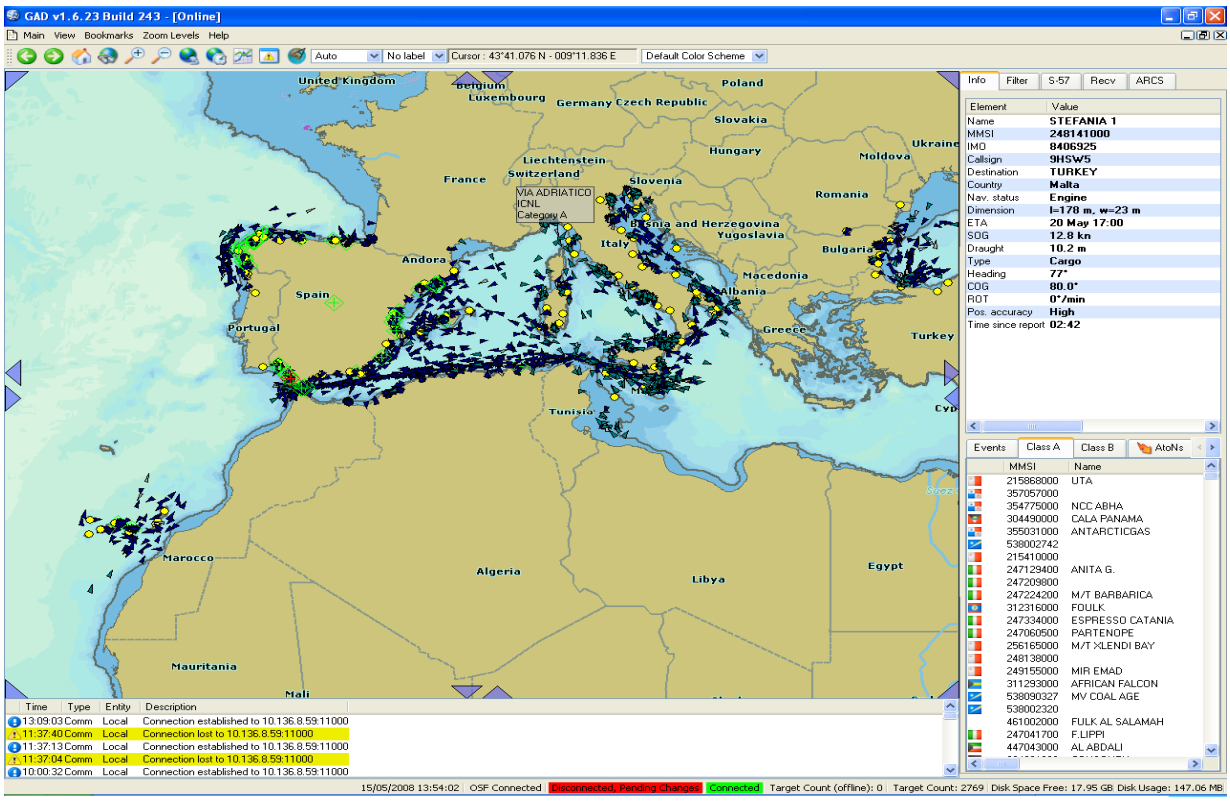


Figura 5.4: Visualizzatore di dati AIS collegato alla subscriber line del proxy.

Per garantire i necessari standard di sicurezza, si è scelto di instaurare tali connessioni tramite rete pubblica in **SSL/TLS** ed il modulo software che assolve la funzione di *proxy* è stato installato in una DMZ (*Demilitarized Zone*).

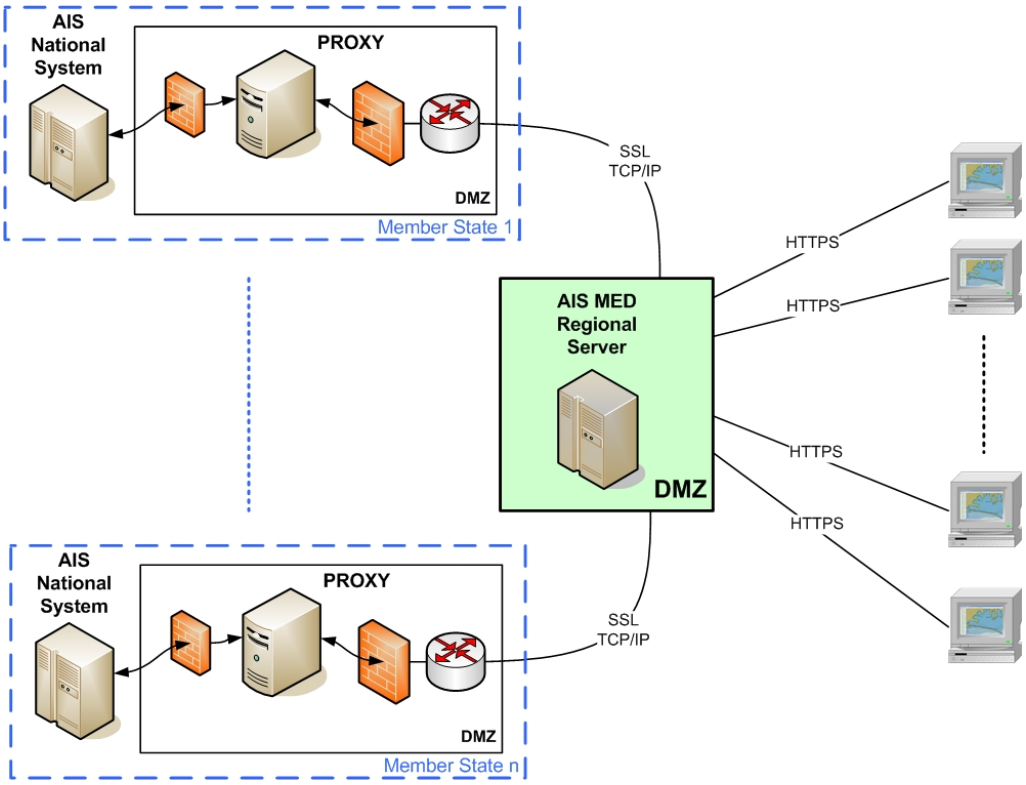


Figura 5.5: sicurezza nello scambio dati AIS.

5.2.1 *Demilitarized Zone* (DMZ)

Una DMZ è un segmento isolato di LAN (una "sottorete") raggiungibile sia da reti interne che esterne che permette connessioni verso l'esterno. Tale configurazione viene normalmente utilizzata per permettere ai server posizionati sulla DMZ di fornire servizi all'esterno senza compromettere la sicurezza della rete interna nel caso una di tali macchine sia sottoposta ad un attacco informatico. Per chi si connette dall'esterno dell'organizzazione la DMZ è infatti una sorta di "strada senza uscita" o "vicolo cieco".

Solitamente sulla DMZ sono collegati i server pubblici (ovvero quei server che necessitano di essere raggiungibili dall'esterno della rete aziendale - ed anche dalla internet - come, ad esempio, *server mail*, *web server* e *server DNS*) che rimangono in tal modo separati dalla LAN interna, evitando di comprometterne l'integrità.

Una DMZ può essere creata attraverso la definizione di *policies* distinte su uno o più firewall.

Oltre al summenzionato firewall, di tipo *packet filter* (ossia che opera a livello di trasporto - layer quattro della pila ISO/OSI-), è possibile implementare un differente approccio, impiegando un servizio di "*reverse proxy*", che agisce a livello applicativo (layer sette della pila ISO/OSI).

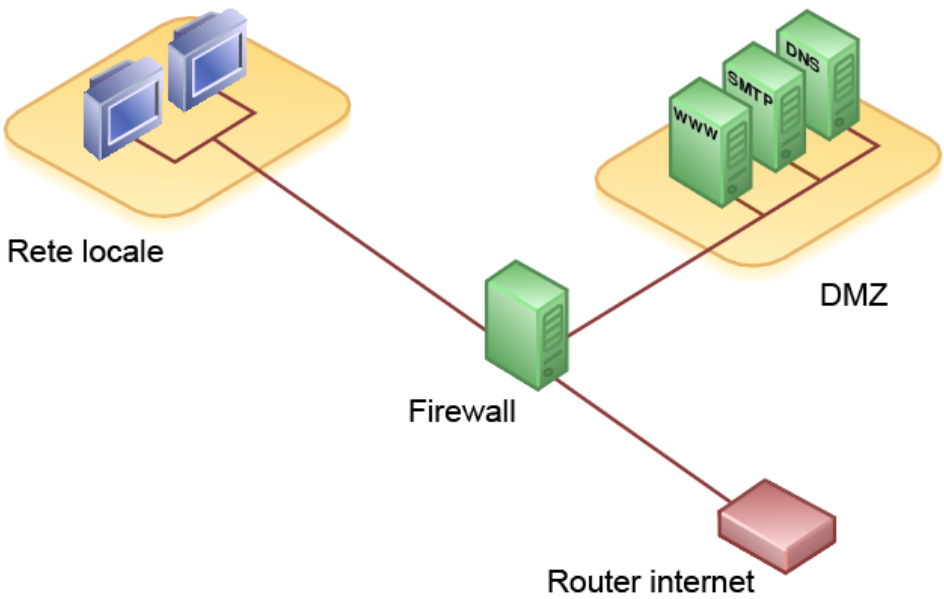


Figura 5.6: schema di una DMZ.

5.3 Server Regionale.

Il server regionale è il perno centrale dell'architettura distribuita per la raccolta e diffusione dei dati AIS degli Stati partecipanti al progetto.

La sua struttura è stata pensata e realizzata in 4 moduli ben distinti:

- modulo per la raccolta dati;
- modulo di memorizzazione dati;
- modulo per la distribuzione dei dati;
- modulo per l'accesso web.

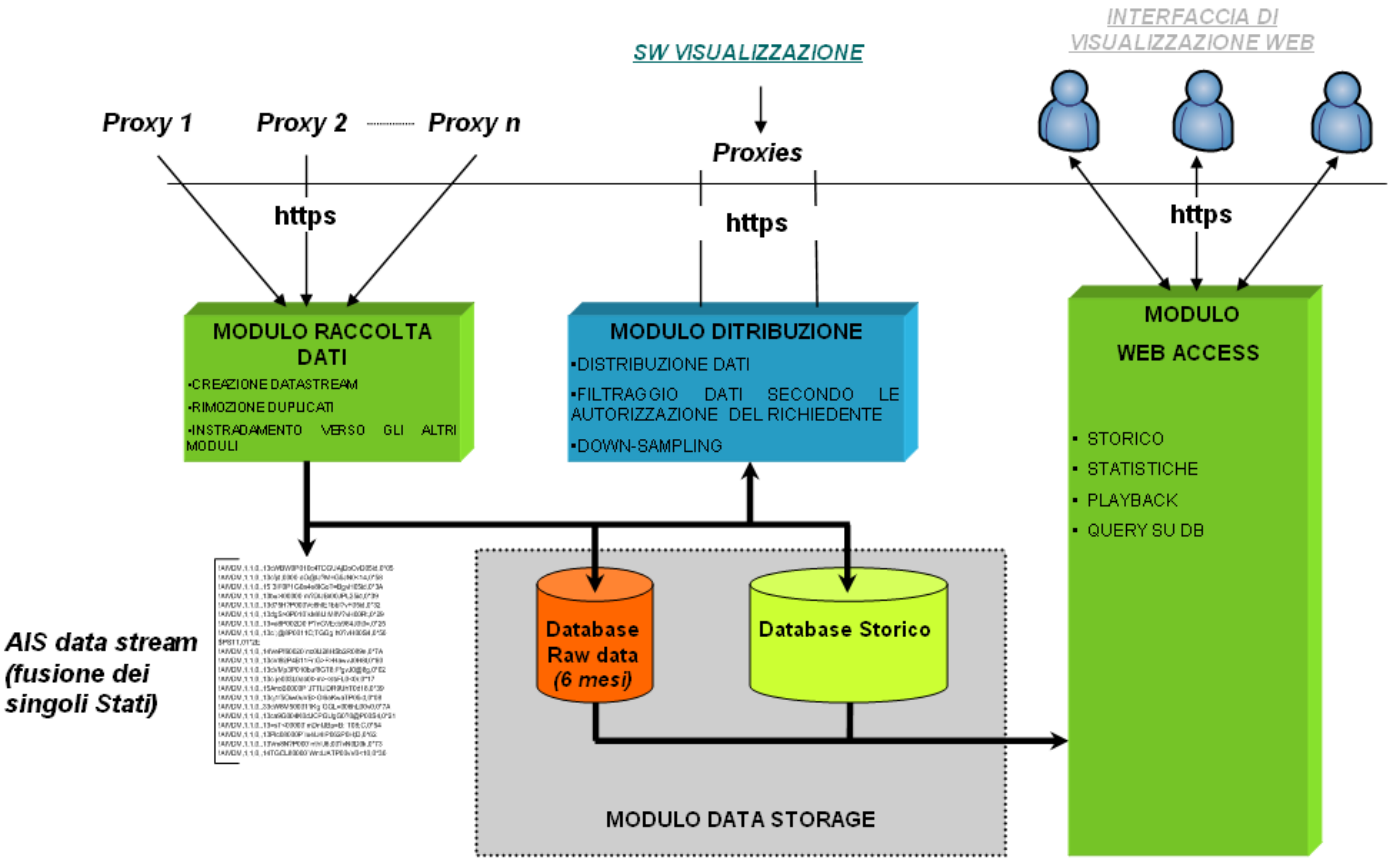


Figura 5.7: Moduli del Server Regionale.

Ogni modulo ha ruolo ben definito che verrà di seguito illustrato, precisando che il tutto si basa su uno strato software ***Linux Red Hat***, con ***Web Server Apache*** e *database management system* ***Oracle 10g***.

Il modulo dedicato alla **raccolta dei dati** lavora tramite una *socket TCP* che riceve i dati "grezzi" provenienti dai *proxies*, che a loro volta ricevono i dati dai rispettivi server nazionali non facendo altro che inoltrarli verso il server regionale MAREΣ.

I dati ricevuti vengono fusi in un unico *datastream* ed instradati verso gli altri moduli, non prima, però, di una importante operazione di rimozione dei duplicati tramite un banale confronto tra le stringhe di dati ricevute. Infatti non è raro che i dati trasmessi dalle unità navali presenti nelle zone di confine tra due Stati, vengano ricevuti da due *base stations* di "nazionalità" differente, quindi memorizzate in due server centrali diversi che si trovano perciò ad inoltrare la stessa informazione verso il server regionale che, se non operasse alcun un controllo, si troverebbe affollato di dati ridondanti.

I dati raccolti dal modulo appena descritto vengono **memorizzati** prima in un database, normalizzato alla terza forma, e poi elaborati da un *datawarehouse* che consente di produrre facilmente relazioni ed analisi statistiche. Tali operazioni sono a carico del modulo di *storage* che può contare sue due database: il primo composto da innumerevoli tabelle dedicate alla raccolta dei cosiddetti "dati grezzi", dimensionato con una capienza tale da assicurare la possibilità di risalire a informazioni pervenute sino ai sei mesi precedenti; il funzionamento del secondo database si basa su un *trigger* giornaliero che, scattando alla mezzanotte di ogni giorno, avvia una fase di memorizzazione dei dati, non grezzi, ma decodificati e leggibili, al fine di poterli lavorare e analizzare per finalità statistiche.

Il terzo modulo è colui che si occupa della **distribuzione** del *datastream* fuso, anch'esso lavora su una *socket TCP* che mette a disposizione le informazioni ai *clients* che vi si collegano, i dati vengono poi trasferiti in maniera sicura tramite HTTPS. Nello specifico chi fa da client sono gli stessi *proxy* che, quindi, da un lato trasmettono i dati del server nazionale a cui sono collegati e dall'altro, tramite uno strato applicativo software (*subscriber line*), mettono a disposizione i dati di tutti gli altri Stati fusi in un unico datastream.

Questo modulo consente anche di filtrare le informazioni distribuite ed, eventualmente, di effettuare una decimazione, in maniera tale da limitare la quantità di dati da trasferire.

L'ultimo modulo da descrivere è l'**interfaccia web** che consente l'accesso, tramite un qualunque accesso internet, ai dati elaborati da MAREΣ. Ovviamente tale accesso è consentito solo tramite inserimento di opportune credenziali di riconoscimento. Il cuore di tale applicazione è il GIS (*Geographical Information System*), la cui principale funzione è quella di elaborare le stringhe AIS memorizzate nel database "grezzo" tramutandole in simboli visualizzati poi su opportuna cartografia georeferenziata.

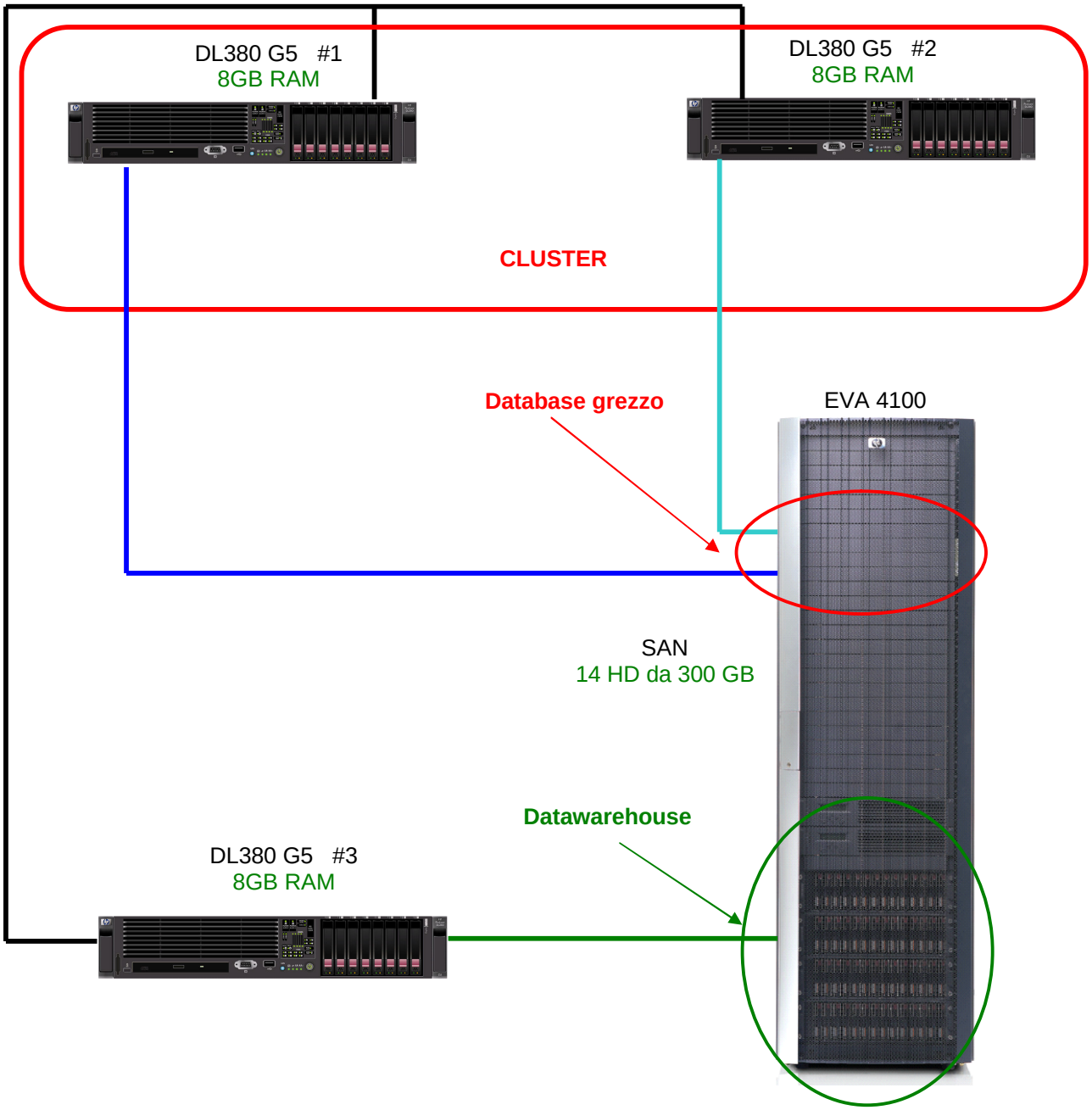


Figura 5.8: architettura fisica del Server Regionale.

Nella figura è illustrata l'architettura fisica in cui si ha una sezione Cluster, costituita da due server HP DL380 G5 e da 4 hard disk da 300 GB, che svolge le seguenti funzioni:

- Raccolta dati dai Proxy nazionali
- Distribuzione dei dati ai Proxy nazionali
- Archiviazione dei dati AIS su database in formato grezzo (*raw data*)

La restante sezione, costituita da un server HP DL380 G5 e da una unità EVA4100 con la possibilità di memorizzare sino a 3,0 TB di dati grezzi, si occupa di offire i seguenti servizi applicativi:

- Presentazione dati su portale WEB
- Analisi statistiche dei dati
- *DataWarehouse*.

5.4 Larghezza di banda.

Per l'AIS di classe A (*upper case*) le informazioni vengono codificate secondo lo standard IEC 61162-2, in stringhe dati di diversa lunghezza: i dati dinamici impegnano 50 bytes, mentre i dati statici e di viaggio utilizzano 115 bytes. Inoltre per assumere il *timestamp* di ogni stringa ricevuta occorre, per ognuna di essa, aggiungere un campo composto da 4 bytes che rappresenti in secondo il numero di secondi trascorsi rispetto ad un *time server* di riferimento.

Quindi, per i dati AIS trasmessi da un transpondere di classe A, la quantità di byte necessaria è

$$D = 50+4+115+4 = 173 \text{ bytes} = 1384 \text{ bits}$$

Pertanto possiamo effettuare un dimensionamento sui requisisti minimi di banda necessaria per la connessione Internet che consente di scambiare dati tra il Server Regionale ed i proxies nazionali:

$$\text{RegBW} = \frac{20000 \cdot 1384}{360 \cdot (1 - X)} \approx 200 \text{ kbits/s}$$

detto X il margine di impegno minimo da assicurare in larghezza di banda, assunto pari a 0,6 (più del 50%), considerando che il tempo necessario a trasmettere in maniera completa tutte le informazioni relative ad un target AIS è pari a 360 sec.

5.5 Collegamento tra "*Proxies* Nazionali" e Server Regionale.

Lo scambio delle informazioni tra i *Proxies* Nazionali ed il Server Regionale avviene tramite una rete aperta, quale internet. Per assicurare i necessari requisiti di sicurezza, in fase di progetto ho scelto di realizzare il collegamento *client-server* mediante una connessione *socket* TCP/IP di tipo protetto, utilizzando un protocollo SSL per la cifratura dei dati scambiati. L'autenticazione del *Proxy* sul Server Regionale viene regolamentato da una ***userid*** ed una ***password***.

La tecnica scelta per la cifratura SSL è stata quella *single way SSL*, con certificati SSL a 128 bit residenti solo sul Server Regionale. Il numero di certificati SSL installati sul Server è pari a due unità, uno per l'accesso sicuro via Web e l'altro per l'accesso via *Proxy*.

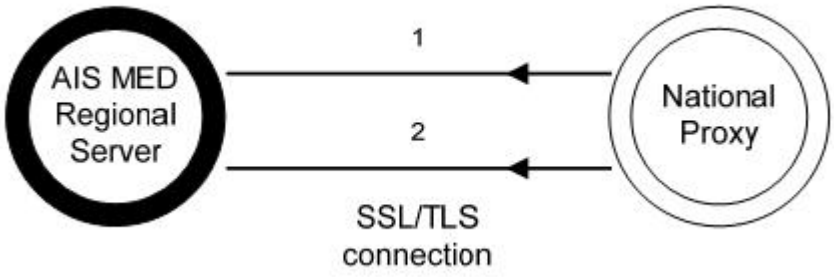


Figura 5.9: schema collegamento SSL/TLS.

La soluzione SSL/TLS (*Secure Sockets Layer/Transport Layer Security*) adottata prevede l'assegnazione di un certificate digitale X.509 v3 rilasciato da un Ente Certificatore riconosciuto a livello internazionale.

La creazione di un canale protetto segue quindi le seguenti fasi:

1. quando il Client (proxy) avvia la richiesta di connessione sicura, il Server invia al client il proprio certificato digitale e i protocolli di cifratura supportati;
2. il Client verifica l'autenticità del certificato ricevuto, procede alla scelta del protocollo di cifratura tra quelli contenuti nel certificato e invia tale scelta al Server;
3. stabilito quindi il protocollo, inizia una fase di negoziazione dei parametri (chiavi) e viene creato un canale sicuro per scambiare i dati.

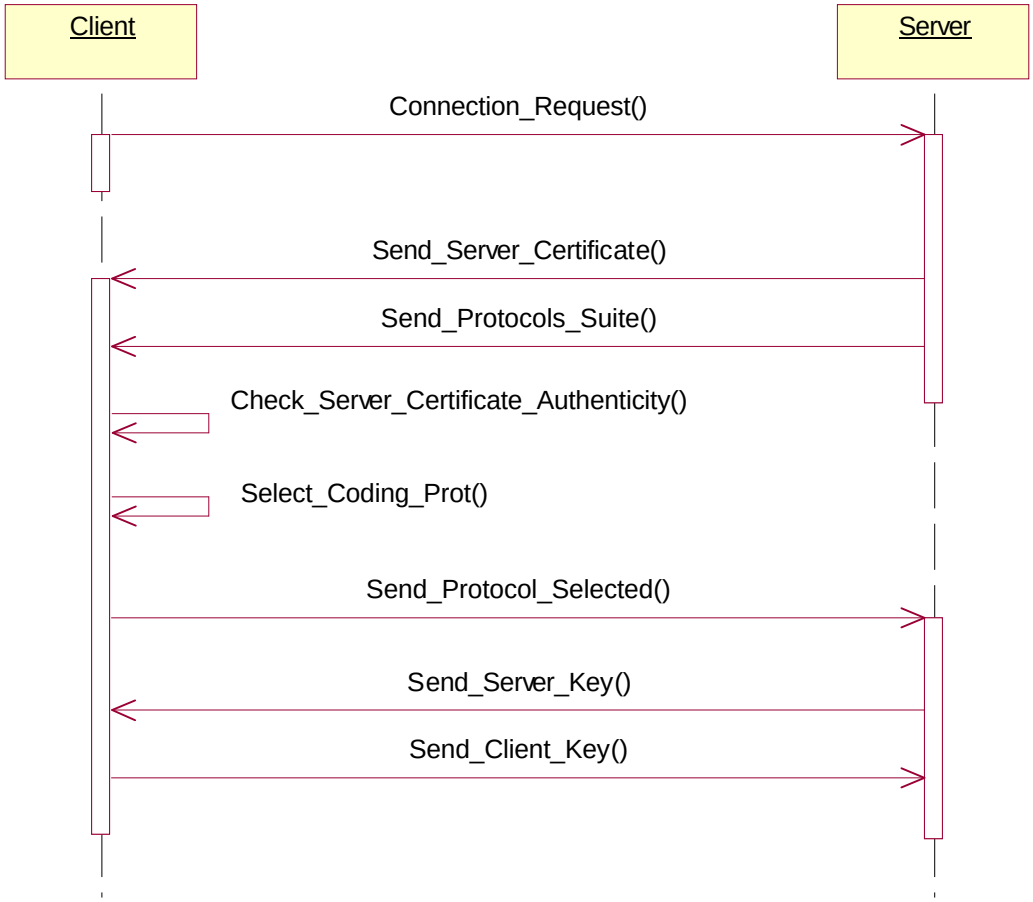


Figura 5.10: *workflow* per una connessione SSL/TLS a singola via.

I vantaggi di tale soluzione consiste nel limitato numero di certificati che l'Ente Certificatore deve gestire e nel ridurre anche il carico di lavoro che grava sul Client. Assicura, inoltre, un elevato livello di cifratura nella comunicazione condivisa tra Client e Server e non vi è latenza nella fase di *handshake* associata all'instaurazione della connessione.

Per ottenere migliori performance, oltre a limitare il numero di connessioni SSL, è importante la scelta della struttura della ***Cipher Suite SSL***, ovvero la combinazione del processo di autenticazione, codifica dati e dell'algoritmo per generare il blocco MAC (*Message Authentication Code*) per il controllo di integrità.

I parametri di configurazione scelti per le connessioni SSL/TLS con il Server Regionale sono:

- **RC4-128 bit**, per la fase di autenticazione;
- **RSA**, per lo scambio delle chiavi;
- **MD5**, come algoritmo MAC.

5.6 Stima dei ritardi di trasmissione.

Nella comunicazioni tra il Server ed un *proxy* i dati trasmessi subiscono, ovviamente, un ritardo rispetto all'istante di invio. Tale ritardo risulta essere la somma di diversi contributi:

- **ritardo di rete**, dovuto al tempo di propagazione da un nodo all'altro della rete
- **ritardo di elaborazione,** da attribuire ai tempi di elaborazione necessari ai nodi della rete per gestire i dati da inoltrare;
- **ritardi dovuti al sistema**, sia il proxy che il server regionale hanno un certo tempo di elaborazione nel completare i rispettivi processi software.

Per valutare il ritardo accumulato prendiamo in considerazione il caso tipico in cui uno Stato partecipante (MS1) riceve dati AIS da un altro Stato (MS2) per il tramite del Server Regionale. Il tempo trascorso da quando il messaggio AIS trasmesso da MS1 fino a quando viene ricevuto da MS2 può essere calcolato come:

dove

$$T = T_{LAN\text{-}MS1} + T_{PROXY\ MS1} + T_{Internet1} + T_{MED\ SERV} + T_{Internet2} + T_{PROXY\ MS2} + T_{LAN\text{-}MS2}$$

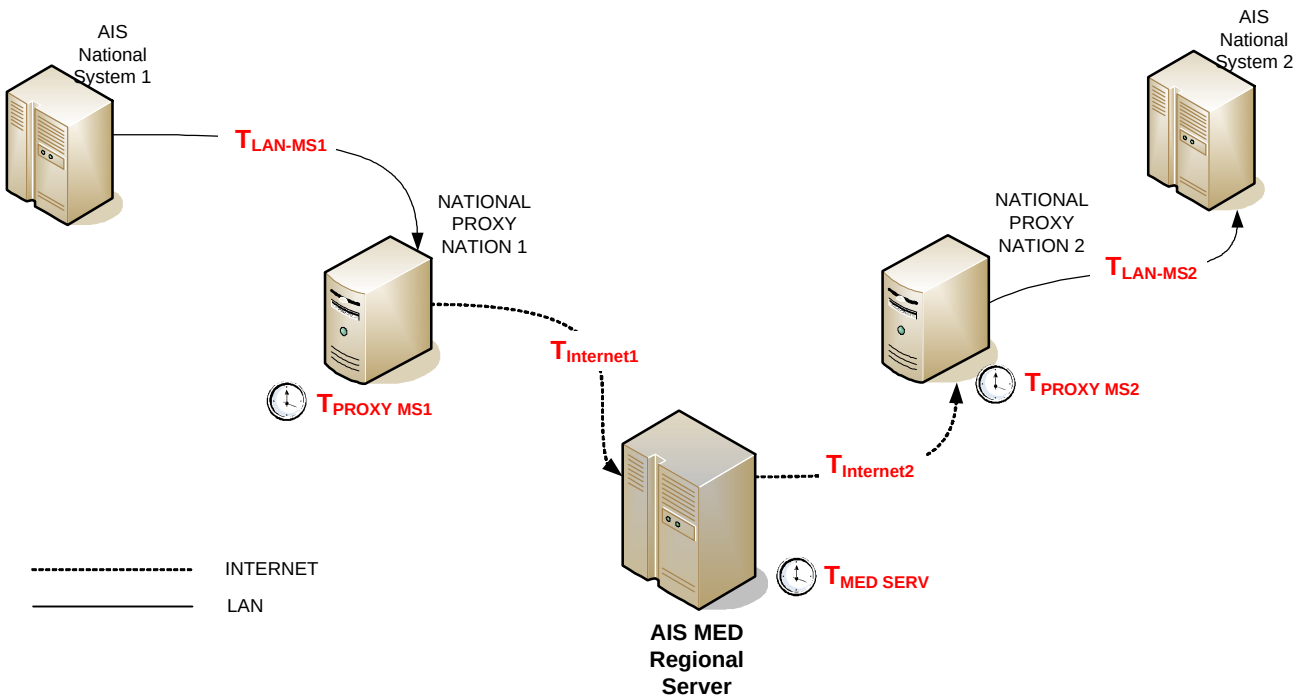


Figura 5.11: stima dei ritardi nella comunicazione.

Nella formula succitata sono indicati una serie di contributi, dettagliati nella seguente tabella, in cui sono evidenziate le stime rilevate statisticamente e dimensionate in millisecondi.

Ritardo	Descrizione	Stima
TLAN-MS	Ritardo di rete nella fase di trasmissione del messaggio AIS tra il Proxy ed il Server Nazionale	< 10 ms
TPROXY-MS	Ritardo dovuto ai tempi elaborativi del Proxy	< 50 ms
TInternet	Ritardo dovuto alla rete internet nella fase di inoltro del messaggio AIS tra il Proxy ed il server Regionale	>50 ms e < 1000 ms
TMED SERV	Ritardo dovuto ai tempi elaborativi del Server Regionale per raccogliere,filtrare e distribuire da e per i proxies	< 100 ms

Per mantenere i requisiti di "*real time*", necessary per le finalità di utilizzo del sistema, I tempi di elaborazione di Proxy e Server sono stati mantenuti bassi (meno di una decina di millisecondi), evitando la necessità di utilizzo di un buffer software. Ad ogni modo, dato che parte dell'architettura si basa sullo scambio dati tramite internet, non si possono escludere ritardi che possano inficiare il funzionamento del sistema.

5.7 Conclusioni.

MARE• si basa su tecnologie semplici e consolidate, in questo lavoro di tesi si è affrontato come combinare tali elementi al fine di predisporre un'architettura solida e sicura.

Il sistema finale è stato pensato, progettato e realizzato con una struttura basata su 4 elementi:

1. un Server Regionale che raccoglie e distribuisce i dati dei server nazionali;
2. un Proxy software per ogni Stato partecipante al progetto, che inoltra dati dai server nazionali verso il Server Regionale e viceversa;
3. la rete internet che assicura il collegamento tra i due nodi di cui sopra;
4. le ASM degli Stati partecipanti, che raccolgono i dati dalla propria rete e li mettono a disposizione del Proxy.

Una volta individuata e scelta una soluzione basata su *proxies*, notevole attenzione è stata posta soprattutto al terzo punto, valutando come trasferire dati di una certa importanza nella maniera più sicura possibile.

I protocolli di sicurezza e di codifica sono protocolli di trasmissione utilizzati per trasmettere in maniera sicura dati, la codifica offre tre vantaggi fondamentali

- **Riservatezza dei dati**, ovvero la capacità di nascondere i dati trasmessi;
- **Autenticità e integrità dei dati**, l'algoritmo matematico di codifica permette ai protocolli di sicurezza di garantire che i dati non siano stati modificati o danneggiati durante la trasmissione;
- **Nessun rifiuto**, un'altra caratteristica dell'algoritmo di codifica è la possibilità di provare che un evento si è verificato.

Un ampio studio delle tecniche di sicurezza informatica, ha portato a prendere in considerazioni due possibili alternative: IPSec. e SSL/TLS. La scelta finale è poi ricaduta sulla seconda, in seguito alle valutazioni sulle differenze emerse dal confronto di queste due soluzioni.

Abbiamo ampiamente illustrato IPSec, un protocollo che garantisce la trasmissione sicura dei dati codificati presso il *Network Layer*, su una rete pubblica come internet.

Una volta che il tunnel IPSec è stato attivato, tutti i protocolli del network layer tra due parti in comunicazione sono codificati. L'IPSec, però, richiede un software per client dedicato, che in molti casi sostituisce o complementa i sistemi client dello stack TCP/IP .

L'SSL utilizza la codifica e l'autenticazione in maniera analoga all'IPSec. e può codificare tutto il traffico tra client e server in maniera similare, ma senza l'utilizzo di un client.

L'SSL è un protocollo di per sé molto veloce; tuttavia, come ogni altro protocollo di codifica, vi sono alcuni calcoli estremamente onerosi per la CPU che devono essere effettuati prima che venga stabilita una sessione sicura. Un esempio è l'algoritmo RSA, che viene utilizzato nell'ambito dell'SSL per negoziare le chiavi tra client e server. Nel processo di negoziazione, il server deve decodificare e verificare una firma digitale, entrambe sono operazioni al quanto impegnative.

Vale la pena di notare che la sequenza delle operazioni di cifratura e autenticazione è diversa nei due casi: in TLS prima si calcola il MAC e poi si cifra il tutto, MAC compreso, mentre in IPsec prima viene effettuata la cifratura e poi si aggiungono i dati di autenticazione, che vengono quindi calcolati sul messaggio cifrato e non su quello in chiaro.

Le differenze tra IPsec e TLS sono per la maggior parte una diretta conseguenza della loro diversa posizione all'interno dello stack TCP/IP. TLS offre un canale sicuro tra due applicazioni, poiché opera al livello dei socket, mentre i terminali della comunicazione IPsec sono due macchine.

Per quanto riguarda i protocolli di livello superiore, IPsec protegge tutto ciò che sta sopra IP, quindi ad esempio TCP, UDP, ICMP. TLS invece si deve appoggiare su un protocollo di trasporto affidabile, e quindi può essere utilizzato solo per proteggere traffico TCP.

In definitiva TLS è adatto a proteggere la comunicazione tra due applicazioni *host-to-host*, mentre IPsec può facilmente rendere sicuro tutto il traffico tra determinati host, o tra intere sottoreti.

RIFERIMENTI

[1] Michele M. Comenale Pinto, G. Spera, "*Profili Giuridici dell'Automatic Identification System (AIS)*"

[2] Leica Geosystems (2001), The Complete Guide to Automatic Identification Systems, Leica Geosystems AG, Heinrich Wild Strasse, CH-9435 Heerbrugg, St.Gallen, Switzerland (http://www.concordelectronics.com/pdf/AIS_Booklet.pdf).

[3] Chang, S.J. (2004), "Development and analysis of AIS applications as an efficient tool for vessel traffic service", Ocean '04 - MTS/IEEE Techno-Ocean '04: Bridges across the Oceans - Conference Proceedings, Marine Technology Society Inc., Columbia, MD 21044, United States, pp.2249-2253.

[4] Gebruers, C. (2005), The "Less Obvious" Benefits of AIS, Cork Constraint Computation Centre, Department of Computer Science, University College Cork, Cork, Ireland.

[5] Stitt, I.P.A. (2004), "AIS and collision avoidance - a sense of déjà vu", Journal of Navigation 57(2),167-80, Cambridge University Press for R. Inst. Navigation at R. Geogr. Soc.

[6] Pratt, C.R. & Taylor, G. (2004), "AIS - a pilot's perspective", Journal of Navigation 57(2),181-188, Cambridge University Press for R. Inst. Navigation at R. Geogr. Soc. (http://www.pilotmag.co.uk/AIS%20Survey.pdf).

[7] Harre, I. (2000), "AIS adding new quality to VTS systems", Journal of Navigation 53(3), 527-539, Cambridge University Press for R. Inst. Navigation at R. Geogr. Soc.

[8] Bulitko, V. & Wilkins, D. (2002), Real-time Decision Making For Shipboard Damage Control, Vadim Bulitko Department of Computing Science, University of Alberta Edmonton, Alberta T6G 2H1, Canada (www.kddresearch.org/Workshops/RTDSDS-2002/papers/RTDSDS2002-BW-05.pdf)

[9] Calfee, S. & Rowe, N.C. (2002), An Expert System and Tutor for Maritime Navigation Rules, U.S. Naval Postgraduate School, Monterey CA 93943 USA (http://www.cs.nps.navy.mil/people/faculty/rowe/ccrt02b.htm).

[10] Pillich, B. & Buttgenbach, G. (2001), "ECDIS - the intelligent heart of the Hazard and collision avoidance system", IEEE Conference on Intelligent Transportation Systems Proceedings, ITSC, Oakland, CA, pp. 1116-1119 (http://ieeexplore.ieee.org/iel5/7537/20514/00948818.pdf?arnumber=948818).

[11] Ghallib, M., Nau, D. & Traverso, P. (2004), "Planning with Time and Resources", Automated Planning: Theory and Practise, The Morgan Kaufmann Series in Artificial Intelligence, 500 Sansome Street, Suite 400, San Francisco, CA 94111, pp.281-373.

[12] Risoluzione IMO A.851(20) "Linee Guida Per Sistemi Di Ship Reporting"

[13] IMO Resolution MSC.74(69), Annex 3, Recommendation on Performance Standards for an Universal Shipborne Automatic Identification Systems (AIS)-

[14] ITU-R Recommendation M.1371-3, Technical Characteristics for a Universal Shipborne Automatic Identification System Using Time Division Multiple Access in the Maritime Mobile Band.

[15] IEC 61993-2 Ed.1, Maritime navigation and radiocommunication requirements - Automatic identification systems (AIS) - Part 2: Class A shipborne equipment of the universal automatic identification system (AIS) - Operational and performance requirements, methods of test and required test results.

[16] Stallings W., *"Crittografia e Sicurezza delle reti"*, McGraw-Hill (2007)