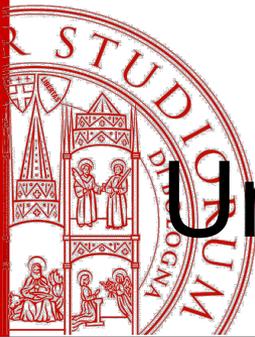


# MIME - Multipurpose Internet Mail Extensions

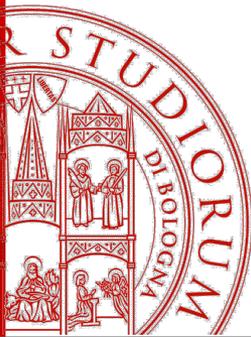
*Angelo Di Iorio*

*Università di Bologna*



# Un passo indietro: limiti di SMTP

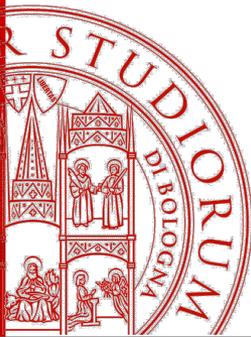
- SMTP è un protocollo *text-based*, per lo scambio di messaggi di posta elettronica e la verifica dei destinatari dei messaggi
- Ha alcuni limiti fondamentali che impediscono la trasmissione di file binari:
  - La lunghezza massima del messaggio è di 1 Mb
  - I caratteri accettati sono solo ASCII a 7 bit
  - Ogni messaggio deve contenere una sequenza CRLF ogni 1000 caratteri o meno
- MIME (Multipurpose Internet Mail Extensions) è stato definito per bypassare questi limiti e qui è rilevante perché permette di indicare codifica e i tipi delle risorse scambiate via HTTP



# Esempio

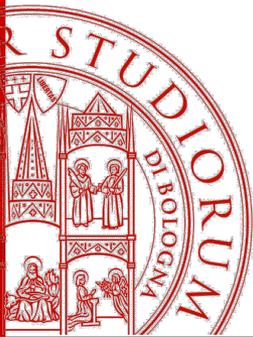
**From:** Giuseppe Verdi <g.verdi@CS.UniBO.IT>  
**Date:** Mer gen 15, 2023 17:25:47 Europe/Rome  
**To:** Paolo Rossi <p.rossi@CS.UniBO.IT>  
**Subject:** Prova  
**Received:** by le (mbox p.rossi) (with Cubic Circle's cucipop (v1.31 1998/05/13) Wed Jan 15 17:25:55 2023)  
**Received:** from [130.136.2.220] (genesis.cs.unibo.it [130.136.2.220])  
by CS.UniBO.IT (8.9.3/8.9.3/Debian 8.9.3-6) with ESMTMP id RAA29182  
for <p.rossi@cs.unibo.it>; Wed, 15 Jan 2023 17:25:45 +0100  
**User-Agent:** Microsoft-Entourage/10.0.0.1309  
**Message-Id:** <BA4B4A1B.D4BE%g.verdi@cs.unibo.it>  
**Mime-Version:** 1.0  
**Content-Type:** text/plain; charset="US-ASCII"  
**Content-Transfer-Encoding:** 7bit

Questo e' il contenuto o corpo del messaggio di posta elettronica.



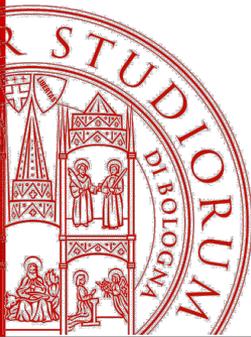
# I servizi MIME

- Dichiarazione di tipo
  - Tutti i messaggi MIME vengono identificati da un **Content Type**, che definisce il tipo di dati del messaggio e aiuta l'applicazione ricevente a gestire il messaggio e a invocare l'applicazione più adatta.
  - N.B.: l'attribuzione dell'applicazione non viene fatta sulla base dell'estensione del nome del file.
- Messaggi multi-tipo
  - Un messaggio MIME può contenere **parti di tipo diverso** (es. un messaggio di tipo testo e un attachment binario). In questo caso si creano dei sottomessaggi MIME per ciascuna parte (con il suo bravo content-type) e il messaggio MIME complessivo diventa “**multi-parte**”, qualificando e codificando in maniera diversa ciascuna sottoparte.



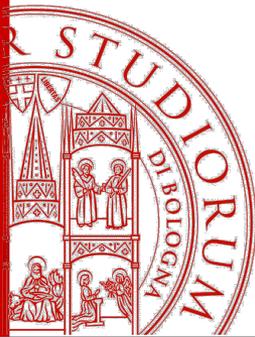
# Header specifici MIME

- MIME introduce alcuni nuovi header:
  - **Content-Type**: il tipo MIME del contenuto. Serve per permettere al ricevente di scegliere il meccanismo più adatto per presentare i dati. Specifica la natura del dato tramite la specificazione di tipo, sottotipo e ulteriori parametri utili.
    - **Content-Type**: `text/plain; charset=ISO-8859-1`
  - **Content-Transfer-Encoding**: il tipo di codifica utilizzata per trasmettere i dati. Serve per la trasmissione su canale SMTP di dati che non sono naturalmente corretti secondo le regole di SMTP: 7bit, sequenze CRLF ogni 1000 caratteri o meno. Sono valori accettabili “7bit” (default), “8bit”, “binary”, “quoted-printable”, “base64” o altre stringhe definite nel registro IANA
    - **Content-Transfer-Encoding**: `base64`



# MIME - Base 64 (1)

- Base 64 è un tipo di transfer encoding MIME suggerito per dati binari o multi-byte.
- Viene identificato un sottoinsieme di 64 caratteri di US-ASCII sicuri, che hanno la stessa codifica in tutte le versioni di ISO 646. Questi sono:
  - le lettere maiuscole (26, 'A' => 0),
  - Le lettere minuscole (26, 'a' => 26),
  - I numeri (10, '0' => 52)
  - I caratteri '+' e '/' (=> 62 e 63 rispettivamente).
- Ogni flusso di dati viene suddiviso in blocchi di 24 bit (3 byte). A loro volta questi 24 bit sono suddivisi in 4 blocchi di 6 bit ciascuno e codificati secondo una tabella prefissata in uno dei 64 caratteri già descritti.

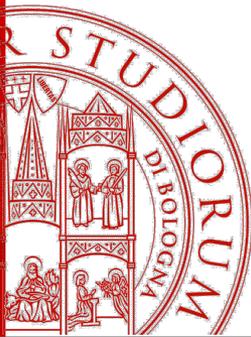


# Tabelle ASCII e Base 64

@	64	P	80	^	96	p	112
A	65	Q	81	a	97	q	113
B	66	R	82	b	98	r	114
C	67	S	83	c	99	s	115
D	68	T	84	d	100	t	116
E	69	U	85	e	101	u	117
F	70	V	86	f	102	v	118
G	71	W	87	g	103	w	119
H	72	X	88	h	104	x	120
I	73	Y	89	i	105	y	121
J	74	Z	90	j	106	z	122
K	75	[	91	k	107	{	123
L	76	\	92	l	108		124
M	77	]	93	m	109	}	125
N	78	^	94	n	110	~	126
O	79	_	95	o	111	del	127

Index	Binary	Char	Index	Binary	Char	Index	Binary	Char
0	000000	A	16	010000	Q	32	100000	g
1	000001	B	17	010001	R	33	100001	h
2	000010	C	18	010010	S	34	100010	i
3	000011	D	19	010011	T	35	100011	j
4	000100	E	20	010100	U	36	100100	k
5	000101	F	21	010101	V	37	100101	l
6	000110	G	22	010110	W	38	100110	m
7	000111	H	23	010111	X	39	100111	n
8	001000	I	24	011000	Y	40	101000	o
9	001001	J	25	011001	Z	41	101001	p
10	001010	K	26	011010	a	42	101010	q
11	001011	L	27	011011	b	43	101011	r
12	001100	M	28	011100	c	44	101100	s
13	001101	N	29	011101	d	45	101101	t
14	001110	O	30	011110	e	46	101110	u
15	001111	P	31	011111	f	47	101111	v

\* Tabelle incomplete, qui a supporto dell'esempio nella prossima slide.  
Maggiori dettagli nella lezione su codifica caratteri



# MIME – Base 64

<i>Input</i>	M	a	n	
<i>Codice ASCII</i>	77	97	110	
<i>mappa bit</i>	0 1 0 0 1 1 0 1	0 1 1 0 0 0 0 1	0 1 1 0 1 1 1 0	
<i>Indice 6-bit</i>	19	22	5	46
<i>Output Base64</i>	T	W	F	u
<i>Codice ASCII</i>	84	87	70	117

- La stringa risultante viene divisa in righe di 76 caratteri (tranne l'ultima, che è lunga quanto deve essere) con l'aggiunta di CR-LF.
- Nella decodifica i codici CR e LF sono da ignorare.
- La decodifica di Base64 è algoritmica, banale, non usa chiavi né calcoli di particolari complessità.
- **Base64 NON È una tecnica crittografica!!!**