Telematica I Parziale
00 Il modello ISO/OSI
Il modello ISO/OSI
Open System Interconnection (OSI)
Modello di riferimento nel campo della telematica.
“Paradigma di riferimento per la definizione di un sistema di elaborazione aperto ed interoperabile”
Sistema aperto: in grado di dialogare con qualsiasi altro sistema di elaborazione e di comunicazione
Sistema chiuso: in grado di interagire solamente con sistemi «proprietari» (es. venduti dal medesimo produttore di HW/SW). A volte sistemi dello stesso produttore non sono nemmeno in grado di interagire, l'interoperabilità è KO.
Definizione mediante una normativa internazionale:
Standard ISO/IEC 7498-1: "Basic Reference Model: The Basic Model". Normativa seguita da una serie di standard relativi al modello OSI in generale e riguardanti aspetti di sicurezza, terminologia, e aspetti di gestione
Raccomandazioni ITU-T serie X.200: Stessi criteri dello standard ISO/IEC 7498-1. Ogni aspetto normato da una specifica raccomandazione corrisponde ad uno standard ISO
Enti Normativi internazionali
International Electrotechnical Commissiion (IEC): definisce standard in materia di ingegneria elettrica ed elettronic;
International Organization for Standardization (ISO): federazione di circa 165 enti nazionali (ES: Ente Italiano di Normazione (UNI), American National Standard Institute (ANSI)). Definisce standard su tutto, tranne che in materie di competenza dell’IEC e di Technical Reports, Technical Specifications, Technical Corrigenda, Guides. Gli standard sono accessibili in rete sul sito ufficiale ISO, ma a costi elevati. Gli standard su Information Technology sono Joint Technical Committee ISO/IEC JTC1.
International Telecommunication Union - Telecommunication Standardization Sector (ITU-T): definisce raccomandazioni identificate da una lettera e da un numero progressivo (es. X.25):
F – Non-telephone telecommunication services.
G – Transmission systems and media, digital systems and networks
H – Audiovisual and multimedia systems
I – Integrated services digital network (ISDN)
J – Transmission of television, sound programme and other multimedia signals
T – Terminals for telematic services
X – Data networks and open system communication
Y – Global information infrastructure and Internet protocol aspects.
Sia gli standard ISO sia le raccomandazioni ITU-T sono soggetti a revisioni periodiche. Quando ci si riferisce a un documento, occorre prestare attenzione a quale versione dello standard o della raccomandazione si utilizza, solitamente, ogni versione riporta l’anno in cui è stata emessa.
OSI – Modello a strati
Ad ogni strato (layer) corrisponde un protocollo e ogni strato fornisce e riceve un servizio.
Layer 1 (Physical): responsabile della trasmissione dei bit che rappresentano l’informazione da trasferire direttamente sul mezzo trasmissivo, qualunque esso sia: doppino in rame, fibra ottica, wireless, ecc.;
Layer 2 (Datalink): responsabile del recupero degli errori trasmissivi, solitamente mediante tecniche di ritrasmissione automatica dei dati corrotti;
Layer 3 (Network): si occupa dell’instradamento del traffico tra sistemi differenti all’interno della rete di comunicazione;
Layer 4 (Transport): responsabile del controllo del traffico tra le estremità della comunicazione;
Layer 5 (Session): controllo del dialogo tra le due entità che realizzano il sistema di comunicazione

distribuito;
Layer 6 (Presentation): unificazione della rappresentazione dell'informazione e dei dati scambiati tra le due entità;
Layer 7 (Application): rappresenta il layer massimo, al cui interno è in esecuzione l'applicativo che ha originato l'informazione (sistema sorgente) e l'applicazione in attesa di ottenere questi dati (sistema destinatario).

OSI – Modello funzionale di un generico strato *x*

Ogni strato è costituito da tutti i sottosistemi che si trovano a un medesimo livello gerarchico, uno strato comprende quindi due sottosistemi: uno facente parte del sistema A, l'altro facente parte del sistema B (idem per il caso di sistemi multipli). La parte attiva di un sottosistema: entity e in generale: sottosistema ed entity sono equivalenti. In un sottosistema è possibile avere molteplici entity per strato. Ogni strato di livello *x* fornisce un servizio allo strato superiore di livello $x + 1$ attraverso un'interfaccia. Le entity si interfacciano a vicenda mediante un Service Access Point (SAP). Es. un'entity di livello 4 (Transport) fornisce un servizio su un T-SAP (Transport-SAP) e riceve un servizio attraverso un NET-SAP (Network SAP).
Il dialogo fra i due strati avviene secondo uno scambio di primitive. Interazione tra sistemi sono unicamente tra entity di pari livello (peer entity) secondo regole definite dal protocollo di livello *x*. Il protocollo è l'insieme di (i) formati secondo cui i dati devono essere strutturati e (ii) procedure per scambiarsi i dati. Gli enti di standardizzazione definiscono i differenti protocolli a cui i sistemi devono allinearsi per poter comunicare. La realizzazione delle interfacce da libera scelta per i costruttori, però rispettando funzionalmente le primitive che definiscono il servizio.

OSI – Incapsulamento dei dati

Protocol Data Unit (PDU).
Service Data Unit (SDU): costituisce il payload della PDU ed è ottenuta a partire dalle PDU dei layer superiori.
Il trasferimento di un blocco di informazione ha origine dal layer APP:
Formazione della PDU nel layer x;
Consegna della PDU al layer $y = x - 1$, che viene ricevuta come una SDU;
Incapsulamento nel formato specifico del protocollo e formazione di una nuova PDU;
Decisione sull'azione successiva da eseguire:
Se $y > 2$: esecuzione dal punto 2;
Se $y = 2$: la nuova PDU conterrà delle Protocol Control Information (PCI) aggiuntive, sotto forma di header e trailer;
Se $y = 1$: la PDU viene trasmessa come una sequenza di bit sul mezzo trasmissivo.
La ricezione all'interno del sistema destinatario prevede l'esecuzione di tutti i passaggi in ordine inverso (decapsulamento), sino ad ottenere il blocco di informazione iniziale all'interno dell'applicazione remota.
Le PDU di alcuni layer hanno dei nomi specifici, che li rendono immediatamente riconoscibili: Layer 7 (Application): messaggio (message), Layer 4 (Transport): segmento (segment), Layer 3 (Network): pacchetto (packet) *e* Layer 2 (Datalink): trama (frame).

Sistema intermedio – Relay

End System (ES): sistema in cui sono presenti tutti i layer, dal layer Physical al layer Application
In generale: due ES non sono collegati in modo diretto, ma per il tramite di sistemi intermedi.
Intermediate System (IS): sistema che realizza solamente un sottoinsieme dei layer, dal layer fisico fino ad un determinato strato. Un IS è rappresentato da una doppia pila di entity (e quindi di protocolli) che possono essere differenti tra loro. Entity messe in comunicazione da una funzione di relay. Il modello OSI prevede esplicitamente sistemi intermedi di L3 che siano in grado di smistare il traffico svolgendo funzioni di instradamento (es. nodi di commutazione a pacchetto)

In generale: si ha sempre che i nodi attraversati in una comunicazione sono molteplici a Molteplici IS collegati in serie.

Servizi connection-oriented vs. connection-less

Layer con servizio connection-oriented

Entità coinvolte nel dialogo: peer entity (stesso layer, sistemi differenti).
Un'entity chiamante apre una connessione, accetta la connessione (ma può anche rifiutarla, inizia la fase di trasferimento dati. Al termine dell'interazione, la connessione viene chiusa (fase di svincolo). Il protocollo operante nello strato x deve fornire delle PDU di controllo, oltre alle PDU per il trasferimento dati.
In generale: le PDU definite dal protocollo operante nel layer x contengono un campo identificativo (es. numero progressivo).
Garanzia contro perdita o duplicazione dei dati, ritrasmissione in caso di errore, controllo del throughput.
Differenti definizioni in base al layer di riferimento:
L7: comunicazione di tipo stateful
L3: servizio di circuito virtuale

Layer con servizio connectionless

Entity coinvolte nel dialogo: entity mittente e una (o più) entity di destinazione. Il mittente manda la PDU al/ai destinatario/i, senza accordi preventivi. Ogni PDU è un'entità a sé stante. Non ha relazioni con le PDU precedenti o con le PDU successive. La PDU può essere persa, oppure essere duplicata. Servizio di tipo best effort.
Molteplici possibilità di comunicazione:
Modalità one-to-one: il campo destination address della PDU contiene un indirizzo di tipo unicast;
Modalità one-to-many: il campo destination address della PDU contiene un indirizzo di tipo multicast o broadcast. È ecessario il supporto di questo tipo di comunicazione da parte della rete di telecomunicazione in uso.
Differenti definizioni in base al layer di riferimento:
L7: comunicazione di tipo stateless
L4: servizio a datagramma
Utilizzo di TCP (L4, affidabile) insieme a IP (L3, non affidabile). Non è un'anomalia. Nel caso servano funzionalità di controllo di errore, tali funzioni sono affidate al layer Transport (L4).

Internetworking

Internetworking: definizioni e gergo

Internet (internetwork): insieme di reti collegate da gateway.
Un gateway (o router) equivale ad un IS (o relay system, def. OSI) di L3.
Un host equivale ad un ES (def. OSI) che accede ad un'internet.
Host e router comunicano mediante un protocollo di internetworking (routed protocol) che non è conosciuto dalle reti intermedie.
I router sono dei nodi di commutazione, collegati non da linee trasmissive, ma da reti. Un'internet è una «rete di reti».
Protocolli di internetworking definiti all'interno di sistemi operativi per Local Area Network (LAN).
Scopo: estendere il servizio su base geografica
Esempi:
Internetwork Datagram Protocol (IDP), all'interno del sistema Xerox Network Services (XNS);
Internetwork Packet eXchange (IPX), all'interno del sistema Netware (Novell, Inc.);
DECnet Routing Protocol (DRP), all'interno delle reti DECnet (Digital).
Sono tutti protocolli connectionless e sono stati superati dall'introduzione della protocol suite TCP/IP.

Internet in pillole

Internet: è l'internet cresciuta attorno alla rete ARPANET (la prima rete a commutazione di pacchetto). I router sono collegati da physical network.
L3 (Network): protocollo IP di internetworking.
L4 (Transport): gli host comunicano tramite TCP oppure UDP.
Stack operativo: TCP/IP oppure UDP/IP.
Il Protocollo IP fornisce il trasferimento di datagrammi attraverso una internet da una sorgente a una destinazione.
Sorgente e destinazione: identificate da indirizzi di lunghezza fissa (indirizzo IP). Provvede alla frammentazione e riassemblaggio di datagrammi che risultino troppo lunghi per attraversare certe reti intermedie.
Esistono 2 versioni del protocollo IP: IP version 4 (IPv4): spazio di indirizzamento 2^{32} e IP version 6 (IPv6): spazio di indirizzamento 2^{128}.
Il *Protocollo TCP* fornisce il trasferimento di un flusso di ottetti (byte) in modo sequenziale e garantito, provvede ad effettuare un controllo di flusso (evitando che il destinatario possa essere "inondato" da un flusso informativo eccessivo, che non è in grado di gestire). Flusso di ottetti emesso dal mittente: organizzato in PDU chiamate segmenti. In generale: l'impacchettamento non è controllato dall'utente. Flusso di ottetti ricevuti dal destinatario: ricostruiti partendo dai segmenti. Instaurazione della connessione: three-way handshake.
Il *Protocollo UDP* si occupa di impacchettare un blocco di dati in un datagramma e di trasferirlo al destinatario. La sua funzione principale è lo smistamento del traffico a diversi applicativi presenti sullo stesso host (tramite il concetto di porte). È un protocollo di tipo connectionless, senza ACK di conferma.

Enti normativi internazionali relativi ad Internet

Internet Engineering Task Force (IETF) è dedicata agli aspetti tecnici di Internet, comprende membri non istituzionali di varie nazioni e produce varie categorie di documenti, i più importanti dei quali sono le Request For Comments (RFC).
Internet Society (ISOC) promuove e coordina l'uso di Internet, supporta e promuove le attività degli enti tecnici di standardizzazione: IETF, Internet Architecture Board (IAB), Internet Engineering Steering Group (IESG), Internet Research Task Force (IRTF).
Internet Assigned Number Authority (IANA) gestisce i piani di numerazione di Internet (es. assegna gli indirizzi IPv4/IPv6)
World Wide Web Consortium (W3C) è dedicato agli aspetti di standardizzazione delle applicazioni di tipo Web.

Request For Comments (RFC)

Riporta informazioni o specifiche riguardanti nuove ricerche, innovazioni e metodologie in ambito Internet. Ogni RFC ha un numero progressivo
RFC1, «Host Software», S. Crocker, Apr. 7, 1969
RFC9003, «Extended BGP Administrative Shutdown Communication», J. Snijders et al., Jan. 2021
Una RFC, una volta emessa, non può più essere modificata. Può essere resa obsoleta da una successiva RFC. Può essere integrata da una successiva RFC.
Ogni RFC ha uno status:

Proposed Standard	Strandard track
Draft Standard	Strandard track
Standard	Strandard track
Experimental	
Informational	
Historic.	
Best Common Practice	

Per sapere se una RFC è ancora valida consultare l'rfc-index
Documenti ritenuti adatti ad essere Internet Standard:
Proposed Standard (~2606): Specifica sufficientemente stabile, con un sufficiente feedback dagli sviluppatori, ed un certo interesse nella comunità Internet; tuttavia un developer deve trattare questa specifica come una specifica ancora immatura;
Draft Standard (~128): Documento ritenuto uno standard sufficientemente maturo, richiedente però la presenza di almeno due implementazioni che abbiano dimostrato la loro interoperabilità;
Standard (~112): Richiede la presenza di un numero significativo di implementazioni, e una notevole esperienza dimostrata dagli utenti su questo standard; riceve un numero progressivo nella lista degli standard.
Documenti ritenuti non adatti ad essere Internet Standard:
Experimental (~416): Documento relativo a qualcosa ancora in fase di ricerca; viene pubblicata a titolo informativo senza che ci sia la pretesa di farlo diventare standard a breve; può essere il risultato prodotto da un gruppo di lavoro Internet, oppure un contributo individuale;
Informational (~2168): Documento di tipo informativo su un determinato argomento; può non disporre di un particolare consenso nella comunità Internet e non rappresenta una raccomandazione di sorta;
Historic (~277): Standard ormai completamente rimpiazzati da nuove specifiche, oppure in disuso;
Best Common Practice (~233): Documenti che contengono suggerimenti o consigli su comportamenti o configurazioni.
Modelli di RFC:
RFC768, «User Datagram Protocol», J. Postel, Aug. 28, 1980;
RFC793, «Transmission Control Protocol», Information Sciences Institute, Univ. of Southern California, Sep. 1981;
RFC791, «Internet Protocol», Information Sciences Institute, Univ. of Southern California, Sep. 1981;
RFC1883, «Internet Protocol, Version 6 (IPv6)», S. Deering, R. Hinden, Dec. 1995;
RFC1945, «Hypertext Transfer Protocol -- HTTP/1.0», T. Berners-Lee et al., May 1996;
RFC2068, «Hypertext Transfer Protocol -- HTTP/1.1», R. Fielding et al., Jan. 1997;
RFC7540, «Hypertext Transfer Protocol Version 2 (HTTP/2)», M. Belshe et al., May 2015;
RFC7252, «The Constrained Application Protocol (CoAP)», Z. Shelby et al., Jun. 2014.

Standard di internet

Non tutte le RFC sono di carattere tecnico:
RFC1149, «A Standard for the Transmission of IP Datagrams on Avian Carriers»;
RFC1924, «A Compact Representation of IPv6 Addresses»;
RFC2549, «IP over Avian Carriers with Quality of Service»;
RFC2795, «The Infinite Monkey Protocol Suite (IMPS)»;
RFC3251, «Electricity over IP»;
RFC3252, «Binary Lexical Octet Ad-hoc Transport»;
RFC4824, «The Transmission of IP Datagrams over the Semaphore Flag Signaling System (SFSS)»;
RFC5514, «IPv6 over Social Networks»;
RFC6592, «The Null Packet»;
RFC7168, «The Hyper Text Coffee Pot Control Protocol for Tea Efflux Appliances (HTCPCP-TEA)»;
RFC8565, «Hypertext Jeopardy Protocol (HTJP/1.0)».

Protocolli in internet

Oltre ai protocolli dei quattro strati inferiori utilizzati dai processi per comunicare fra loro, ne esistono altri di servizio (non rappresentati in figura) che non interfacciano direttamente i protocolli applicativi, ma che sono essenziali per il funzionamento di Internet.
I più comuni protocolli di servizio sono Internet Control Message Protocol (ICMP) e Address Resolution Protocol (ARP):

ICMP: obbligatorio, nel senso che ogni nodo (host o router) di Internet è tenuto a gestirlo. Un messaggio ICMP (che viene utilizzato ad esempio tutte le volte che si usa la utility ping) è incapsulato in un datagramma IP. Quindi ICMP è considerato un protocollo di L4, anche se non ha funzioni di trasporto;
ARP: serve solamente nel caso (peraltro frequentissimo) che il nodo sia collegato ad una rete LAN. Un messaggio ARP (che viene utilizzato ad esempio tutte le volte che un computer collegato a una LAN parte da freddo e deve comunicare con l'esterno) è incapsulato in un frame della LAN (tipicamente Ethernet). Quindi ARP può essere considerato un protocollo di L3 che, durante certe fasi della comunicazione, viene a sostituire IP.

Processi e porte

Ogni porta corrisponde ad uno specifico processo.
Attraverso una porta possono comunicare più processi remoti.
Associazione fra processi univocamente determinata dalla tupla.
<protocol; $remote_{IP}$; $remote_{PORT}$; $local_{IP}$; $local_{PORT}$>
Il campo PORT del segmento TCP (o del datagramma UDP) serve per distinguere fra diversi processi applicativi che risiedono in uno stesso host.
Dato che il campo è di 8 bit, si ha un massimo (teorico) di 64k diversi processi TCP presenti su un host, più altri 64k processi UDP.

Il modello client/server

Client e server sono due processi

Se la comunicazione è connectionless (protocollo UDP) l'associazione esiste solo per la durata di una transazione, ossia dall'istante in cui il server riceve il messaggio di richiesta dal client, all'istante in cui il client riceve il relativo messaggio di risposta. Una successiva richiesta/risposta fra la stessa coppia client/server fa parte di una nuova associazione.
Se la comunicazione è connection oriented (protocollo TCP) l'associazione è «trasportata» da una connessione TCP. L'associazione esiste dall'istante in cui il server accetta la connessione TCP all'istante in cui ha inizio (da parte del client oppure del server) la procedura di svincolo della connessione. Il chiamante della connessione TCP è sempre il client, il chiamato è sempre il server.
I due processi associati (ossia che dialogano tramite l'associazione) sono due peer entity (o semplicemente peer) di livello applicativo.

Trasporto connectionless client/server

Protocollo di trasporto: UDP (server UDP).
Si suppone che al server si acceda in modo connectionless e che l'applicativo sia di tipo stateless.
Tipicamente il server è sempre attivo e viene messo a disposizione dell'utenza all'atto del boot dell'host su cui si trova. Il fatto che il servizio sia presente corrisponde al fatto che sulla porta X corrispondente è presente un processo -> La porta X è «aperta»
La transazione inizia sempre dal client che effettua una richiesta al server. La richiesta è effettuata dal processo client scrivendo (operazione write) un messaggio di richiesta nella socket locale.
Socket: funzione, fornita tipicamente dal sistema operativo, che realizza un'Application Program Interface (API)
Il trasferimento del messaggio di RESPONSE avviene mediante la stessa sequenza di operazioni, questa volta nel verso server -> client.
In definitiva: la transazione prevede uno scambio di due soli messaggi, senza alcuna fase preliminare di inizializzazione. Il trasporto UDP è adatto poiché comporta un overhead minimo. La mancanza di affidabilità può causare inconvenienti e ad essa deve sopperire l'applicativo, di solito con meccanismi di timeout (se non riceve una risposta entro un certo tempo il client ripete la richiesta).

Trasporto connection-oriented client/server

La transazione inizia sempre dal client che tuttavia, prima di mandare la richiesta al server, deve

assicurarsi che sia presente una connessione di trasporto. Nell'esempio la connessione non esiste e l'applicativo del client deve richiederla alla propria entity di trasporto, la quale stabilisce la connessione mediante il meccanismo three-way- handshaking.
Come nel caso precedente avviene il trasferimento del messaggio di RESPONSE.
Trasporto affidabile, quindi non occorre che l'applicativo preveda dei meccanismi di recupero di eventuali messaggi persi, inoltre (a differenza del caso precedente) il messaggio di RESPONSE può essere lungo e trasportato da un flusso di più segmenti TCP.

Well-known services e numeri di porta

Indipendentemente dall'host che lo fornisce, un servizio è identificato univocamente dalla coppia <protocol; port>
Porte suddivise in tre intervalli:
0-1023: well-known ports
1024-49151: registered ports
49152-65535: dynamic/private ports
Assegnate dalla IANA e pubblicati nella RFC1700.

01 Domain Name System (DNS)

Domain Name System (DNS)

Domain Name Space

Ad ogni nodo dell'albero sono associate una (o più) stringhe alfanumeriche che, concatenate, formano un nome di dominio (domain name).
RFC1034
A domain is identified by a domain name, and consists of that part of the domain name space that is at or below the domain name which specifies the domain. A domain is a subdomain of another domain if it is contained within that domain. This relationship can be tested by seeing if the subdomain's name ends with the containing domain's name. For example, A.B.C.D is a subdomain of B.C.D, C.D, D, and ".".
È l'insieme di tutti i nomi di dominio possibili.
Spazio unico in tutta Internet, con struttura gerarchica ad albero.
La radice (root) non ha nome, ma viene identificata da "."
com, it, xyz, edu, … sono Top Level Domain (TLD). Un TLD è un dominio che è sottodominio solamente di root
In genere, le foglie dell'albero corrispondono a singoli host.
Essendo la struttura connessa e ad albero: da qualsiasi nodo è possibile raggiungere qualsiasi altro nodo e il cammino fra due nodi è unico.

Domain Names

edu, com, it, dia, unipr, … sono label (stringhe solitamente alfanumeriche, inclusive anche del carattere "-")
Domain Name: concatenazione di label
La label di destra è la più importante.
unipr.it. (unipr.it), caio.cce.unipr.it, bario.tlc.unipr.it sono Fully Qualified Domain Name (FQDN) unici
unipr.it = UNIPR.IT = UniPR.it = … (case insensitive)
RFC1035: prescrive che il primo carattere di una label sia alfabetico.
Limitazione tolta da RFC1123 e RFC2181.
In definitiva la sintassi dei nomi di dominio prescrive unicamente che:
Ogni label sia lunga da 1 a 63 bytes;
Il FQDN non debba superare complessivamente i 255 bytes, compresi i separatori.
Il FQDN di lunghezza zero corrisponde al root dell'albero.
Una generica sequenza binaria potrebbe costituire una label.

È possibile avere label che iniziano con caratteri numerici, comunemente poco usate e talora ritenute invalide.

Domain Name System (DNS)

Database distribuito composto da differenti entità:

Domain Name Space;

Name Server (NS): Processo server in ascolto sulla porta 53 (TCP/UDP). TCP 53: utilizzata per operazioni di zone transfer. UDP 53: utilizzata per le interrogazioni normali. Gestione dei dati. Può delegare altri NS per la risoluzione di una richiesta;

Resolver: Processo client (routine del sistema operativo. Interrogazione dei NS. Quando accede a un altro NS, un NS agisce come un client.

Resource Record DNS

Codice	Descrizione	Funzione
A	Record di indirizzo	Restituisce un indirizzo IPv4 a 32 bit, normalmente utilizzato per collegare un nome host al suo indirizzo IPv4.
AAAA	Record di indirizzo IPv6	Restituisce un indirizzo IPv6 a 128 bit, normalmente utilizzato per collegare un nome host al suo indirizzo IPv6.
CNAME	Record di nome canonico	Permette di collegare un nome DNS ad un altro. La risoluzione continuerà con il nuovo nome indicato dal record CNAME. Questa funzione è molto utile quando, ad esempio, sullo stesso server sono disponibili più servizi come FTP, HTTP, ecc. operanti su porte differenti. Ciascun servizio potrà avere il suo riferimento DNS (ad esempio ftp.example.com. e www.example.com.). È molto utile anche quando sullo stesso server ci sono più istanze Web con nomi differenti ma stesso indirizzo. Questa funzione richiede, però, il supporto da parte del server di identità multiple con riconoscimento dell'intestazione HTTP.
MX	Server di posta	Collega un nome di dominio ad una lista di server di posta autorevoli per quel dominio. I record indicano anche la preferenza di un server rispetto ad un altro.
NS	Riferimento ai server DNS	Delega una zona DNS ad essere gestita da un server DNS autorevole per quel nome di dominio.
PTR	Record puntatore	Puntatore ad un nome canonico utilizzato per la risoluzione DNS inversa. Inserendo un record PTR per un nome canonico di dominio nella zona `in-addr.arpa.` (`ip6.arpa.` nel caso di IPv6) fa sì che si possa risalire al nome host dal suo indirizzo IP.
SOA	Record di start of authority	Restituisce informazioni autorevoli sulla zona DNS, incluso il server DNS principale, l'e-mail dell'amministratore, il numero seriale del dominio (utile per sapere se i dati della zona sono stati variati) e diversi timer che regolano la frequenza di trasferimento e la durata di validità dei record.
SPF	Record SPF	Specifica dati relativi al protocollo SPF. In alternativa questi dati possono essere inseriti in un record TXT.
SRV	Localizzatore di servizi	Un servizio generalizzato di ricerca di servizi molto utile per nuovi protocolli per non dover aggiungere tipi di record specializzati come ad esempio il record MX.
TXT	Record di testo	Era stato pensato per aggiungere commenti leggibili ad un record DNS. Al momento è utilizzato anche per trasferire informazioni di sicurezza in accordo alla RFC1464.

Record «CODICE» NAME (Owner Name), Type, Data.

Record «A» habana.cedi.unipr.it, A, 160.78.31.139

Record «NS» unipr.it, NS, caio.cce.unipr.it

Record «PTR» 139.31.78.160.in-addr.arpa., PTR, habana.cedi.unipr.it

Record «MX» unipr.it, MX, 10 caio.cce.unipr.it

Un NS gestisce un master file, che è un insieme di Resource Record (RR) in formato testuale.

Owner Name è il nodo a cui appartiene il RR, è sempre un domain name.

Type identifica il tipo di RR richiesto.

Data è il contenuto del RR.

Il tipo PTR è un puntatore ad un record di tipo A: il campo Data contiene il domain name del record puntato, il nome del record PTR è quello di un nodo fittizio appartenente al dominio in-addr.arpa. Dominio suddiviso in ulteriori quattro livelli di sottodominio, ognuno corrispondente a un campo dell'indirizzo IP contenuto nel campo Data del record puntato e presi in ordine invertito (il meno significativo è quello più a sinistra).

Scopo: effettuare una name resolution inversa (ottenere da un indirizzo IP un domain name) mediante una query diretta.

Il tipo MX contiene il campo Data composto da due sottocampi: preferenza (campo numerico) e host (domain name). In presenza di record multipli: la preferenza è sui record con valori più bassi.

Assegnazione dei domain name

Internet Corporation for Assigned Names and Numbers (ICANN), ente non-profit. In precedenza

queste funzioni svolte da InterNIC.
Accredita i Registrar:
Autorizzati a vendere, in regime di concorrenza, i nomi di dominio;
Corrispondono all'ICANN una quota in base alle operazioni effettuate;
Trasferimento di un dominio: processo che cambia il registrar designato in precedenza;
In genere: gli Internet Service Provider (ISP) non sono Registrar.
Gestione dello spazio di indirizzi IP ed assegnazione degli identificatori di protocollo.
Con riguardo al DNS, ICANN assegna determinati TLD:
Di tipo generico (gTLD, es. com, edu, gov, mil, org, …);
Di tipo country code (ccTLD, es. it, eu, fr, br, ca, uk, …).

Regional Internet Registries (RIR)

Cinque organizzazioni che gestiscono l'allocazione dei domain name
per Internet nelle regioni del mondo:
American Registry for Internet Numbers (ARIN): North America;
RIPE Network Coordination Centre (RIPE NCC): Europe, Middle East and Central Asia;
Asia-Pacific Network Information Centre (APNIC): Asia and the Pacific region;
Latin American and Caribbean Internet Address Registry (LACNIC): Latin America and the Caribbean region;
African Network Information Centre (AfriNIC): Africa.
I soci dei RIR sono essenzialmente gli Internet Service Provider (ISP)

Accesso ai Name Server

Il Resolver fa delle richieste (query). Operazione fondamentale: da un domain name ottenere un indirizzo IP (name resolution). Esiste anche l'operazione inversa, da indirizzo IP ottenere un domain name (inverse name resolution). La query viene fatta al NS del dominio in esame. Nel caso in cui il NS non abbia l'informazione richiesta: Ricorsione (recursive query) oppure Iterazione.

Name resolution

Risoluzione ricorsiva: il Resolver emette una query di tipo ricorsivo. Il NS (se non ha l'informazione) la richiede ad un NS di livello superiore, il quale può rivolgersi a un altro NS (di livello superiore o non), ecc. Alla fine, al Resolver viene consegnata la risposta;
Risoluzione iterativa: il Resolver emette una query non ricorsiva. Il NS (se non ha l'informazione) restituisce la miglior informazione che possiede.
Il Resolver emette una nuova query ad un nuovo NS, ecc.
I Root Name Server conoscono i NS per tutti i TLD.
In genere, i Resolver degli end system sono processi piuttosto semplici, in grado di contattare un solo NS predefinito. Emettono sempre delle query ricorsive.
A livello più alto, dove i NS contattano altri NS, si usano anche query iterative.
I NS di alto livello spesso rifiutano query ricorsive.

Risoluzione di un domain name: file HOSTS

File di testo presente solitamente su piattaforme Unix e Windows.
Originariamente era l'unico modo per effettuare risoluzioni dirette di
nomi. Limite: ogni nuovo host aggiunto nella rete di comunicazione richiede una modifica di questo file su tutti gli host.
Con l'avvento del DNS, il file HOSTS è rimasto e viene solitamente interrogato dai Resolver prima di accedere al NS. Consente di forzare determinati domain name.
Indirizzo convenzionale 127.0.0.1. Host locale a cui corrisponde il nome di dominio localhost.
Convenzione universalmente diffusa e serve ad esempio a far comunicare tramite la pila di protocolli TCP/IP due processi che risiedono sullo stesso host.
Nota: indirizzi della classe privata (quindi non unici in tutta Internet) non devono essere comunicati

come raggiungibili dall'esterno del dominio in cui sono definiti tramite interrogazioni DNS né annunciati esternamente tramite protocolli di routing.
Nota: il file HOSTS viene talvolta utilizzato per evitare siti Web
pericolosi o che si vogliono comunque bloccare.

Domain Name Space – Authoritative NS

Ogni TLD è associato ad uno o più authoritative NS.
Le informazioni possedute da ogni authoritative NS sono valide per l'area di competenza dello specifico TLD.
Ogni authoritative NS non deve conoscere tutti gli host esistenti in tutti i sottodomini. Delega ad altri NS.
Ogni sottodominio del TLD è affidato ad un NS che ha autorità su di esso.

Domain Name Space – Start of Authority (SOA)

Authoritative NS	dns.nic.it.
Start of Authority (SOA)	caio.cce.unipr.it. server.fis.unipr.it.

Autorità e zone

Ogni dominio ha sempre almeno due server che hanno autorità su di esso: Server primario e Server secondario. Server primario e server secondario vengono allineati periodicamente. Server secondario effettua una connessione TCP al server primario. Per acquisire una zona. Per effettuare controlli di aggiornamento tramite il Zone Transfer Protocol (AXFR, RFC5936). I server secondari designati per un dominio possono essere molteplici. Un NS ha autorità su una zona. Zona: insieme di uno o più domini.

Struttura di un messaggio DNS

Header: lunghezza fissa (12 bytes):
bytes 1-2: Transaction Identifier – serve per mantenere un'associazione tra una query ed una response
bytes 3-4: Flags.
QR – indica se il messaggio è una query oppure una response
OPCODE – 0: standard query (consiste nell'ottenere il record fornendo al server l'owner del record); 1: inverse query (consiste nell'ottenere il record fornendo al server il contenuto (campo DATA) del record; di solito non è onorata dai server); 2: server status request
AA – Authoritative Answer: in una response, indica se il server DNS è authoritative per l'host richiesto
TC – Truncation: indica se il messaggio è stato troncato a causa di un'eccessiva lunghezza
RD – Recursion Desired: indica se il client ha espressamente richiesto una ricerca ricorsiva
RA – Recursion Available: in una response, indica se il server DNS che sta rispondendo supporta la ricorsione
Z – campo riservato per utilizzi futuri
RESPCODE – Response Code: contiene il codice di risposta: 0: no error, …
bytes 4-5: QDCOUNT
bytes 7-8: ANCOUNT
bytes 9-10: NSCOUNT
bytes 11-12: ARCOUNT

Body: lunghezza variabile

Formato di un Resource Record in un messaggio DNS

NAME: contiene il FQDN del nodo all'interno dell'albero DNS;
TYPE: identifica il tipo di record restituito;
CLASS: impostata al valore «1» (Internet) per record che coinvolgano hostname e IP address;
TTL: fissato dal gestore del NS, indica la durata (in secondi) per cui il ricevente è autorizzato a tenere

nella propria cache il record; passato questo tempo, il record scade ed occorre richiederne una nuova copia al NS;
RLENGTH: specifica la lunghezza (in byte) del campo RDATA;
RDATA: stringa di byte che descrive la risorsa, il cui formato dipende da TYPE e CLASS.

Protocollo WHOIS

RFC3912 «WHOIS Protocol Specification»
Protocollo di tipo query/response operante mediante il protocollo TCP. Fornisce un servizio di tipo white pages per dare informazioni su nomi di dominio registrati. Esistono anche programmi con interfaccia Web (utilizzabili tramite browser). Alcuni Registrar potrebbero porre delle limitazioni sulle informazioni fornite in risposta ad una query. Informazioni solamente su domini di propria competenza. Informazioni solamente su domini appartenenti ad un certo TLD.
Il *server* attende sulla porta TCP 43.
Il *client* emette una richiesta in formato umanamente leggibile (stringa di testo) e riceve una risposta in formato leggibile.

Microsoft Name Server

NetBIOS Name Service (NBNS)

RFC1001 «Protocol standard for a NetBIOS service on a TCP/UDP transport: Concepts and methods».
RFC1002 «Protocol standard for a NetBIOS service on a TCP/UDP transport: Detailed specifications».
Servizio nato per lavorare in un ambiente LAN (rete a mezzo condiviso) e derivato da Microsoft LAN Manager (OS per LAN).
NetBIOS è l'interfaccia API (interfaccia fra layer Session/Application) dell'architettura LAN Manager, in ascolto sulla porta UDP 137.
Prima versione: chi riceve il nome di dominio lo associa al MAC Address corrispondente, lo inserisce in una name cache e lo utilizza per effettuare una risoluzione dinamica.
Porting su TCP/IP: la name resolution restituisce un indirizzo IP (invece di un MAC Address).
Messaggio NBNS: ha lo stesso formato del messaggio DNS.
Differenza: name resolution di tipo peer-to-peer (senza un server centralizzato) con query emesse in modo broadcast.

Risoluzione NetBIOS: file LMHOSTS

LMHOSTS contiene la corrispondenza fra indirizzi (fisici o IP) e nomi simbolici ed è, per i nomi NetBIOS, l'analogo del file HOSTS per i nomi di dominio Internet.
Tuttavia viene usato in modo differente. Un nodo Windows risolve i nomi usando la name cache (riempita mediante dynamic resolution: messaggio broadcast oppure query a name server).
Parola chiave PRE: il nome viene precaricato nella name cache quando l'host parte, altrimenti viene utilizzato solo se fallisce la risoluzione dinamica.
Windows Internet Name Service (WINS): Classificazione di un nodo (sulla base della dynamic resolution operata).
B-node: utilizza una comunicazione broadcast per risolvere i nomi.
P-node: utilizza una comunicazione point-to-point con un server NetBIOS (ad esempio un WINS server).
M-node: utilizza come prima opzione una comunicazione broadcast (B-node), quindi una query diretta (P-node) se il broadcast fallisce.
H-node (hybrid node): utilizza come prima opzione una query diretta (P-node), quindi broadcast (B-node) se il name server non risponde o non ha il nome nella base dati WINS.
Il comando CLI Windows ipconfig /all fornisce informazioni sulla configurazione di rete dell'host, fra cui il tipo di nodo.

Estensione della NetBIOS Name Resolution

Problema di questo protocollo: non funziona attraverso i router.

Soluzioni: File LMHOSTS replicato: Gestione manuale, Molto problematico in presenza della funzionalità di Dynamic Host Configuration Protocol (DHCP) attiva.
WINS: Sistema client/server (WINS Server), Integrato con DNS.
In genere, i nodi Windows seguono un determinato ordine per la name resolution:
1. File HOSTS
2. Query DNS
3. Risoluzione NetBIOS.

Problemi del DNS:
DNS lavora con messaggi di tipo plain text;
Assenza di sessioni, utilizzando UDP;
Struttura ad albero con delega delle richieste;
Ogni entità è responsabile per una parte limitata dell'intero dominio operativo;
I Resolver possono essere vittime di attacchi, dirottamento del traffico, ed errori;
Necessario un certo grado di fiducia tra le entità.

Domain Name System Security Extensions (DNSSEC)
RFC4033 «DNS Security Introduction and Requirements».
Introduce nuovi layer in DNS per rendere verificabili le richieste, nuovi record type, il concetto di Public Key Infrastructure (PKI) e il concetto di chain of trust per la validazione dei dati.

02 Telnet SSH

Protocollo Telephone Network (TELNET)

Protocollo Telephone Network (TELNET)
Progettato per consentire una connessione interattiva ad un computer remoto, fornendo all'operatore una cosiddetta «shell remota». Nato in un contesto di centri di elaborazione basati su un elaboratore centrale (host) cui erano collegati terminali a carattere.
RFC1282: remote login (rlogin), che può essere considerata una variante semplificata di TELNET. Come TELNET, rlogin ha lo scopo di mettere in comunicazione un utente con un computer remoto. rlogin non è uno standard di Internet. A differenza di TELNET, rlogin non realizza però prestazioni di Network Virtual Terminal (NVT), poiché è stato ideato per consentire una connessione solamente con sistemi UNIX (quindi compatibili tra loro).

Login su host remoto
Scenario di riferimento: un utente U3, dotato di un proprio terminale T3 situato presso un centro di elaborazione A, vuole accedere ad un host H4 situato in un altro centro di elaborazione B.
Prima configurazione: configurazione di riferimento in assenza di una rete che colleghi i due host. Essendo prima dell'avvento di ARPANET, l'unica infrastruttura utilizzabile era la rete telefonica che metteva a disposizione un circuito fisico di collegamento. Occorre utilizzare un terminale apposito, collegato non ad una porta dell'host locale H1, ma alla rete telefonica tramite modem.
Seconda configurazione: configurazione di riferimento dopo l'avvento di ARPANET che collega i vari host. La stessa prestazione può essere ottenuta mediante un circuito virtuale (associazione) realizzato dalle due peer entity TELNET client (C3), attivo sull'host H3 dell'utente, e TELNET server (S4) attivo sull'host H4 di destinazione. In entrambi i casi l'obiettivo è di rendere disponibile una «shell remota» all'utente U3.

Caratteristiche fondamentali
Scopo: fornire un dispositivo di comunicazione bidirezionale, di uso generale.
Uso primario: remote login, tramite terminale alfanumerico, ad un host remoto.
TELNET client: Attivato (su un host Unix) dal comando telnet nella CLI.
Comportamento di default: connessione alla porta TCP 23 dell'host remoto.
Tre concetti di base: Network Virtual Terminal (NVT), Negoziazione opzioni e Comportamento simmetrico.

TELNET sulle macchine attuali
Sui sistemi UNIX esiste un programma, richiamabile da CLI, chiamato telnet che è un client TELNET. Spesso è possibile attivare un daemon, generalmente denominato telnetd, che corrisponde ad un server TELNET.
Network Virtual Terminal (NVT)
NVT è un terminale a caratteri (non grafico) dotato di una tastiera e di una stampante. La stampante è spesso emulata dal monitor di un terminale fisico. Sia il client che il server devono comportarsi in modo da apparire come un NVT. Di base NVT funziona in modalità two-way alternate (half-duplex, controllato dal comando GO-AHEAD). Un carattere ricevuto sulla connessione TCP viene stampato sulla stampante. Un carattere battuto sulla tastiera viene trasmesso sulla connessione TCP e in aggiunta può essere stampato sulla stampante locale (eco locale). Se il carattere è inviato solamente in linea e l'host remoto lo rimanda indietro, il carattere viene sempre stampato sulla stampante locale e si ha eco remota. Se sono attive sia eco locale che eco remota un carattere battuto una volta viene stampato due volte sulla stampante locale (tutti i caratteri raddoppiati).
GO-AHEAD: comando mediante il quale chi ha terminato l'invio dei dati rilascia il permesso all'altra entità di mandarli a sua volta. Analogo alla parola convenzionale "passo" usata in comunicazioni vocali via radio (PTT).
TELNET: Negoziazione delle opzioni
Sia server che client possono richiedere prestazioni aggiuntive, oltre a quelle di base di un NVT. Concesse abilitando determinate opzioni. Le opzioni devono essere negoziate solitamente all'inizio della connessione TCP. Sia server che client possono chiedere di attivare un'opzione (simmetria di funzionamento). Se l'altra parte fornisce un riscontro positivo, allora l'opzione è attivata. La richiesta di un'opzione sconosciuta è ignorata, e questo vale come riscontro negativo. È possibile modificare opzioni anche durante il funzionamento.
Codifica dell'informazione
NVT usa l'alfabeto USASCII (7 bit) trasportato in ottetti.
Caratteri di controllo utilizzati: Carriage Return (CR) e Line Feed (LF) è <CRLF>
Bit più significativo pari a 0 (codice < 128): rappresenta un carattere dati.
Bit più significativo pari a 1 (128 < codice < 255): rappresenta un carattere di controllo.
Un messaggio di negoziazione è composto da tre byte:
Un byte che rappresenta la sequenza Interpret As Command (IAC): codice decimale 255;
Un byte per identificare l'azione.
Byte con codice decimale 251: WILL
Byte con codice decimale 252: WON'T
Un byte per il codice del comando.
Un messaggio non di negoziazione è composto da due soli byte, di cui il primo è sempre IAC.
Modalità di trasmissione dati
Default Mode: Utilizzato nel caso in cui nessuna altra modalità sia invocata.
Character Mode: Ogni carattere viene spedito al server, ma il server potrebbe ritardare l'invio della risposta. Crea un certo overhead, siccome devono essere spediti 3 segmenti TCP per ogni carattere.
Line Mode: Il client accumula i caratteri all'interno di una riga di testo, che poi invia per intero al server.
Sintassi per la negoziazione delle opzioni
WILL XXX --- Il mittente desidera attivare l'opzione XXX.
DO XXX --- Il ricevente restituisce riscontro positivo sull'attivazione dell'opzione XXX.
DON'T XXX --- Il ricevente restituisce riscontro negativo sull'attivazione dell'opzione XXX.
DO XXX --- Il mittente richiede al ricevente di attivare l'opzione XXX.
WILL XXX --- Il ricevente restituisce riscontro positivo sull'attivazione dell'opzione XXX.

WON'T XXX --- Il ricevente restituisce riscontro negativo sull'attivazione dell'opzione XXX. XXX è un numero intero che identifica l'opzione che si sta negoziando.

Opzioni TELNET

Ogni opzione è identificata da un codice.

Molte RFC separate specificano varie opzioni: 1 ECHO (RFC857), 3 SUPPRESS GO-AHEAD (RFC858): necessaria per la comunicazione in modalità full- duplex, in caso si abbiano più comandi da inviare senza dover attendere la risposta di ciascun comando, 5 STATUS (RFC859): utile per richiedere all'altra entità in comunicazione lo stato delle opzioni attualmente in uso e 32 TERMINAL SPEED (RFC1079).

Complessivamente almeno 17 RFC riguardanti TELNET.

Secure Shell (SSH)

Secure Shell (SSH)

In modo simile a TELNET, utilizza TCP come protocollo di trasporto. Fornisce un livello di sicurezza superiore rispetto a TELNET. Fornisce servizi aggiuntivi. In ascolto sulla porta TCP 22. Entrambe le entità coinvolte nella comunicazione si autenticano a vicenda. Tutti i messaggi sono cifrati. Utilizzo della crittografia a chiave pubblica. Utilizzato come canale di tunneling sicuro per altre applicazioni: Trasferimento file, Port Forwarding e Virtual Private Network (VPN).

Macchine basate su Unix: Directory nascosta .ssh contiene tutti i files di configurazione e File known_hosts contiene le chiavi pubbliche di tutti i server con cui ci sia stata una connessione SSH precedente.

Esistono due versioni del protocollo SSH: SSH-1 e SSH-2: SSH-2 è stato introdotto per sopperire ad alcuni problemi di SSH1 ed include alcune nuove features; SSH-1 è più popolare di SSH-2 per motivi di licensing.

OpenSSH: versione free di SSH con codice sorgente open source.

SSH – Stack protocollare

Layer di trasporto: Gestisce lo scambio iniziale delle chiavi, Istanzia la crittografia e la compressione dei dati, Gestisce il processo di rinnovo delle chiavi (key re-exchange) a determinate condizioni, Al superamento di un'ora di attività e Al superamento di 1 GB di dati trasferiti.

Layer di autenticazione utente: Gestisce l'autenticazione del client, Gestisce l'autenticazione tramite password e Gestisce l'autenticazione mediante chiave pubblica.

Sicurezza

SSH protegge contro diversi tipi di attacchi:

IP spoofing: in cui un host remoto invia pacchetti che fingono di provenire da un altro host affidabile.

Spoofer sulla rete locale: che può fingere di essere il router per l'accesso all'esterno;

IP source routing: in cui un host può fingere che un pacchetto IP provenga da un altro host affidabile;

DNS spoofing: in cui un utente malintenzionato falsifica i Resource Record del Name Server.

03 Email

E-mail, Struttura dei messaggi

Messaggio di posta elettronica (e-mail)

Un messaggio è composto da una busta (envelope) e da un contenuto (content). La busta serve a consegnare il messaggio a destinazione. Il contenuto è ciò che viene consegnato al destinatario. La busta è usata dai Message Transfer Agent (MTA). Il contenuto è usato dagli User Agent (UA). La busta è descritta nella specifica del protocollo Simple Mail Transfer Protocol (SMTP). L'intestazione (header) del contenuto è generalmente manipolata dai MTA (tipicamente campo Received ed altri).

Struttura di un messaggio

Envelope: Costruita mediante alcuni comandi SMTP. MAIL From: ldavoli@unipr.it e RCPT To: Mario Rossi.

Una mailbox può essere rappresentata mediante differenti formati:

addr-spec: costituito dal nome utente e dal dominio, separati dal carattere @;
angle-addr: corrisponde ad un addr-spec racchiuso tra i caratteri < e >;
name-addr: costituito da un campo display name (es. Mario Rossi) seguito dal campo; angle-addr. Il campo display name deve essere racchiuso tra virgolette doppie (“”) se contiene caratteri speciali (es. “Mario P. Rossi”).
Contenuto: Passato da UA a MTA mediante il comando DATA. Comprende due parti: Header (generato ed utilizzato da UA) e Body (utilizzato dall’utente).

Header e Body

Header: Sequenza di header fields: Il campo deve essere composto di caratteri ASCII stampabili escluso il “:” e deve essere composto di caratteri ASCII esclusi CR e LF.
Body: Sequenza di righe separate da una sequenza <CRLF>
<caratteri ASCII><CRLF> (una riga)
<caratteri ASCII><CRLF> (una riga)
<caratteri ASCII><CRLF> (una riga)
...
<caratteri ASCII><CRLF> (una riga).
I caratteri CR e LF non possono apparire se non accoppiati nella sequenza <CRLF>.
Header separato dal body da una riga vuota (ossia dalla sequenza <CRLF><CRLF>).

Header fields

Campo di tipo Origination Date:
Date: date-time <CRLF>
Specifica la data e l’orario in cui il mittente ha consegnato il messaggio al MTA per essere trasmesso (non quando il mittente ha preparato l’e-mail nel proprio UA).
Formato: Thu, 25 Feb 2021 09:55:06 -0600
L’ultimo parametro identifica la zona, ossia la differenza dal Coordinated Universal Time (UTC). Il segno + indica che l’orario in oggetto supera l’orario UTC (ossia ci si trova ad Est di Greenwich). Il segno – indica che l’orario in oggetto anticipa l’orario UTC (ossia ci si trova ad Ovest di Greenwich).
Campi di tipo Originator:
Esistono più campi di questa categoria.
From: mailbox-list <CRLF>: Specifica il mittente (o i mittenti)
Sender: mailbox <CRLF>: Specifica il mittente che ha effettivamente inviato l’e-mail (presente solamente nel caso in cui questa mailbox sia differente dal valore contenuto nel campo From:).
Reply-To: address-list <CRLF>: Specifica una mailbox a cui inviare una risposta (nel caso in cui mancasse questo campo, come indirizzo per la risposta viene adottato il valore contenuto nel campo From:).
Campi di tipo Destination Address:
To: address-list <CRLF>: Specifica i destinatari principali del messaggio
Cc: address-list <CRLF>: Specifica i destinatari che debbano essere messi in copia per conoscenza (carbon copy)
Bcc: address-list <CRLF>: Specifica i destinatati che debbano essere messi in copia per conoscenza senza farli apparire tra i destinatari visibili del messaggio (blind carbon copy).
address-list: è una sequenza di mailbox separati da “,”.
Nota: i campi di tipo Origination Date, Originator, e Destination Address sono obbligatori, tutti gli altri campi possono essere omessi.
Campi di tipo Identification:
Message-ID: msg-id <CRLF> Identificatore univoco del messaggio
In-Reply-To: 1*msg-id <CRLF>
References: 1*msg-id <CRLF> Generati quando il messaggio è una risposta oppure appartiene ad un

thread .
msg-id deve essere garantito unico a cura del mittente, e ha un formato simile ad un angle-addr con una sintassi di tipo timestamp@domain.
La scrittura 1*msg-id indica che è presente una sequenza di campi msg-id in numero imprecisato, ma non inferiore ad 1. Quando i messaggi passano nel sistema di trasporto vengono spesso modificati, anteponendo degli header addizionali, tipicamente di tipo trace e resent. L'aggiunta di tali campi non cambia l'identità del messaggio.
Campi di tipo Informational:
Subject: string <CRLF> Specifica il soggetto dell'e-mail (opzionale, ma utile per contestualizzare un messaggio in mezzo ad altri all'interno della casella di posta)
Comments: string <CRLF>
Keywords: key1, key2, ... <CRLF>
Non hanno una funzione specifica per un MTA, ma sono intesi solo per leggibilità da parte di un utente umano.
Campi di tipo Resent:
I campi di questa categoria sono aggiunti ogni volta che l'utente reinserisce il messaggio nel sistema di trasporto (Message Transfer System)
Resent-Date: date-time <CRLF>
Resent-From: mailbox-list <CRLF>
Resent-Sender: mailbox <CRLF>
Consentono al messaggio di apparire al destinatario come inviato dal mittente originale, non da chi lo ha reinserito nel MTS. I valori ammessi sono quelli definiti in precedenza anche per altri campi. Nota: non vi è certezza che questi campi siano sempre considerati dai sistemi intermedi durante le operazioni di forward.
Campi di tipo Trace:
Return-Path: path <CRLF> Il valore path è conforme a quanto specificato da RFC2821
Received: list <CRLF>
L'uso di questi campi, per tracciare il percorso seguito dal messaggio, è specificato in RFC2821 (SMTP)
Campi personalizzati (custom):
Tutti i campi il cui inizi con la sequenza X- (es. XMailer) devono essere considerati come campi custom, non definiti da RFC2822.

E-mail Trasferimento di informazione generica, oltre il text-plain (uuencode, MIME)

Inserimento di file binari all'interno di un messaggio

SMTP parla in modalità NVT.
Caratteri leciti: ASCII. Sequenze di controllo obbligatorie (<CRLF>) come separatori di linea. RFC2822 ammette nel body di un messaggio solamente caratteri ASCII.
Problema: trasferimento di un flusso di ottetti generici.
Soluzione: trasformare una generica sequenza di byte in una stringa ASCII «RFC2822-compliant». stringa ASCII: ottavo bit = 0 (128 combinazioni). «RFC2822-compliant»: né <CR> né <LF> isolati.

Alterazione del flusso di byte: uuencode

Da tre byte consecutivi (flusso originale) vengono generati quattro byte (flusso derivato) prendendo sequenze di 6 bit ciascuna. Ogni nuovo byte (in grado di rappresentare un valore intero compreso fra 0 e 63) viene aumentato del valore 32 (HEX: 0x20) in modo da rappresentare un carattere ASCII non di controllo. Ogni nuovo byte rappresenterà un valore 32-95 (0x20-0x5F). Il byte 32 (0x20) viene convertito in 0x60 per evitare il carattere "non stampabile" SP (spazio). Il flusso derivato aumenta di 4/3 (≅ 35%). I byte così ottenuti sono organizzati in linee di testo lunghe al massimo 45 caratteri e precedute da un byte che ne specifica la lunghezza. Siccome solitamente si costruiscono linee di

lunghezza massima, il byte di lunghezza vale 45+32=77, corrispondente alla lettera M. Un file «uuencoded» si riconosce facilmente dal fatto che solitamente ogni linea, esclusa l'ultima, inizia con la lettera M.

Organizzazione di un file "uuencoded"

Il file originario è suddiviso in linee di 45 byte, ognuna delle quali produce una linea derivata di 45x4/3 = 60 caratteri. Ogni linea è preceduta da un byte di lunghezza (anch'esso aumentato di 32). M<…60caratteri…>. Il flusso di byte ottenuto è preceduto e chiuso da informazioni di controllo. Il risultato è un file di tipo text-plain leggibile e modificabile con un comune text editor. Il byte iniziale di lunghezza dice quanti caratteri conteneva il file originario, non quello derivato. Se in una linea si hanno 60 byte derivati, quella originale ne conteneva 45 bytes.

Esempio di generazione di file "uuencoded"

File originario: logo.zip, di 9940 byte

Flusso derivato: 13256 byte 1. 9940 = 3313 x 3 + 1 2. 3313 x 4 + 4 = 13256

File derivato costituito da diverse componenti: Preambolo, 220 linee piene (M), 1 linea breve (H, non tutti utili), H = 9940 - ($\lfloor 9940/45 \rfloor$ x 45) = 40 à 40 + 32 = 72 («H»).

Postambolo. File iniziale diviso in blocchi di 3 byte: ne risulta un byte isolato. Il flusso derivato contiene sempre un numero di byte multiplo di 4, quindi l'ultimo byte isolato genera 4 byte aggiuntivi. Si suddivide il flusso derivato in blocchi di 60 byte (oppure si suddivide il flusso originario in blocchi di 45 byte), ottenendo 220 linee piene, con un avanzo di 52 byte (40 byte nel caso del flusso originario). I byte residui del file originario, aumentati di 32, producono il carattere ASCII "H".

File logo.zip "uuencoded"

begin 666 logo.zip: ultima riga del preambolo, ha un formato standard. I caratteri 666 sono i tre byte che descrivono i permessi di lettura/scrittura/esecuzione del file (convenzione Unix).

Il carattere di chiusura (accento grave) e la linea end rappresentano il postambolo.

I caratteri iniziali di ogni riga (M e H) hanno il significato enunciato in precedenza.

Limitazioni di uuencode

Tipico degli ambienti UNIX. Orientato a file binari (non testi internazionali). Possibili problemi di traduzione con caratteri non «mail safe» che non subiscono alterazioni nei passaggi attraverso sistemi di relay SMTP. Trasparente agli UA, quindi richiede interventi manuali dell'utente finale per la decodifica. Tutti questi problemi sono stati superati con l'introduzione delle Multipurpose Internet Mail Extensions (MIME). Messaggio MIME: normale messaggio RFC2822-compliant identificato da un apposito header. Gli UA che lo riconoscono gestiscono il messaggio in modo appropriato (codifica corretta, separazione allegati, attivazione plug-in).

Multipurpose Internet Mail Extensions (MIME)

RFC2045, «Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies»

RFC2046, «Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types»

RFC2047, «Multipurpose Internet Mail Extensions (MIME) Part Three: Message Header Extensions for Non-ASCII Text»

RFC2048, «Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures»

RFC2049, «Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples»

RFC2231, «MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations»

RFC2387, «The MIME Multipart/Related Content-Type»

RFC3023, «XML Media Types»

RFC3676, «The Text/Plain Format and DelSp Parameters»

RFC3798, «Message Disposition Notification»

MIME – Nuovi header fields

MIME version: identifica la presenza di un messaggio MIME all'interno di un documento
Content-Type: dichiara il tipo di contenuto del messaggio. MIME type: combinazione di type e sub-type. Il type text ha parametri addizionali che indicano la codifica, altri tipi presentano parametri propri. Sebbene definito originariamente per i messaggi di posta elettronica, questo tipo di header e i relativi tipi registrati sono usati spesso in altri contesti.
Content-Transfer-Encoding: specifica come manipolare il messaggio perché sia accettato dal sistema di trasferimento.
Content-ID: deve essere univoco (analogamente al campo message-id).
Content-Description: testo associato al body (es. descrizione del contenuto di un'immagine).
Non tutti questi header sono obbligatori, mentre i primi tre header sono i più utilizzati.

MIME – Media Type

Definiti nel RFC2046
text: testo scritto
image: richiede un display per essere mostrato
audio: richiede un dispositivo audio per essere riprodotto
video: prevede una visualizzazione di immagini in movimento
application: altro tipo di dati
multipart: molteplici tipi tra quelli precedenti, in serie
message: messaggio incapsulato
model: definito da RFC2077
MIME types più comuni e maggiormente utilizzati nelle applicazioni moderne
Tipo model: utilizzato per applicazioni scientifiche, simulazioni e simili:
model/vrml: realtà virtuale;
model/mesh: mesh computazionali per calcolo numerico;
model/iges: tipo proposto per applicazioni CAD/CAM.
Codifica di tipo identity = no encoding: usare trasporto corrispondente:
7bit: trasporto standard SMTP
8bit: servizio ESMTP
binary: non supportato (per il momento).
Servono per indicare il tipo di dati contenuti nel corpo del messaggio:
7bit, 8bit: line-oriented (per non superare la lunghezza massima di linea, divieto di o isolati)
binary: non line-oriented (è ammesso uno stream di byte qualsiasi), ma non è compatibile con il protocollo SMTP.
quoted-printable: utilizza un meccanismo di escape e serve per testi più o meno leggibili prevalentemente a 7 bit, con alcuni caratteri speciali (es. lettere accentate).
base64: usa una tecnica di espansione simile a uuencode e serve per file binari «densi» (es. immagini, codice eseguibile).

MIME – Codifica quoted-printable

Il carattere non-ASCII À è sostituito dal suo codice esadecimale. Il carattere = ha la funzione di escape (ossia segnala la presenza di un carattere codificato). Nel caso in cui il carattere di escape faccia parte del testo, deve essere codificato (ossia sostituito dalla sequenza =3D), anche se è un carattere ASCII stampabile. Nel caso in cui si usasse 8bit come Content-TransferEncoding, il testo potrebbe essere trasferito senza bisogno di codifica.

MIME – Codifica base64

A differenza di uuencode, con la codifica base64 i caratteri possibili sono solamente quelli classificati come caratteri mail safe (sono esclusi tutti i caratteri soggetti a varianti nazionali).
RFC2049, *caratteri mail safe:* A-Z, a-z, 0-9, "'", "(", ")", "+", ",", "-", ".", "/", ":", "=", "?"

Il carattere «pad» serve a gestire file che non sono esatti multipli di 3 byte.
uuencode: questa funzione è svolta dal byte di lunghezza posto all'inizio di ogni linea.

File codificato base64

Il file originario è suddiviso in blocchi di 3 byte (24 bit). Da ogni sequenza di 6 bit si genera un carattere ASCII usando la tabella di conversione: Non ci sono residui: no padding, Avanza 1 byte (8 bit): si producono 2 caratteri + "== ", Avanzano 2 byte (16 bit): si producono 3 caratteri + "=". Alla fine il flusso codificato ha un numero di byte multiplo di 4. Caratteri organizzati in linee di lunghezza massima pari a 76 caratteri. Non si usano byte di lunghezza.

MIME – Codifica multipart

Parte finale dell'header e parte iniziale del body di un messaggio multipart.
boundary usato per dividere le varie parti. Costituito dal parametro dichiarato nell'header boundary preceduto da due caratteri -. Accorgimento adottato per facilitare la ricerca della stringa in determinate implementazioni e per avere un certo grado di compatibilità con uno standard datato (RFC934). Deve essere generato in modo tale che non sia presente in nessuna delle parti del body.
Nota: il body di questo messaggio contiene più body parziali, ognuno dei quali ha alcuni header specifici. Es. il primo body parziale contiene i due header specifici Content-Type e ContentTransfer-Encoding, tutti gli altri header del messaggio globale non sono ridefiniti.
Parte finale del body del precedente messaggio multipart.
Il secondo body parziale ridefinisce i suoi header specifici Content-Type e Content-Transfer-Encoding, ed aggiunge un header Content-Disposition. Questo header segnala allo UA ricevente che il contenuto deve essere archiviato come allegato e ne specifica il nome.

MIME – Header Extensions

RFC2047
Scopo principale: consentire la presenza di caratteri internazionali negli header, mediante l'inserimento di campi codificati.
Campo codificato
"=?" charset "?" encoding "?" encoded-text "?="
charset: può assumere uno qualsiasi dei valori registrati da IANA
encoding: può assumere due valori
B: identico a base64
Q: identico a quoted-printable, ma con il carattere UNDERSCORE ("_") al posto del carattere SPAZIO (0x20), e la necessità di codificare il carattere "?".

04 SMTP

Simple Mail Transfer Protocol (SMTP)

Simple Mail Transfer Protocol (SMTP)

RFC282. Modalità di funzionamento di tipo push. Il mittente spedisce il messaggio di posta verso il destinatario. SMTP serve esclusivamente per spedire messaggi di posta. Comunemente affiancato a protocolli di tipo pull (POP3 o IMAP). Consentono il prelievo dei messaggi di posta nella parte finale del loro percorso. Processi intermedi che comunicano per scambiarsi messaggi di email (quindi agendo come una coppia client/server SMTP): Mail Transfer Agent (MTA). Un applicativo che svolge le funzioni di MTA è detto mailer. Si basa sull'utilizzo di connessioni TCP in cui il server ascolta sulla wellknown port 25.

E-mail: comunicazione asincrona

Trasferimento di tipo store-and-forward.
Scenario di riferimento: l'utente A vuole inviare un messaggio di posta elettronica all'utente B.
Il processo è schematizzabile in tre fasi successive:
1. L'utente A, tramite il proprio User Agent (UA, talvolta indicato come Message User Agent, MUA) compone il messaggio di posta elettronica e lo consegna al MTA locale, che lo mette in una coda di

trasmissione
2. L'MTA dell'utente A trasferisce il messaggio all'MTA dell'utente B, che lo riceve e lo deposita nella mailbox dell'utente B
3. L'utente B estrae il messaggio dalla propria mailbox.
Non è richiesto a mittente e destinatario di accedere contemporaneamente al sistema. Il sistema prende in consegna un messaggio, lo inserisce in una coda trasmissiva e lo inoltra in un secondo tempo. La consegna al sistema non garantisce che il messaggio venga trasferito alla mailbox del destinatario. Il trasferimento alla mailbox non garantisce che il destinatario legga il messaggio.

SMTP – Caratteristiche principali

Messaggi con formato NVT ASCII «stampabili», più l'eventuale carattere SP (spazio, terminati da <CRLF>). Vietato l'uso di CR o LF isolati.
Interazione di tipo query/response two-way alternate (half-duplex) tra client e server:
Comando: client to server: Corrisponde ad una keyword di 4 caratteri (in alcuni casi, seguita da SP ed altri caratteri), più eventuali parametri
Risposta: server to client: Corrisponde ad una stringa di 3 cifre decimali, seguita in genere da SP e da una stringa alfanumerica esplicativa.
Possibile una sessione unica per molteplici messaggi.

Esempio di sessione SMTP

Il client telnet sul sistema A (cl1) simula un MTA chiamante. Il server SMTP di riferimento sul sistema B (smtpsrv) è in ascolto sulla porta TCP 25.
220 smtpsrv.unipr.it ESMTP Postfix (Ubuntu): Banner del server SMTP, variabile in funzione del SW e dell'host su cui il SW del server SMTP è in esecuzione. La parte più significativa è il reply code (o response code) iniziale 220, avente significato di conferma positiva.
HELO: Comando necessario per il client per iniziare il dialogo presentandosi. Viene usato una volta sola all'inizio della sessione. Il parametro [cl1.unipr.it] è opzionale (come tutti i parametri indicati tra parentesi quadre).
MAIL: Comando utilizzato una volta sola per ogni messaggio. Indica il mittente del messaggio (o reverse path, ossia una mailbox a cui viene inviata una mail di risposta, se nella mail di risposta non si specifica il destinatario).

Codici di risposta (reply codes)

RFC3463. Convenzione comune a molti protocolli applicativi in Internet.
Prima cifra del reply code: 2 = corretto, 3 = corretto, ma occorrono altri dati, 4 = errore, occorre ripetere l'azione e 5 = errore irrecuperabile.
Le altre due cifre sono meno importanti e dipendono dal protocollo. Le risposte hanno un aspetto abbastanza discorsivo tale da fornire informazioni sullo stato dell'interazione. Nei sistemi moderni, i codici di risposta sono spesso integrati da un ulteriore codice di stato composto da tre campi numerici separati da un carattere.
Indirizzo e-mail
Stringa di caratteri che identifica una mailbox (mittente o un destinatario).
Struttura: local-part@domain
domain: è un FQDN e serve per identificare (tramite un record MX nel DNS) l'host su cui è in esecuzione il server SMTP a cui appartiene la mailbox. Case insensitive.
local-part: identifica la mailbox e deve essere interpretata solo dal server SMTP. Può essere case sensitive (dipende dal server, di solito è case insensitive).
I comandi SMTP sono case insensitive. La lunghezza massima di local-part è di 64 caratteri. La lunghezza massima di un indirizzo e-mail è pari a 256 caratteri (compresi i caratteri «.» e «@»).

Esempio di sessione SMTP

RCPT: Questo comando deve essere presente almeno una volta in un messaggio, e deve essere ripetuto

nel caso ci siano più destinatari. La risposta 250 2.1.5 Ok indica semplicemente che la sintassi del comando è corretta, non significa che l'MTA sia in grado di consegnare il messaggio. Il server SMTP non ha informazioni sull'effettiva esistenza della mailbox mario.rossi@example.com (potrebbe anche non esistere). Un server SMTP deve essere in grado di accettare almeno cento comandi RCPT.
DATA: Comando che indica l'inizio del corpo del messaggio. Il server si mette in modalità input (ossia in attesa di linee che non vengono interpretate come comandi) e tutto ciò che riceve lo considera come corpo del messaggio. Il server considera la fine dell'inserimento quando riceve una linea costituita dal solo carattere «.» (in prima colonna). Risposta a questo comando: 354 End data with <CR><LF>.<CR><LF>. Nota: il codice di risposta inizia con il carattere decimale 3.
250 2.0.0 Ok: queued as AEE251106AC: Codice informativo sull'effettiva presa in carico dell'e-mail da parte del server SMTP.
QUIT: Comando necessario per chiudere la sessione. Il server SMTP restituisce un codice di conferma ed un messaggio di chiusura. In alternativa al comando QUIT, nel corso della medesima sessione sarebbe possibile iniziare il trasferimento di un'altra e-mail inviando un nuovo comando MAIL.

SMTP

Funzionalità di Relay

Extended/Enhanced SMTP (ESMTP)

Funzionalità di Relay

Scenario di riferimento: l'MTA originante dell'utente A non è in grado di collegarsi direttamente all'MTA del destinatario B. Viene stabilita una sessione con un MTA intermedio (MTA C) che riceve l'e-mail e che, in un secondo tempo, la ritrasmette, operando quindi come un relay di livello applicativo. Il chiamato (sistema C) dovrà esaminare gli indirizzi dei destinatari e, per ogni mailbox non di sua pertinenza (non appartenente al suo domain space), identificherà un server SMTP al quale si collegherà fungendo questa volta da chiamante. Situazione abbastanza comune quando il destinatario finale (utente B) si trova all'interno di un'entità «chiusa» i cui mailer non sono visti al di fuori del proprio dominio, tranne uno (MTA C) che serve da gateway verso il mondo esterno.

Relay SMTP

Relay system di livello applicativo che utilizza il protocollo SMTP. Funzione tipicamente svolta dai Mail Exchanger (MX) al servizio di un dominio. Un MX è un Message Transfer Agent (MTA) con diversi compiti. Inoltra all'interno del proprio dominio di competenza messaggi provenienti dall'esterno del dominio. Inoltra all'esterno del proprio dominio messaggi provenienti dall'interno del proprio dominio.
Tipicamente: ogni altro tipo di relay (e-mail originate all'esterno del proprio dominio di competenza e destinati all'esterno) è disabilitato, per scoraggiare fenomeni di SPAM.
Server SMTP configurato in modo da permettere a chiunque di inviare posta attraverso di esso (e-mail originate all'esterno del proprio dominio di competenza e destinati all'esterno): open mail relay.
Soluzione fortemente scoraggiata, per prevenire fenomeni di SPAM.
Molti server SMTP bloccano la posta in arrivo da un open mail relay.

Liste di distribuzione (mailing list)

Per inviare la medesima e-mail a molteplici destinatari è possibile definire una mailing list. Una mailing list è una pseudo-mailbox (exploder). L'MTA ricevente (server SMTP) può espandere la lista in due modalità:
Modalità alias: la pseudo-mailbox è sostituita a turno da ogni suo indirizzo componente;
Modalità list: la pseudo-mailbox è sostituita dall'intera lista, e l'indirizzo mittente è sostituito dall'indirizzo dell'amministratore della lista di distribuzione.
Non è indispensabile definire una lista di distribuzione in un server SMTP: la lista può anche essere gestita localmente dal mittente. Ad esempio, configurandola nel proprio programma di posta (MUA), che provvederà ad inviare più messaggi ai singoli destinatari.

Vantaggi di una lista gestita localmente nel MUA: Semplicità di controllo per evitare loop, Una lista contiene una lista che contiene una lista che porta a una Quantità di messaggi fuori controllo; Semplicità di controllo per evitare duplicati e Numero di e-mail che saranno generate noto a priori.
Vantaggi di un exploder remoto (realizzato dal server SMTP): Possibilità di invio di messaggi ad una lista senza doverne conoscere a priori tutti i membri e Gestione della lista centralizzata presso l'amministratore, non delegata ai singoli mittenti.

Extended/Enhanced SMTP (ESMTP)

RFC2821. Identificato dall'invio del comando iniziale EHLO (al posto del comando iniziale HELO) dal client al server SMTP. *Il server SMTP deve rispondere dichiarando le proprie capacità:* 250-smtpsrv.unipr.it, 250-PIPELINING, 250-SIZE 41943040, 250-VRFY, 250-ETRN, 250-STARTTLS, 250-AUTH PLAIN LOGIN, 250-ENHANCEDSTATUSCODES, 250-8BITMIME, 250-DSN e 250 SMTPUTF8.
Struttura generale del protocollo SMTP inalterata. I comandi principali sono i medesimi, salvo accettare parametri aggiuntivi. Alcuni codici di risposta potrebbero cambiare rispetto a RFC821 (SMTP di base).
Server SMTP compliant con RFC821: all'arrivo del comando EHLO risponde con un codice di errore 500. A questo punto il client può tentare con l'invio del comando HELO.

Recap sui principali comandi SMTP

RFC2821, minimum implementation.
EHLO (oppure HELO): inizializza la sessione.
MAIL: inizializza una nuova transazione (un nuovo messaggio e-mail), richiedendo la specifica del return path.
RCPT: specifica la mailbox di destinazione (in caso di destinatari multipli, il comando deve essere ripetuto).
DATA: fornisce al server SMTP il contenuto del messaggio.
RSET (RESET): provoca l'aborto della transazione, mantenendo però attiva la connessione TCP.
VRFY (VERIFY): il server SMTP verifica che l'argomento ricevuto in input sia una mailbox.
EXPN (EXPAND): il server SMTP verifica che l'argomento sia una lista (nel caso in cui lo sia, fornisce l'elenco dei componenti).
HELP: il server SMTP fornisce un elenco dei comandi e altre informazioni ausiliarie.
NOOP: comando per richiedere un OK dal server SMTP.
QUIT: chiude la sessione tra client e server SMTP, svincolando la connessione TCP.

Trasporto a 8 bit in ESMTP

Attivazione del trasferimento a 8 bit: mediante il parametro BODY=8BITMIME nel comando MAIL. A questo punto, il server SMTP preserva tutti i bit di ogni ottetto. Il messaggio è comunque organizzato a linee (sequenze di byte comprese fra due coppie <CRLF>). Limiti su lunghezza massima e carattere terminatore identici allo standard originale. Nel caso in cui, durante la consegna del messaggio, debba essere effettuata un'operazione di relay inoltrando il messaggio ad un server SMTP che non accetta l'estensione 8BITMIME. Necessario codificare il messaggio in una codifica MIME quoted printable o Base64.

Ulteriori estensioni ESMTP

SIZE (Message Size Declaration, RFC1870): Dichiarato come supportato da un server SMTP, può essere dichiarato dal client come parametro SIZE=L del comando MAIL. L'identifica la dimensione (in byte) del messaggio, inclusi header, body e sequenze interne alle linee definite con il comando DATA, ma escludendo il carattere terminatore del comando DATA stesso. Il server SMTP può rifiutare il messaggio, in risposta ad un comando RCPT (il server non ha sufficiente spazio per il messaggio verso questa mailbox di destinazione).
ETRN (Extended Turn, RFC1985): Comando invocato dal client su un server SMTP per forzare il

server stesso ad elaborare i messaggi (eventualmente presenti nella coda di invio del server SMTP) appartenenti al client.
PIPELINING (RFC2920): Non richiede un'invocazione da parte del client nei confronti del server SMTP. Identifica la proprietà del server di elaborare i comandi che riceve nell'ordine in cui li riceve dal client, senza dover fornire un feedback al client dopo ogni comando (come dettato dal meccanismo two-way alternate). Maggiore efficienza.
DSN (Delivery Status Notifications, RFC3461): Definisce un insieme di parametri aggiuntivi per i comandi MAIL e RCPT. Parametri addizionali per il comando MAIL.
RET: in caso di notifica d'errore, specifica se restituire il messaggio intero (RET=FULL) o solamente gli headers (RET=HDRS).
ENVID (Envelope Identifier): lunghezza massima di 100 caratteri USASCII, definito dal client ed identico per tutti i messaggi inviati dal server SMTP in una certa sessione di invio (utile per identificare la sessione di invio). Parametri addizionali per comando RCPT.
NOTIFY: specifica le condizioni per cui un server SMTP debba generare delle DSN per lo specifico destinatario. Valori ammessi: NEVER, SUCCESS, FAILURE, DELAY. Accetta valori multipli (separati da «,»), tranne nel caso del valore NEVER, che deve apparire come valore unico.
ORCPT (Original Recipient): specifica un destinatario «originale» che debba corrispondere al destinatario indicato con il comando RCPT corrente. Sintassi: ORCPT=rfc822;Bob@Example.COM

Autenticazione SMTP

Consiste nel dichiarare le proprie generalità dimostrando che sono valide. Per alcuni protocolli è indispensabile (supporto ad autenticazione fornito da standard), per altri è prevista mediante successive integrazioni al protocollo. SMTP di base non la prevede. In ESMTP è solitamente usata per avere accesso alle funzionalità di relay. Storicamente i relay (quando agenti come client nei confronti di un successivo server SMTP destinatario dei messaggi) erano autenticati mediante indirizzo IP. Problema: indirizzo IP dinamico. Supporto a meccanismi di autenticazione. Dalla lista di estensioni supportate e dichiarate dal server SMTP (comando EHLO), il client deve inviare un comando AUTH X, per informare il server sulla modalità scelta di autenticazione.
Valori possibili per X: PLAIN (utilizza un encoding Base64); LOGIN (utilizza un encoding Base64); GSSAPI (Generic Security Services Application Program Interface); DIGEST-MD5; MD5; CRAM-MD5 (Challenge-Response Authentication Mechanism basato sull'algoritmo HMAC-MD5); OAUTH10A (basato su token OAuth 1.0a HMAC-SHA1, RFC5849); OAUTHBEARER (basato su token di tipo bearer OAuth 2.0, RFC6750).
Se l'utente non si autentica, la sessione può anche procedere (dipende dalla configurazione del server SMTP), ma al primo comando RCPT in cui venga specificato un destinatario non autorizzato (che richiede relay) si ottiene un rifiuto.

Accesso al server SMTP da fuori dominio

Tutti gli host interni alla Intranet possono inviare e ricevere e-mail da qualunque destinazione usando il mailer di dominio. Un MTA esterno può collegarsi al mailer di dominio e trasferire messaggi solamente se destinati all'interno del dominio. Un MTA esterno vuole collegarsi al mailer di dominio e trasferire posta destinata all'esterno del dominio (servizio di relay): possibile solamente previa autenticazione. Es.: utente della Intranet con un PC portatile che debba inviare e-mail in modalità di remote working. Se l'utente non è connesso alla rete aziendale (es. tramite VPN), apparirà sulla rete Internet con un indirizzo IP esterno allo spazio di indirizzi aziendale, quindi il mailer vedrà l'utente come un host esterno.

Autenticazione SMTP - AUTH PLAIN

Successivamente alla ricezione del codice di risposta 334 da parte del server SMTP, il client invia le credenziali codificate in formato Base64.

Autenticazione SMTP - AUTH LOGIN

Username e password disaccoppiati e richiesti dal server tramite dei messaggi codificati in formato Base64.

Autenticazione SMTP - AUTH CRAM-MD5

Il server SMTP risponde con un challenge codificato in Base64, che decodificato si mostra con un formato simile ad una mailbox

dec_base64(PDIxNzM0NTI4MTkuMzgzNzgyNjBAbWFpbHNydi51bmlwci5pdD4=) =.

2173452819.38378260 è una stringa random tipicamente generata a partire dal pid del processo del server SMTP ed il timestamp corrente.

Il client genera un digest tramite l'algoritmo HMAC-MD5, utilizzando come secret la password dell'utente:

digest = hmac-md5(challenge){password}

digest = hmac-md5(){password1234}

digest = a0c95bef72d70266f10e02d3cb17bd5e

Il client invia una stringa contenente username e digest, separati da uno spazio e codificati in Base64, al server STMP

enc_base64(username digest)

enc_base64(ldavoli a0c95bef72d70266f10e02d3cb17bd5e)

bGRhdm9saSBhMGM5NWJlZjcyZDcwMjY2ZjEwZTAyZDNjYjE3YmQ1ZQ==

Message Submission Agent (MSA)

RFC4409. Allo scopo di separare le funzioni di accesso al sistema di posta da quelle di inoltro della posta è stata identificata la funzionalità svolta da un Message Submission Agent (MSA). Rispetto ad un utente che invia un'e-mail, un MSA si comporta fondamentalmente come un server ESMTP. Impone l'autenticazione. Effettua dei controlli aggiuntivi (es. validità della data/ora dei messaggi, indirizzo del destinatario senza dominio, ecc.). Aggiunge eventuali header per forzare determinati criteri di validità/uniformità. Completate le funzioni sulle e-mail in ingresso, i messaggi sono inoltrati secondo le modalità consuete. Le funzioni di accesso sono svolte connettendosi alla porta dedicata 587. Allo scopo di limitare lo SPAM, molti MTA, oltre a bloccare l'accesso da parte degli open relay, bloccano anche indirizzi IP contenuti in determinate liste. Es: nel caso in cui lo UA avesse un indirizzo IP dinamico e cercasse di accedere direttamente all'MTA bypassando il MSA, verrebbe probabilmente bloccato.

05 E-mail Accesso al sistema

Un MTA in ogni stazione di lavoro: necessario?

User Agent (UA). È presente in ogni stazione di lavoro. Deve interagire con il MTA locale.

Message Transfer Agent (MTA). Deve comportarsi come un server, per ricevere la posta elettronica. Deve essere sempre attivo e connesso ad Internet. Deve avere visibilità esterna, disponendo di un FQDN registrato nell'ambiente DNS. Quindi un MTA: Non può avere un indirizzo IP di una classe privata (può averlo, ma non deve essere l'unico indirizzo IP su cui è raggiungibile). Costituisce un elemento critico, a rischio di attacchi esterni. Non conviene realizzare un MTA su ogni computer.

Post Office Protocol version 3 (POP3)

Versione attualmente più diffusa del protocollo mediante il quale un utente finale accede alla propria mailbox.

Maildrop: deposito finale dei messaggi destinati a una data comunità di utenti. Processo server che gestisce le mailbox dell'utenza.

Un MTA che realizzi le funzioni di maildrop deve essere conosciuto all'esterno tramite un record MX inserito nel DNS, e nella pratica corrisponde ad un server SMTP. Un MTA realizzato sul computer dell'utente non dispone di un record MX conosciuto dai DNS, pertanto non è raggiungibile dall'esterno (non è un server SMTP, ma un client SMTP). *POP3* consente di prelevare i messaggi di posta elettronica dal maildrop e trasferirli sull'host dell'utente. Il MTA del maildrop è anche un server POP3. Il MTA dell'utente è un client POP3. I più comuni client di posta elettronica (es. Outlook, Thunderbird,

ecc.) combinano differenti funzionalità. Funzionalità come UA (per preparare messaggi, visualizzarli, archiviarli, ecc.). Funzionalità come client SMTP (per inviare posta). Funzionalità come client POP3 (per prelevare posta dal maildrop). Protocollo alternativo a POP3: Internet Message Access Protocol (IMAP).

POP3 – Caratteristiche

RFC1939. Basato sul protocollo di trasporto TCP. Un server POP3 si pone in attesa di eventuali richiesta sulla porta 110. Dialogo tra client e server secondo il paradigma request/response (comandi/risposte), con comandi case-insensitive. Risposte: +OK, -ERR. Protocollo stateful, di tipo text-based. Basato sull'utilizzo di caratteri con codifica USASCII. Linee terminate dalla sequenza. Ogni e-mail nel mail drop è identificata da un identificativo intero progressivo.

RFC1939 successivamente aggiornata da altre RFC:

RFC1957, «Some Observations on Implementations of the Post Office Protocol (POP3)»
RFC2195, «IMAP/POP AUTHorize Extension for Simple Challenge/Response»
RFC2449, «POP3 Extension Mechanism»
RFC3206, «The SYS and AUTH POP Response Codes»
RFC5034, «The Post Office Protocol (POP3) Simple Authentication and Security Layer (SASL) Authentication Mechanism».

POP3 – Stati operativi

1 AUTHORIZATION: A seguito all'avvenuta connessione TCP il server POP3 mostra il messaggio di saluto e il client deve autenticarsi ed ottenere quindi l'autorizzazione per gestire la posta.
2. TRANSACTION: Il client emette comandi verso il server POP3 e riceve risposte dal server POP3. È possibile agire sul maildrop, Chiedere informazioni relative al maildrop, Scaricare messaggi di posta elettronica e Cancellare messaggi di posta elettronica.
3. UPDATE: Si entra in questo stato mediante l'invio del comando QUIT. Le modifiche apportate al maildrop vengono rese effettive. Es. comandi di cancellazione precedentemente memorizzati. La connessione TCP termina.

POP3 – Comandi fondamentali

STAT: Consente di ottenere informazioni sullo stato del maildrop: restituisce il numero di messaggi presenti e lo spazio da essi occupato;
LIST [<numero messaggio>]: Senza parametri indica la dimensione di ogni messaggio, altrimenti solo quella del messaggio indicato dal parametro;
RETR <numero messaggio>: Visualizza il messaggio indicato, trasferendolo dal maildrop al client POP3;
TOP: Visualizza un numero predefinito di linee dalla testa del messaggio specificato
DELE: Cancella dal server il messaggio indicato;
RSET: Cancella le operazioni di cancellazione DELE in precedenza inviate al server, facendo abortire la transazione (la connessione TCP rimane attiva);
NOOP: Non esegue alcuna operazione, restituisce solamente un messaggio +OK nel caso in cui il server POP3 risponda;
QUIT: Richiede l'esecuzione delle operazioni pendenti, termina la sessione POP3 corrente e si disconnette dal server.

Questi comandi possono essere emessi solamente in stato TRANSACTION. Dopo che l'utente si sia autenticato ed abbia iniziato la sessione POP3. Parametri inclusi fra parentesi quadre: opzionali, gli altri obbligatori.

POP3 – Autenticazione utente

L'accesso alla mailbox deve essere riservato e protetto tramite misure di autenticazione. Per accedere al server POP3, il client deve autenticarsi. L'autenticazione deve avvenire ad inizio sessione, immediatamente dopo la connessione TCP. POP3 prevede due meccanismi di autenticazione.

USER/PASS. Authenticated POP (APOP). Il server POP3 deve supportare almeno uno dei due metodi (meglio se entrambi).

POP3 – Autenticazione USER/PASS

Il client emette il comando USER. Il server conferma la validità della mailbox. Il client emette il comando PASS. La password è trasferita in chiaro, senza alcuna forma di protezione.

POP3 – Autenticazione APOP

Un server POP3 che accetta questo metodo inserisce un timestamp nella propria bandiera di benvenuto. Corrisponde ad un identificatore unico (campo Message-ID dell'header secondo RFC2822). Il client emette il comando APOP.

Parametro <digest>: Il timestamp (comprensivo delle parentesi angolari) è concatenato con una stringa fissa, denominata secret, e condivisa e nota solamente al server ed al client. Crittografia simmetrica. Della stringa risultante ne viene calcolato l'hash mediante l'algoritmo MD5. digest = md5(<timestamp><secret>).

Quindi: ad ogni sessione POP3 viene utilizzata una stringa di autenticazione unica. Il digest, costituito da 128 bit, è trasferito in formato HEX tramite una stringa di 32 caratteri ASCII, in cui ogni byte è rappresentato da due caratteri e usando lettere minuscole.

POP3 – Comandi opzionali

TOP <numero messaggio> <N>: Visualizza un numero di linee dalla head del messaggio indicato, trasferendole dal server POP3. Le <N> linee comprendono header e, se <N> è sufficientemente grande, eventuale testo;

UIDL [<numero messaggio>]: Senza parametri restituisce gli identificatori univoci di tutti i messaggi nel maildrop, altrimenti restituisce l'identificatore univoco del messaggio indicato dal parametro <numero messaggio>.

Siccome in genere l'identificativo numerico di uno stesso messaggio cambia tra una sessione e la successiva, l'identificatore univoco (stringa) assegnato dal server POP3 è utile per avere un riferimento univoco. Quando il client POP3 si connette, richiede mediante il comando UIDL gli identificatori di tutti i messaggi presenti nella maildrop, e dalla lista ottenuta determina quali abbia già scaricato e quali no. Il client deve mantenere un elenco degli identificatori dei messaggi scaricati dal server e non cancellati.

POP3 – Esempio di sessione

Sono state omesse molte linee di header field (sostituite con ...)

Header field Status: RO

Nota: l'header field Status non fa parte della specifica ma viene usato comunemente da molti maildrop che lo inseriscono nel messaggio.

RO: Read Old, il messaggio è vecchio ed è già stato letto.

Estensioni POP3 (RFC 2449)

CAPA: Comando per richiedere al server POP3 la presenza di funzionalità ulteriori rispetto a quelle di base. Può essere emesso sia in stato AUTHORIZATION sia in stato TRANSACTION

SASL: RFC4422 Schema generale, destinato a essere utilizzato con protocolli applicativi di tipo stateful, mediante il quale si specificano uno o più meccanismi di autenticazione, aggiuntivi rispetto a quelli di base. APOP quindi non viene elencato.

STLS: Identifica l'abilità del server POP3 di supportare il trasferimento dei messaggi mediante il sub-strato Transport Layer Security (TLS) o Secure Socket Layer (SSL). Opzione basata sull'utilizzo di tecniche di crittografia ed indipendente dal meccanismo di autenticazione del server POP3. *Nota:* alcuni server usano il modo (deprecato) di realizzare il trasporto SSL non tramite il comando STLS, ma mediante la porta alternativa 995.

Internet Message Access Protocol version 4 (IMAP4)

IMAP4 – Caratteristiche

RFC2060. Definito come evoluzione del protocollo POP3. Protocollo di tipo stateful, usa il trasporto TCP ed un server IMAP4 rimane in attesa di eventuali richieste sulla porta 143. Consente di elaborare tutta la posta su un server remoto e centralizzato, in modo di avere a disposizione l'intero insieme dei propri messaggi da qualsiasi macchina ci si connetta. POP3: scarica in locale i messaggi di posta elettronica degli utenti. Dialogo basato sul paradigma request/response. Ogni comando deve avere un identificatore, ovvero un prefisso alfanumerico (es. A0001) chiamato tag, che lo precede, ed il cui compito è quello di far corrispondere le risposte del server alle rispettive richieste del client. Riconosce sintassi MIME ed utilizza caratteri USASCII, con linee terminate dalla sequenza. Possibilità di avere più comandi in corso. Possibilità di creare, cancellare e rinominare mailbox. Supporto della modalità di lavoro offline per i client e successiva sincronizzazione quando possibile.

IMAP4 – Comandi any state

Comandi che possono essere emessi dal client in qualsiasi stato della sessione.

CAPABILITY: Richiede al server IMAP quali siano le funzioni supportate;

NOOP: Comando che non esegue alcuna operazione, risponde solamente in modo positivo se il server è attivo. Utilizzato per evitare la disconnessione dal server causa timeout;

LOGOUT: Indica al server IMAP che il client vuole terminare la sessione, svincolando pertanto la connessione TCP.

IMAP4 – Comandi non-authenticated

Comandi che possono essere emessi dal client connesso al server IMAP ma che si debbano ancora autenticare per avere la possibilità di gestire la propria maildrop.

AUTHENTICATE: Comando utilizzato dal client per specificare al server IMAP che tipo di autenticazione sarà adottata per il login. Il client specifica uno o più meccanismi di autenticazione, in ordine di preferenza decrescente. I valori relativi ai meccanismi di autenticazione devono essere scelti tra quelli contenuti nel banner di benvenuto restituito dal server IMAP;

LOGIN <username> <password>: Comando utilizzato dal client per autenticarsi sul server IMAP.

IMAP4 – Comandi authenticated

Comandi che possono essere emessi dal client connesso al server IMAP solamente nel caso in cui la fase di autenticazione abbia avuto successo.

SELECT <mailbox>: Comando utilizzato per selezionare la mailbox con cui lavorare;

EXAMINE <mailbox>: Comando utilizzato per seleziona la mailbox su cui lavorare, ma con accesso in sola lettura;

CREATE <mailbox>: Comando utilizzato per creare una nuova mailbox;

DELETE <mailbox>: Comando utilizzato per cancellare definitivamente una mailbox;

RENAME <old-name-mailbox><new-name-mailbox>: Comando utilizzato per rinominare una mailbox;

STATUS <mailbox>: Comando utilizzato per richiedere lo stato corrente della mailbox;

APPEND <mailbox> <messaggio>: Comando utilizzato per aggiungere un messaggio alla mailbox selezionata;

CLOSE: Comando utilizzato per chiudere la mailbox corrente (selezionata con il comando SELECT), eliminare i messaggi marcati con il flag \Delete, e riportare il flusso in stato authenticated;

EXPUNGE: Comando utilizzato per rimuovere i messaggi marcati con il flag \Delete dalla mailbox corrente

SEARCH <search criteria>: Comando utilizzato per eseguire una ricerca tra i messaggi che soddisfano i parametri di ricerca indicati;

COPY <message set> <mailbox>: Comando utilizzato per copiare i messaggi nella mailbox;

UID <command> <arguments>: Comando utilizzato per esegue i comandi impartiti considerando l'identificativo UID al posto dell'indice intero del messaggio;

SUBSCRIBE <mailbox>: Comando utilizzato per aggiungere la mailbox indicata alla lista delle caselle

attive da visualizzare in un client IMAP;
UNSUBSCRIBE <mailbox>: Comando utilizzato per eliminare la mailbox indicata alla lista delle caselle attive da visualizzare in un client IMAP.

IMAP4 – Flag

IMAP implementa una sistema di attributi in grado di indicare lo stato corrente di un messaggio. Necessario vista la possibilità di accedere alla mailbox da più client IMAP in contemporanea, e la gestione centralizzata sul server IMAP.
Ogni messaggio può essere associato a differenti flag che ne indicano lo stato:
\Seen: indica che il messaggio è stato letto
\Answered: indica che è stata inviato un messaggio di reply al messaggio in oggetto;
\Flagged: indica che sul messaggio è stato impostato un flag Urgent o Attention;
\Deleted: indica che il messaggio è stato marcato per la cancellazione;
\Draft: indica che il messaggio non è completo ed è pertanto marcato come bozza;
\Recent: indica che il messaggio è appena stato depositato nella maildrop.

06 FTP

File Transfer Protocol (FTP) Gestione di file su host remoto

FTP – Funzionalità

RFC959. Consente il trasferimento di file fra sistemi e la gestione di file su host remoti.
Host remoto: corrisponde all'host su cui è in esecuzione il server FTP.
Host locale: corrisponde all'host su cui è in esecuzione il client FTP.
Trasferimento di file: avviene comunemente fra host locale e host remoto.
Download: trasferimento host remoto à host locale.
Upload: trasferimento host locale à host remoto.
È possibile anche il trasferimento di file tra due host remoti.

FTP – Tipologie di Connessioni

FTP utilizza due tipi di connessioni TCP.
1. Control Connection: Comunicazione usata per lo scambio di comandi e risposte. Collega i processi USER-PI e SERVER-PI (PI=Protocol Interpreter). Usa una connessione sulla well-known port TCP 21 del server FTP. Segue la sintassi del protocollo TELNET.
2. Data Connection: Comunicazione fra due processi, chiamati Data Transfer Process (DTP), utilizzata per il trasferimento dei dati. Il processo in ascolto (chiamato) è detto «passivo» ed attende su una porta spesso allocata dinamicamente. Durante un'unica control connection si possono avere più data connection.

FTP – Schema funzionale

User-FTP: applicativo che è in esecuzione sull'host locale. Interagisce con l'utente ed è il client FTP da cui inizia la transazione.
Server-FTP: applicativo che è in esecuzione sull'host remoto. Interagisce solo con l'applicativo User-FTP e con il file system dell'host su cui è in esecuzione.
Piattaforme Unix: in genere sono fornite di un applicativo ftpd (ftp daemon) che svolge queste funzioni e che, se configurato all'interno del file di boot opportuno, all'avvio dell'host si mette in attesa sulla porta 21. Sulla data connection il processo passivo (chiamato) è quello denominato User-DTP e la porta passiva non è precisata. Viene specificata al Server-FTP col comando PORT. Nel caso in cui tale comando non venga emesso, il Server-FTP tenta di connettersi alla porta di default, ossia la porta da cui l'User-DTP ha iniziato la data connection.
La porta 20 è riservata sull'host remoto per effettuare le data connection verso i vari clienti. *Modalità di funzionamento:* Active FTP.

FTP – Comandi generici

USER <username>: Comando utilizzato per l'inserimento dello username dell'utente che si deve

autenticare. Spesso l'accesso ai server FTP può avvenire in forma anonima, mediante l'username anonymous («Anonymous FTP Server»).
PASS <password>: Comando utilizzato per l'inserimento della password dell'utente che si deve autenticare (trasferita in chiaro). Solitamente, allo username anonymous è associata una password corrispondente all'indirizzo e-mail dell'utente che si vuole connettere.
REIN: Re-inizializza la connessione FTP allo stato in cui era quando il client si è connesso la prima volta al server FTP. Ai trasferimenti dati in corso è concesso di completare. La control connection rimane attiva ed il sistema si aspetta un comando USER come comando successivo.
HELP [<command>]: Restituisce informazioni su uno specifico comando (se passato come parametro), altrimenti un aiuto generale.
NOOP: Comando che non esegue alcuna operazione specifica, spesso utilizzato per mantenere attiva la connessione.
QUIT: Comando che richiede la chiusura della sessione FTP, a cui segue un'azione di svincolo della connessione TCP da parte del server FTP.
Punto debole del protocollo FTP già presente in POP3 e in altri protocolli: le password sono trasferite in chiaro. In genere il problema è superato non aggiungendo meccanismi di autenticazione più raffinati. Protezione del meccanismo originale mediante uno strato che garantisce la sicurezza basato su algoritmi crittografici (SSL o TLS).

FTP – Codici di risposta

Messaggio di risposta contenente un codice a tre cifre.
Esito (simile al protocollo SMTP) – prima cifra:
1xx: Positivo (preliminare)
2xx: Positivo (azione completata)
3xx: Positivo (risposta intermedia, azione non completata)
4xx: Negativo (inconveniente temporaneo)
5xx: Negativo (non riprovare)
Tipologia – seconda cifra:
x0x: errore di sintassi
x1x: risposta a una richiesta di informazione
x2x: risposta relativa a una connessione
x3x: risposta relativa ad account/autorizzazioni
x4x: non usato
x5x: risposta relativa al file system sul server.

Tipi di dato

ASCII: Il mittente converte in NVT-ASCII (8bit) per l'invio: un carattere ASCII (7 bit) è incapsulato in un ottetto (8 bit). Il ricevente converte nel suo formato;
EBCDIC: Alfabeto a 8-bit utilizzato nei mainframe;
IMAGE (flusso di byte): Tipo comunemente utilizzato per file non testuali. Solitamente indicato come binary;
LOCAL: Il parametro definisce la lunghezza in bit della parola. Es. L 32 identifica parole di 32 bit.

FTP - Comandi di tipo "data transfer"

Comandi per la specifica della rappresentazione dei dati:
STRU (struttura del file):
STRU F (File – sequenza di ottetti, default);
STRU R (Record – sequenza di record);
STRU P (Page – pagine indicizzate, accesso random).
TYPE (tipo di dati):
TYPE A (ASCII, default);

TYPE E (EBCDIC);
TYPE I (Image);
TYPE L (Local).
MODE (modalità di trasferimento dei bit):
MODE S (Stream);
MODE B (Block);
MODE C (Compressed);
MODE S: Il file è trasmesso come un flusso di ottetti, se il file non è strutturato (STRU F) la sua fine è segnalata dalla chiusura della data connection da parte del mittente;
MODE B: Il file è trasmesso come una serie di blocchi, ognuno preceduto da un header di 24 bit, descrittore di 8 bit (EOF, ecc.); byte count di 16 bit (indica la lunghezza del blocco). La fine del file è segnalata dal descrittore e la data connection può essere mantenuta per un trasferimento successivo;
MODE C: Serve per comprimere un byte ripetuto N volte trasmettendolo una volta sola, preceduto da un byte di conteggio.

FTP - Comandi di controllo delle directory

PWD: Restituisce l'indicazione della directory in cui ci si trova attualmente;
CWD <new directory>: Comando che consente di spostarsi dalla directory corrente ad una nuova directory;
CDUP: Comando che consente di risalire l'albero delle directory fino al root;
RMD <directory>: Comando che consente di rimuovere la directory specificata;
MKD <new directory>: Comando per la creazione di una nuova directory.
La sintassi non è identica esattamente identica ai comandi da CLI degli OS Windows e Unix.

Connessioni FTP

Nota: control connection e data connection sono connessioni TCP. La control connection è sempre iniziata dall'host dell'utente (processo User-PI). User-PI chiama Server-PI sulla porta 21. La data connection è iniziata (di default) dall'host server (processo Server-DTP):
1. User-DTP si mette in attesa su una porta ("passive mode")
2. User-PI informa il processo Server-PI su quale sia la porta su cui sta attendendo (comando PORT)
3. Server-DTP chiama User-DTP dalla porta 20 alla porta dichiarata. In una data connection di questo tipo, il processo User-DTP agisce come un server, mentre il processo Server-DTP come un client.
Nel caso in cui non venga utilizzato il comando PORT, User-DTP viene chiamato sulla porta di default (la porta da cui User-PI ha iniziato la control connection).

Accesso ad un host esterno

Problema: se la Intranet utilizza indirizzi IP privati (come è consuetudine), la funzionalità di Network Address Translation (NAT) attiva sul router di frontiera altera i valori <IP address, port> di origine (e specificati dai comandi PORT/PASV), impedendo quindi il funzionamento del protocollo FTP.
Soluzione: il router di frontiera deve essere a conoscenza dell'esistenza dei servizi FTP attivi nella Intranet, in modo da poter manipolare alcuni comandi (RFC2766).
Nota: i comandi PORT e PASV costituiscano un'anomalia dal punto di vista architetturale, siccome presuppongono che l'applicativo sia a conoscenza dei dettagli sui protocolli sottostanti (Transport per decidere la porta, Network per conoscere l'indirizzo IP).
Normalmente il firewall blocca, per ragioni di sicurezza, le connessioni TCP in ingresso. Connessioni in cui il chiamante è esterno all'Intranet ed il chiamato interno. Eventuale eccezione su particolari host e porte note e prefissate a priori.
Quindi: la data connection non può essere stabilita.
Soluzione: modalità "passiva" (iniziata dal comando PASV): la data connection è chiamata dall'interno.

FTP – Comandi di controllo delle connessioni

PORT h1,h2,h3,h4,p1,p2: Comando mediante il quale il client dichiara al server su quale indirizzo IP e porta un DTP (in attesa di una data connection).
h1,h2,h3,h4: valori (in formato decimale) corrispondenti alla dotted decimal notation di un indirizzo IPv4.
p1,p2: byte più significativo e meno significativo (in formato HEX) component il numero di porta (porta = p1 * 162 + p2).
PASV: Comando mediante il quale il client richiede al server FTP di mettersi in modalità passiva (in attesa di una data connection).
Risposta dal server: 227 Entering Passive Mode (h1,h2,h3,h4,p1,p2). Indirizzo IP e porta su cui chiamare un DTP per effettuare la data connection.

Trasferimento file tra due server FTP remoti

User FTP ha in corso due control connection contemporanee, ognuna con un server FTP differente. Interazione simultanea con più server remoti per comandare una data connection trasversale ed un file transfer diretto fra i server.

Controllo connessione tra due server FTP

1. Il client FTP richiede al server A di mettersi in modalità passiva
2. Il server A risponde con i dettagli sul proprio indirizzo IP e porta su cui è in attesa
3. Il client FTP inoltra queste informazioni al server B
4. A questo punto, viene forzato un comando di trasferimento file sul server B, in modo da avviare il trasferimento dal server B al server A
Nota: per ragioni di sicurezza, molti server rifiutano le data connection con host diversi da quelli con cui hanno la control connection attiva, rendendo di fatto impossibile la triangolazione. Il server B potrebbe rifiutarsi di effettuare la data connection verso il Server A, siccome il server A ha un indirizzo IP differente rispetto a quello del client FTP (host utente).

FTP – Comandi di gestione dei file

RETR: Innesca il trasferimento di una copia del file dal Server-DTP all'altra estremità della data connection (solitamente allo User-DTP);
STOR: Richiede al Server-DTP di accettare il trasferimento dati attraverso la data connection, e di salvare i dati come file sul server. Se un file con il medesimo nome esiste già, il file esistente viene sovrascritto;
RNFR: Rename From. Specifica il vecchio path del file che deve essere rinominato;
RNTO: Rename To. Specifica il nuovo path del file che deve essere rinominato;
DELE: Comando che causa la rimozione del file specificato.
I comandi RNFR e RNTO devono essere eseguiti in sequenza, per rinominare effettivamente un file.

FTP – Altri comandi di servizio

LIST: Richiede al server di restituire al DTP passivo una lista di file, nel caso in cui sia invocato su un path che identifica una directory, oppure informazioni su uno specifico file.
NLST: Name List. Simile al comando LIST, con l'eccezione che il server restituisce solamente una lista di file, non riportando informazioni su dimensioni file, autorizzazioni, ecc.
STAT: Richiede al server di rispondere con un'indicazione sullo stato delle operazioni, tramite la command connection (es. relativa ad un trasferimento in corso).

FTP – Minimum implementation

Configurazione "minima" che qualsiasi server FTP deve accettare:
TYPE – ASCII Non-print
MODE – Stream
STRUCTURE – File, Record
Commands: USER, QUIT, PORT, TYPE, MODE, STRU, RETR, STOR, NOOP
Parametri di trasferimento:

TYPE – ASCII Non-print
MODE – Stream
STRUCTURE – File.

FTP – Estensioni

RFC3659. *REST (Restart of Interrupted Transfer). MDTM (File Modification Time).* Può essere utilizzato per determinare quando un file sia stato modificato l'ultima volta
SIZE (File Size): Utilizzato per ottenere la dimensione di un file (in byte) in trasferimento dal Server-DTP. Tale valore cambia in funzione dei parametri STRU, MODE, TYPE validi al momento sulla data connection.
MLST (Machine List For File). Estensione del comando LIST
MLSD (Machine List For Directory). Estensione del comando LIST
FEAT: Restituisce la lista delle feature implementate dal server FTP.

FTP – Estensioni – Internazionalizzazione

RFC2640
Path Names: Occorrono estensioni nel caso di OS che ammettono spazi o stringhe non-ASCII nei nomi di percorso. Pathname codificato in UTF-8, diventando una sequenza di byte che possono assumere valori nell'intervallo 0x01-0xFF. Nel pathname possono essere presenti caratteri quali <SP>, <CR>, <LF>. Nel caso in cui nel pathname appaia un <CR>, esso va codificato con la sequenza di due byte 0x00.
Selezione lingua: Comando LANG. Es.: LANG fr (il client FTP richiede messaggi in francese). La selezione lingua viene eseguita secondo le regole di negoziazione FEAT (RFC2389).

Trivial File Transfer Protocol (TFTP)

RFC1350. FTP semplificato. Alternativo al protocollo FTP. Privo di qualsiasi meccanismo di sicurezza (no login e meccanismi di controllo di accesso). Nessun meccanismo efficiente di recupero errori. Uso limitato pressoché esclusivamente in ambiente locale. TFTP utilizza UDP come protocollo di trasporto (stack UDP/IP). Meccanismi di autenticazione, controllo di accesso, confidenzialità, controllo di integrità. Potrebbero essere implementati ai livelli superiori o inferiori rispetto a quello in cui è attivo TFTP. Attenzione ai permessi concessi al processo del server TFTP. Sicurezza del file system. Solamente file pubblici devono essere condivisi in accesso read-only al server TFTP. Tipicamente: funzionalità di listing, deleting, renaming, and scrittura dei file sono disabilitati.

TFTP - Download di un file

La well-known port UDP 69 serve al client per contattare il server. Il server risponde da una porta dinamica a cui dovranno essere inviati tutti i successivi messaggi. Non è prevista alcuna fase di autenticazione, chiunque può accedere ad un server TFTP. Non è possibile rinominare file, cancellarli o gestire directory. Le uniche operazioni possibili sono trasferimenti di file (download e upload). Un file è trasferito in blocchi numerati, di lunghezza fissa di 512 byte. Un blocco di lunghezza < 512 byte segnala la fine del file. Se un file ha lunghezza multipla di 512 occorre trasferire un blocco finale di lunghezza nulla (pari a 0). Ad ogni messaggio DATA deve corrispondere un riscontro (messaggio ACK). Qualsiasi anomalia (perdita di ACK, ecc.) provoca l'annullamento di tutta l'operazione.

TFTP – Upload di un file

Transazione del tutto simile a quella mostrata per il download di un file. I messaggi DATA fluiscono dal client verso il server, mentre i messaggi ACK fluiscono in verso opposto.

TFTP – Funzionalità e caratteristiche

Scopo: Consentire attività di download ed upload di file. Nessuna altra funzionalità (rimozione, cambio nome, directory, ecc.).
Caratteristiche principali: Basato su UDP (datagrammi), Nessuna autenticazione (login), Qualsiasi situazione di errore à transazione interrotta immediatamente, Estrema semplicità: utilizzato in dispositivi dotati di ridotte potenzialità (small memory footprint).

07 NNTP

Network News Transfer Protocol (NNTP) Messaggistica fra gruppi di interesse

Protocollo NNTP

Origine: Unix USErs' NETwork (USENET). Bacheca per messaggi pubblici. I messaggi sono raggruppati in NEWSGROUP. Un NEWSGROUP corrisponde a un argomento. I NEWSGROUP sono organizzati in una gerarchia.

Scopo: Distribuzione, ricerca, impostazione di messaggi (articoli). Solo articoli nuovi vengono trasmessi. Un host NNTP è un deposito organizzato di articoli. NNTP migliora le prestazioni di altri metodi di distribuzione (es. mail).

NNTP – Caratteristiche principali

RFC977. Messaggi in codifica NVT ASCII. Dialogo di tipo request/response (basato su comandi/risposte). Una risposta è sempre preceduta da un codice di 3 cifre decimali (come SMTP, POP, FTP, ecc.). Porta riservata: TCP 119.

NNTP – Codici di risposta

1xx – Informazioni generali
2xx – Comando ricevuto, esecuzione avvenuta con successo
3xx – Comando ricevuto, ma sono necessarie altre informazioni
4xx – Comando ricevuto, esecuzione non avvenuta
5xx – Comando non valido
x0x – Informazioni sulla connessione, sulle impostazioni o simili
x1x – Informazione sul newsgroup selezionato
x2x – Informazione sull'articolo selezionato
x3x – Informazione sul mittente del messaggio
x4x – Informazione sul destinatario del messaggio
x5x – Risposta di autenticazione
x8x – Messaggio per comandi avanzati non standardizzati
x9x – Messaggio di debugging.

NNTP – Comandi di richiesta di un articolo

ARTICLE <message-id>: Selezione tramite identificativo () contenuto nell'header dell'articolo. Il server NNTP restituisce l'articolo specifico;
ARTICLE <n>: Selezione tramite numero dell'articolo () nel newsgroup corrente. Nel caso in cui il parametro venga omesso: si assume l'articolo corrente. Il server NNTP restituisce l'articolo specifico.
Comandi simili ad ARTICLE:
BODY: Richiede solamente il corpo del messaggio;
HEAD: Richiede solamente l'intestazione del messaggio;
STAT: Aggiorna il puntatore all'articolo corrente (non invia testo).

NNTP – Comandi di controllo gruppi

LIST: Richiede un elenco dei newsgroup disponibili (inclusi il primo e l'ultimo numero articolo);
NEWNEWS <date> <hour>: Richiede tutti i nuovi articoli disponibili creati a partire da un certo istante;
NEWGROUPS <date> <hour>: Richiede tutti i nuovi gruppi disponibili a partire da un certo istante;
GROUP <group-name>: Specifica il gruppo per il quale viene fatta una richiesta.

NNTP – Comandi di invio messaggi

POST: Il client segnala al server NNTP che ha un articolo da inviare. Se la risposta è positiva il client manda l'articolo. L'articolo deve essere formattato secondo RFC850 (formato messaggi USENET). L'articolo deve essere terminato con una linea contenente un carattere «.».
IHAVE: Informa il server che il client possiede il messaggio contrassegnato da <massage-id>. Se il server NNTP non dovesse avere bisogno di quel messaggio (es. ne ha già una copia), restituisce un

codice 435, altrimenti un codice 335 differisce dal comando POST poiché serve per inviare messaggi già inviati tra elaboratori.

NNTP – Controllo delle connessioni

Non esiste un comando di inizio. La sessione inizia non appena il server NNTP riceva la chiamata e risponda con il proprio banner di benvenuto.

SLAVE: Dichiara al server NNTP che il client che ha emesso il comando è uno «slave server» e non uno UA di un utente finale.

QUIT: Emesso dal client, richiede al server NNTP la chiusura della sessione TCP, restituendo anche un ACK.

08 WWW

World Wide Web (WWW) Concetti generali

World Wide Web (WWW)

Complesso di documenti ipertestuali risiedenti su vari host e collegati fra loro mediante hyperlink. Gli utenti navigano attraverso questi documenti utilizzando un browser Web. Oltre al testo, i documenti possono contenere immagini, video e altre risorse di tipo multimediale. Il World Wide Web fu iniziato al CERN di Ginevra nel 1989 da Tim Berners-Lee. Attualmente è il direttore generale del World Wide Web Consortium (W3C).

Testi e ipertesti

Ipertesto: modello di interfaccia utente che permette al fruitore di un documento di «navigare dietro richiesta» mediante un PC (generalmente).

Tail anchor: origine dell'hyperlink all'interno di un documento.

Destination anchor: destinazione dell'hyperlink (tipicamente un documento).

Di norma: la tail anchor viene associata ad una zona del documento origine che ha attinenza logica con la destinazione.

Un testo si legge (sequenzialmente), un ipertesto si naviga. Il documento di origine e quello di destinazione tipicamente risiedono su host diversi. Occorre che tra i due host sia attivo un protocollo di comunicazione capace di trasferire il documento di destinazione sull'host A. Trasferimento dietro richiesta dell'utente.

Host A: client.

Host B: server.

Nota: il documento di destinazione non è necessariamente un ipertesto. Potrebbe non avere link o potrebbe non essere nemmeno un testo (es. un'immagine).

Standard per gli ipertesti

HyperText Markup Language (HTML): Standardizzato dal W3C, è il più diffuso standard per la definizione del formato degli ipertesti. Basato su «plain text" con inserimento di tag. Linguaggio di markup: unisce il testo «utile» e l'informazione per la gestione dello stesso inframezzandoli in un unico stream. Altri esempi di linguaggi di markup: RTF, LaTeX. Permette di definire gli hyperlink.

HyperText Transfer Protocol (HTTP): Definito dall'IETF in una serie di RFC. Principale protocollo applicativo del Web. Usato dal client per instaurare una connessione con il server ed accedere ad un documento residente su un «Information Server».

World Wide Web (WWW): insieme degli Information Server.

Web: tipi di Information Server

*Più tipi di servizi nel Web:*www (protocollo HTTP, HTTPS), ftp (protocollo FTP), nntp (protocollo NNTP), coap (protocollo CoAP), mailto (servizio di posta elettronica) e altri.

Ogni servizio servizio richiede il proprio protocollo. Il Web è un ambiente multiprotocollo. Un browser (Internet Explorer, Microsoft Edge, Mozilla Firefoz, Google Chrome, Apple Safari, ecc.) è un applicativo che in genere contiene molteplici client (client HTTP, client FTP, ecc.).

Indirizzamento nel Web URI, URL e URN

Indirizzamento: URI, URL e URN
Un servizio fornisce l'accesso ad una risorsa.
Uniform Resource Identifier (URI):
Uniform Resource Locator (URL): identifica dove si trovi la risorsa;
Uniform Resource Name (URN): identifica la risorsa in base ad un identificatore univoco e persistente (anche in caso di indisponibilità futura). Il termine URI è meno restrittivo, usato nelle specifiche (RFC). Nella pratica: l'URL è il tipo di URI utilizzato più comunemente.
L'URL è la stringa che il browser utilizza per localizzare una risorsa nel Web.
URI – Sintassi
Campi scheme e path obbligatori. La presenza del campo authority è segnalata dalla sequenza "//" iniziale.
scheme:hierarchical-part:
scheme: identifica il metodo d'accesso (protocollo). Stringa alfabetica in formato minuscolo. Informa il client su come accedere alla risorsa specificata dalla parte hierarchical-part.
hierarchical-part: Struttura generale sostanzialmente identica per tutti gli schemi, con qualche particolarità specifica.
URI – Codifica
Solo caratteri ASCII grafici (no control chars).
Alcuni caratteri (delimiter) sono riservati: /, ., #, ?, ;, :, $, … (altri) …
SPACE e caratteri riservati: Codifica con sequenza di escape «%hex-code»
RFC3986 classifica i caratteri secondo una suddivisione ben definita:
unreserved = ALPHA / DIGIT / "-" / "." / "_" / "~"
gen-delims = ":" / "/" / "?" / "#" / "[" / "]" / "@"
sub-delims = "!" / "$" / "&" / "'" / "(" / ")» / "*" / "+" / "," / ";" / "="
Caratteri unreserved: servono per comporre i nomi dei campi
Caratteri delimiter: servono per delimitarli.
URL con schema file
host è un domain name (o un indirizzo). La risorsa è accessibile tramite file system. L'host localhost è sottointeso. In questo caso il campo scheme non indica un protocollo. Il campo path segue le convenzioni del file system corrente (Windows, Unix, ecc.), mentre la sintassi è quella tipica di Unix. I nomi delle directory sono separati dal carattere «/» (slash) e non dal carattere «\» (backslash), come in Windows.
URL con schema http
host è un domain name oppure un indirizzo IP (IPv4 o IPv6)
port identifica la porta su cui è in ascolto il server http. Porta di default del protocollo HTTP: TCP 80.
path corrisponde ad un http selector.
http selector: path che identifica una risorsa (di solito un file) nell'albero delle directory.
La radice di tale albero non è la radice del file system sul server HTTP, ma una directory a valle di tale radice denominata Document Root.
Il server HTTP non ha accesso a directory esterne alla Document Root.
La Document Root ha come path "/" e può essere omessa.
Il campo scheme «http» non può essere omesso, anche se sono comuni indirizzi Web in cui sia omesso. In questi casi provvede il browser Web a completare l'URL.
URL con schema ftp
user: nome utente. Valore di default: anonymous.
port: porta del server ftp. Valore di default: 21.
path: percorso della directory a cui si vuole puntare. Navigato tramite il comando CWD.
Come per HTTP: la Document Root ha come path «/» e può essere omessa.

Nota: il carattere «@» facente parte della password deve essere codificato tramite la sequenza di escape «%40» per non essere interpretato come separatore fra password e host.

Query String

Valida con il metodo HTTP, è una sequenza di caratteri appesa all'URL. Valida anche con il protocollo CoAP.

Scopo: trasferire informazioni dal client al server. Informazioni tipicamente ottenute mediante un modulo (form HTML) che il server ha presentato all'utente e in cui l'utente ha inserito dei valori. Ogni valore è associato ad un parametro. Le coppie valore/parametro sono assemblate per comporre la query string. Le coppie sono comunemente separate dal carattere "&". Tuttavia esisteva una raccomandazione del W3C per sostituire tale carattere con ";". Raccomandazione mai adottata. Non esiste un limite specifico al numero o alle dimensioni dei campi, alla lunghezza della query string, ed alla lunghezza totale dell'URL.

Unico limite: capacità dell'host che deve elaborare l'URL.

Fragment

Un hyperlink può avere come destination anchor non l'inizio di un documento HTML, ma un suo punto intermedio. In tal caso, all'URL che indirizza all'interno del documento si appende una sequenza chiamata fragment identifier.

In entrambi i casi il client (browser) acquisisce tale risorsa.

Primo URL: il browser si posiziona all'inizio del testo.

Secondo URL: il browser si posiziona nel punto intermedio identificato (nel codice HTML) dal tag. Nel caso in cui tale tag mancasse, il browser si posizionerebbe all'inizio del documento HTML.

URL relativo e URL assoluto

URL assoluto: URL in cui sono presenti i campi scheme e host. Necessario per identificare in modo univoco una risorsa nel Web.

URL relativo: URL in cui mancano i campi scheme e host.

Nel caso in cui dal campo path dell'URL relativo siano rimosse alcune componenti più significativi (partendo da sinistra), si ha ancora un URL relativo. Concatenando scheme, host e la parte di path rimossa con l'URL relativo è possibile ricostruire l'URL assoluto ed identificare così una risorsa.

Base URL: URL assoluto posto davanti all'URL relativo.

In un documento HTML che contiene link relativi ad altri documenti, la base URL può essere dichiarata esplicitamente. Altrimenti si usa come base URL l'URL della encapsulating resource (ossia del documento stesso).

L'utilizzo di URL relativi in documenti HTML è utile perché consente la portabilità dei documenti stessi. Consente di trasferire un intero sito Web da un host ad un altro senza dover modificare tutti i link interni. Gli URL assoluti cambiano, mentre gli URL relativi rimangono immutati.