VoIP: Media Security

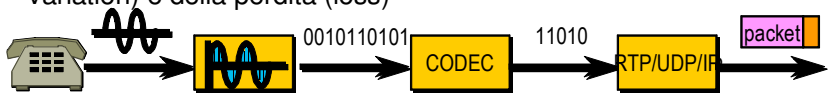
Luca Veltri
(mail.to: luca.veltri@unipr.it)

Corso di Sicurezza nelle reti, Reiti di telecomunicazioni C, a.a. 2009/2010
http://www.tlc.unipr.it/veltri

RTP

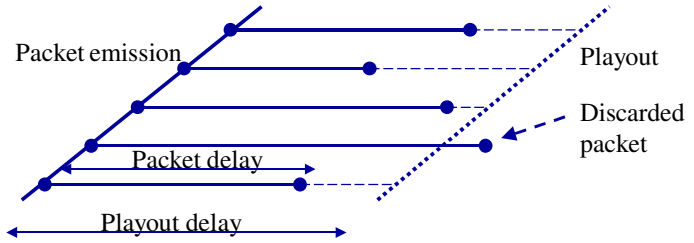
Digitalizzazione e pacchettizzazione della voce

- Partendo da un segnale audio (e.g. voce con banda 300-3400Hz), per inviarlo tramite una rete dati (e.g. IP) si deve:
  - **campionare il segnale analogico (e.g. a 8kHz)**
  - **quantizzare e codificare in binario i campioni, otienendo così un flusso dati**
  - **opzionalmente codificare il flusso dati a blocchi (frame) in modo da ottenere dati più compatti (si possono usare vari codec di compressione)**
  - **spedire i frame all'interno di pacchetti (e.g. IP), eventualmente insieme ad informazione di controllo (e.g. RTP) per facilitarne in ricezione la decodifica e riproduzione**
- La maggior parte di questi passi e l'attraversamento della rete contribuiscono ad introdurre un ritardo variabile (delay e delay variation) e della perdita (loss)



Requisiti per il trasferimento real-time

- Pacchetti generati con ritmo regolare (e.g. flusso audio/video) devono essere riprodotti a destinazione con lo stesso ritmo
  - **un buffer di playout a destinazione può essere usato per ammortizzare la variabilità dei ritardi**
  - **i pacchetti devono però includere numeri di sequenza e istanti di tempo relativi**



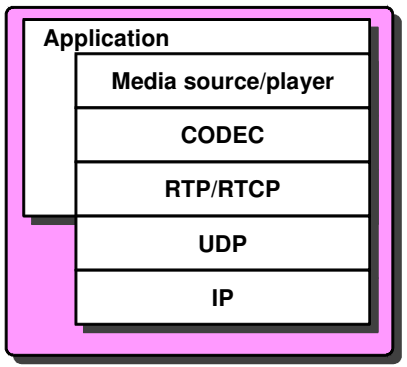
Requisiti per un protocollo real-time

- TCP non è adatto per applicazioni real-time poiché:
  - **il controllo di congestione del TCP non si adatta ad una trasmissione in tempo reale**
  - **il controllo di errore del TCP introduce ritardi troppo elevati; i pacchetti persi non possono essere ritrasmessi in caso di dati real-time**
  - **non mantiene relazione temporale tra istanti diversi di invio e ricezione**
  - **non supporta collegamenti multicast**
- UDP al contrario, si adatta meglio, poiché:
  - **supporta mulitcast a livello IP**
  - **non utilizza meccanismi di controllo di flusso, congestione e recupero di errore**
- Però UDP non ha funzionalità per riordino dei pacchetti in sequenza e recupero della corretta cadenza temporale

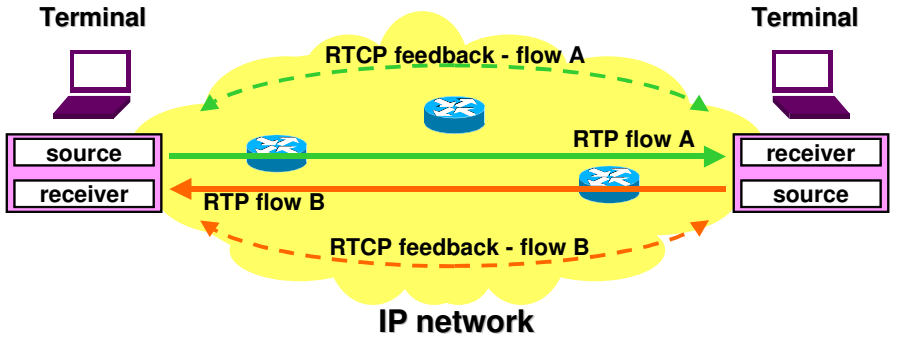
Real-time Transport Protocol (RTP)

- Standard IETF RFC 3550 and 3551 (2003)
  - **Protocollo di trasporto end-to-end per applicazioni real-time**
    - source identification
    - payload type identification
    - sequence numbering
    - timestamping
    - delivery monitoring
  - **E' un protocollo dati e non instaura sessioni; per questa funzione si possono usare altri protocolli come SIP**
  - **Lo standard definisce anche il protocollo RTCP (RTP Control Protocol) per scambiare informazione di controllo tra sorgente e destinazione sulla qualità del flusso RTP**
- RTP di solito viene usato sopra UDP/IP
  - **supporta sia comunicazione punto-punto che punto-multipunto (con IP multicast)**
- Può trasportare qualsiasi media e codec (protocollo trasparente)

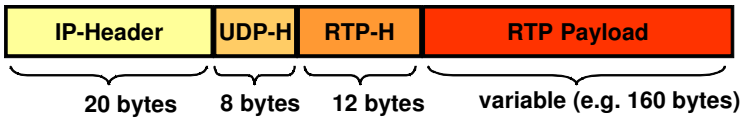
Protocol Architecture



Point-to-point RTP

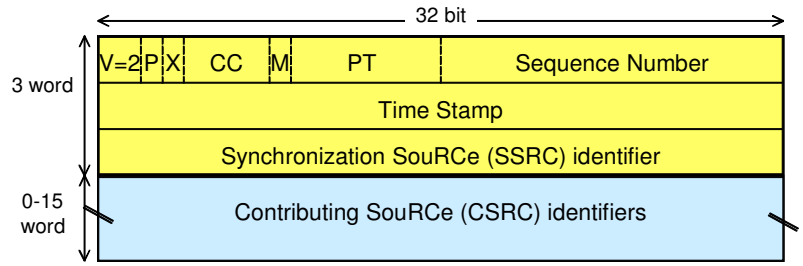


IP/UDP/RTP packet



- IP header: 20 octets (IP addresses, …)
- UDP header: 8 octets (port identifier, …)
- RTP header: 12 octets
  - **source id**
  - **payload type**
  - **sequence number**
  - **time stamp**
- Payload: encoded voice or multi-media stream (i.e. 20 bytes each 20 ms)

RTP Header Format



V = Version [2bit]
P = Padding [1bit]
X = Extension [1bit]
CC = CSRC Count [4bit]
M = Marker [1bit]

RTP Header Format (cont.)

- Version (2bit)
  - **current version = 2**
- Padding (1 bit)
  - **If set, the packet contains one or more additional padding octets at the end; the last octet contains the number of padding octets**
- Extension (1bit)
  - **If set, the fixed header is followed by exactly one header extension**
- Marker Bit (1 bit)
  - **marks significant events (depends on the payload type), e.g. frame boundaries ("in-band signaling")**
- Payload Type (7 bit)
  - **identifies the format of the RTP payload and determines its interpretation by the application (e.g. PCM A-law, G.729 codecs...)**
  - **additional payload type codes may be defined dynamically through non-RTP means**

RTP Header Format (cont.)

- Sequence Number (16 bit)
  - **increments by one for each RTP data packet sent,**
  - **may be used by the receiver to detect packet loss and to restore packet sequence**
  - **the initial value of the sequence number is random (unpredictable)**
- Time Stamp (32 bit)
  - **the timestamp reflects the sampling instant of the first octet in the RTP data packet**
  - **the clock frequency is dependent on the format of the data (for PCM audio codec f= 8kHz, T= 125µs)**
  - **if RTP packets are generated periodically, the nominal sampling instant as determined from the sampling clock is to be used**
  - **the initial value of the timestamp is random**

RTP Header Format (cont.)

- Sinchronization SouRCe - SSRC (32 bit)
  - **identifies the synchronization source**
  - **this identifier is chosen randomly, with the intent that no two synchronization sources within the same RTP session will have the same SSRC identifier**
- Contributing SouRCe - CSRC (32 bit)
  - **The CSRC list identifies the contributing sources for the payload contained in this packet**
  - **CSRC identifiers are inserted by mixers**
  - **the CSRC can be both a terminal identifier or a mixer identifier**
- CSRC Count - CC (4 bit)
  - **counts the number of CSRC (0-15)**

RTP Payload Types

Payload Type	Encoding Name
0	PCMU
1	1016
2	G721
3	GSM
4	unassigned audio
5	DV14 (8KHz)
6	DV14 (16KHz)
7	LPC
8	PCMA
9	G722
10	L16 Stereo
11	L16 Mono
12	TPSO
13	VSC
14	MPA
15	G728
16-23	unassigned

Payload Type	Encoding Name
23	RGB8
24	HDCC
25	CelB
26	JPEG
27	CUSM
28	NV
29	PicW
30	CPV
31	H261
32	MPV
33	MP2T
34-71	unassigned
72-76	reserved
77-95	unassigned
96-127	dynamic

RTP Control Protocol (RTCP)

- Protocollo per scambiare informazioni di controllo tra i partecipanti di una sessione RTP
  - **è definito per essere usato accanto a RTP**
  - **è di fatto opzionale e non tutte le applicazioni lo usano (specialmente se implementano altro protocollo di controllo si sessione, e.g. SIP)**
- RTCP invia i pacchetti alla porta UDP successiva a quella usata da RTP
  - **RTP usa porta pari**
  - **RTCP usa porta dispari successiva**
- I pacchetti RTCP dovrebbero essere generati ad un ritmo tale pari al massimo al 5% del traffico RTP

Tipi di pacchetti RTCP

- Sender Report (SR)
  - **inviati dal sender a tutti i receiver**
  - **contengono statistiche e altre informazioni (bytes sent, timestamps, etc.)**
- Receiver Report (RR)
  - **inviati dal ogni ricevitore al sender**
  - **contengono statische di ricezione (estimated packet loss, inter-arrival jitter, round trip delay, etc.)**
- Source Description (SD)
  - **inviati dal sender a tutti i receiver**
  - **contengono informazioni addizionali sulla sorgente (e.g. CNAME, e-mail, phone no., etc.) che possono essere usate per associare più flussi alla stessa sorgente e per identificare la stessa**
- Bye
  - **inviati dal sender a tutti i receiver per comunicare la chiusura di una sessione (nel caso non vengano impiegati altri protocolli appositi come SIP)**

RTP Attacks: DoS

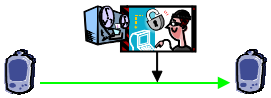
- RTP Denial of Service (DoS)
  - **utilizzando di SSRC falsi**
    - possibili risultato: disconnessione della sessione
  - **RTCP "BYE" non sincronizzato con il protocollo di segnalazione**
    - possibili risultato: lo scambio di flussi multimediali termina senza che venga notificato allo strato di controllo/segnalazione
  - **cambiando i valori dei report di ricezione, e.g. riportando piu' perdite di pacchetti, oppure ripordando un jitter più alto**
    - possibili risultato: il codec viene modificato dinamicamente in maniera adattiva, viene scelto un codec con minore qualita'

RTP Attacks: play-out and eavesdropping

- RTP play-out

  - **Spoofing del SSRC, numero di sequenza piu' alto, timestamp piu' alto**
    - Risultato: il contenuto immesso verra' presentato all'utente perche' ritenuto piu' attuale

- Call Eavesdropping



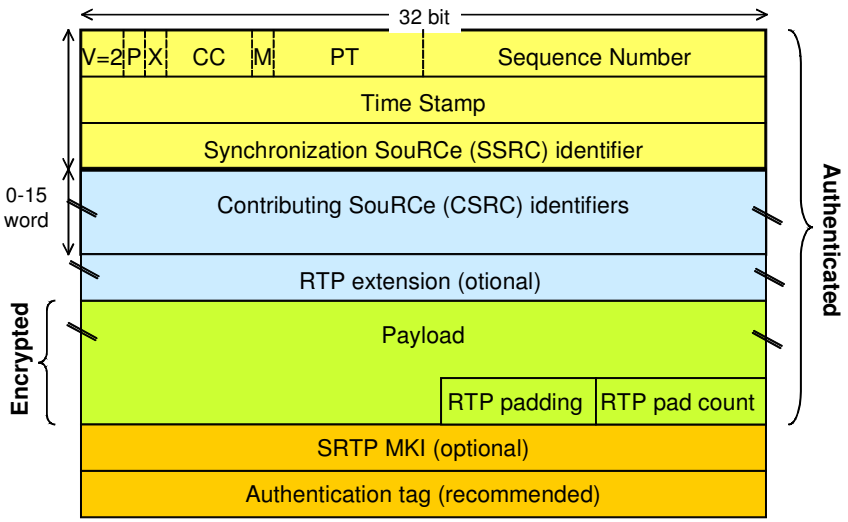
  - **Cattura di flussi RTP**
    - Dato che RTP identifica i codec utilizzati, e' facile ricostruire il flusso voce (anche in real time)
    - Risultato: ascoltare/registrare le conversazioni
    - Risultato: ascoltare i toni DTMF (passati su RTP) per rubare password e PIN
    - e.g. WireShark ha un "RTP Analysis module"

RTP Security

Secure Real-time Transport Protocol (SRTP)

- Extension (RFC 3711) of RTP, which can secure RTP and RTCP traffic
  - **confidentiality of the RTP and RTCP payloads**
    - default use of AES128
  - **integrity of the entire RTP and RTCP packets**
    - default use of HMAC-SHA1
  - **together with protection against replayed packets**
  - **a single "master key" can provide keying material for confidentiality and integrity protection (for SRTP and SRTCP)**
    - achieved with a key derivation function
- SRTP can be be implemented as "bump in the stack"
  - **between RTP and the transport layer**
    - SRTP intercepts RTP packets and then forwards an equivalent SRTP packet

SRTP packet format



SRTP packet format (cont.)

- SRTP introduce only two new fields (8-bit alignment is assumed):
  - **Master Key Identifier (MKI): configurable length**
    - OPTIONAL
    - defined, signaled, and used by key management
    - identifies the master key from which the session key(s) were derived that authenticate and/or encrypt the particular packet
    - the MKI does not identify the SRTP cryptographic context
    - may be used by key management for the purposes of re-keying
  - **Authentication tag: configurable length**
    - RECOMMENDED
    - used to carry message authentication data
    - provides authentication of the RTP header and of the encrypted payload
    - indirectly provides replay protection by authenticating the sequence number
    - MKI is not integrity protected as this does not provide any extra protection

SRTP Pre-defined Cryptographic Transforms

- SRTP allows the use of different encryption and message authentication algorithms
- SRTP defines also default transforms
  - **map the SRTP packet index and secret key into a pseudo-random keystream segment**
  - **each keystream segment encrypts a single RTP packet**
  - **packet encryption consists of generating the keystream segment corresponding to the packet, and then bitwise exclusive-oring it onto the payload of the RTP packet**
- Default cipher used to generate the keystream is AES
  - **two default modes have been defined:**
    - Segment Integer Counter Mode AES
    - AES in f8-mode

SRTP Security Association

- SRTP richiede che il sender e il receiver siano sincronizzati sugli algoritmi e sulla coppia di Master_key e Salt_key da utilizzare
- E' necessario un meccanismo sicuro di scambio di chiavi
  - **SRTP non definisce un meccanismo specifico**
  - **possono essere usati diversi approcci:**
    - chiavi e algoritmi pre-condivisi (statici)
    - protocolli di negoziazione
- Alcuni meccanismi che possono essere usati sono:
  - **MIKE: Multimedia Internet KEYing, RFC 3830**
    - authenticated key establishment protocol
    - k=..
  - **SDP Security Descriptions for Media Streams (SDES), RFC 4568**
    - key exchange protocol
    - a=crypto:1 AES_CM_128_HMAC_SHA1_80 inline:d0RmdmcmVCspeEc3QGZiNWpVLFJhQX1cfHAwJSoj