LA POSTA ELETTRONICA

Posta Elettronica

- È uno dei servizi più diffusi ed importanti della rete Internet (Netflix?!?!).

- Nata inizialmente come servizio locale per mettere in comunicazione gli utenti di uno stesso grande sistema time-sharing.

- La sua nascita risale al 1971, quando Ray Tomlinson installò su ARPANET un sistema in grado di scambiare messaggi fra le varie università, ma chi ne ha realmente definito il funzionamento fu Jon Postel.

- Nel tempo, tramite lo sviluppo di protocolli per lo smistamento, è **diventata uno strumento di comunicazione su larghissima scala**.
- **É il canale di comunicazione ufficial dell'ateneo!!**

Posta – Attori/Struttura

- Il **Mail User Agent (MUA)** o client di posta (thunderbird, outlook,...), cioè un programma usato da un utente per inviare/consultare i messaggi. Può anche essere un'applicazione web (webmail).
- Il **Mail Submission Agent (MSA)**, che si occupa di ricevere i messaggi da un **MUA** ed inviarli ad un **MTA**.
- Il **Mail Transit Agent (MTA)**, che si occupa di ricevere mail da un **MSA** o da un altro **MTA**, e ad instradarle ad un altro **MTA** oppure ad un **LDA**.
- Il **Mail Delivery Agent (MDA)** o **Local Delivery Agent (LDA)**, che si occupa, se *la destinazione finale del messaggio è nel sistema corrente*, di consegnare il messaggio alla casella di posta dell'utente indicato.
- Il **Mail Access Agent (MAA)**, che permette di consultare/scaricare i messaggi.
- Il **Mail Retrival Agent (MRA)**, che scarica la posta da un **MAA** e la rende disponibile in locale.
- **MSA è integrato nel MTA, MRA è integrato nel MUA**.
- Ad eccezione delle iterazioni tra **MRA** e **MAA**,**tutte le comunicazioni fra i vari agenti (Agent) avvengono attraverso il protocollo SMTP.**

Posta – Flusso

1) Un utente (**mittente**) scrive una email usando un **MUA**.

2) **MUA** invia la mail ad un **MSA/MTA**.

3) **MTA** controlla l'indirizzo di destinazione (`utente@dominio`):
   - Se **dominio è tra quelli serviti** da **MTA** in questione (è cioè un indirizzo locale), ed utente è effettivamente valido, questa viene girata al **LDA**, che la consegna nella casella di posta associata, e il viaggio termina (punto **5**). In caso contrario l' **MTA** rifiuta il messaggio.
   - Se **invece l'indirizzo non è locale**,e **MTA** accetta di instradare il messaggio (relay), **MTA** mette il messaggio in una coda d'uscita e si procede.

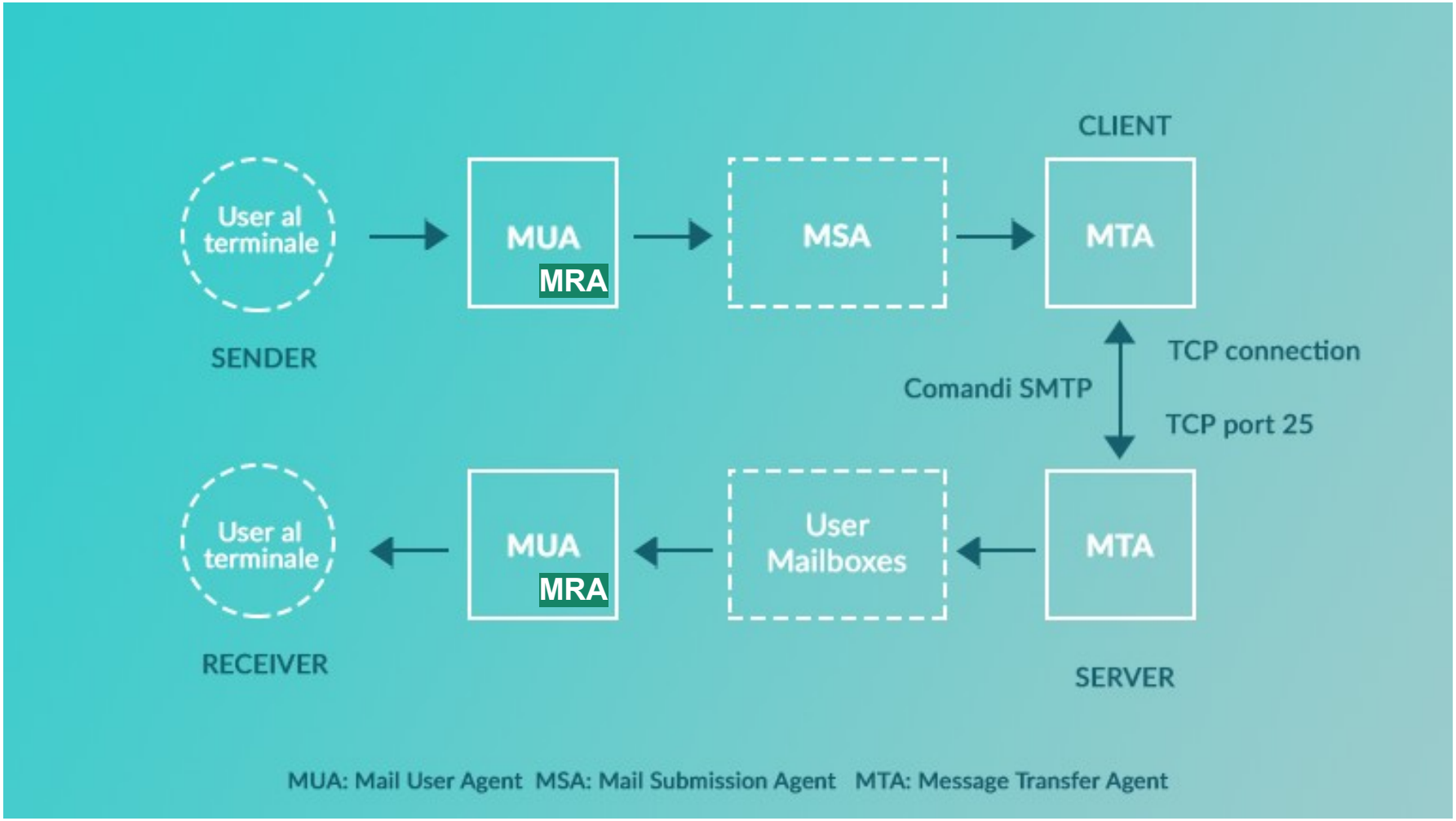
Posta - Flusso

4) Dopo aver estratto dalla coda il messaggio, **MTA** controlla quale sia il **record DNS MX** associato al dominio, o, ~~se non presente, cerca un **record 'A'** (relativo al nome dns dell'host dato)~~ e contatta l'**MTA** che risponde a quell'host, cercando di inviargli il messaggio.

- Se l'invio avviene correttamente, il messaggio è gestito da **MTA** di destinazione, che procede dal punto **3**).
- Se **MTA** contattato non risponde, il messaggio torna in coda.
- Se **MTA** contattato rifiuta il messaggio, oppure se il messaggio è stato troppo tempo in coda, viene mandata una mail all'indirizzo indicato dal (del) mittente, notificando la mancata consegna, e il procedimento termina.

5) A questo punto il messaggio si trova nella inbox del **destinatario**.

Posta - Flusso

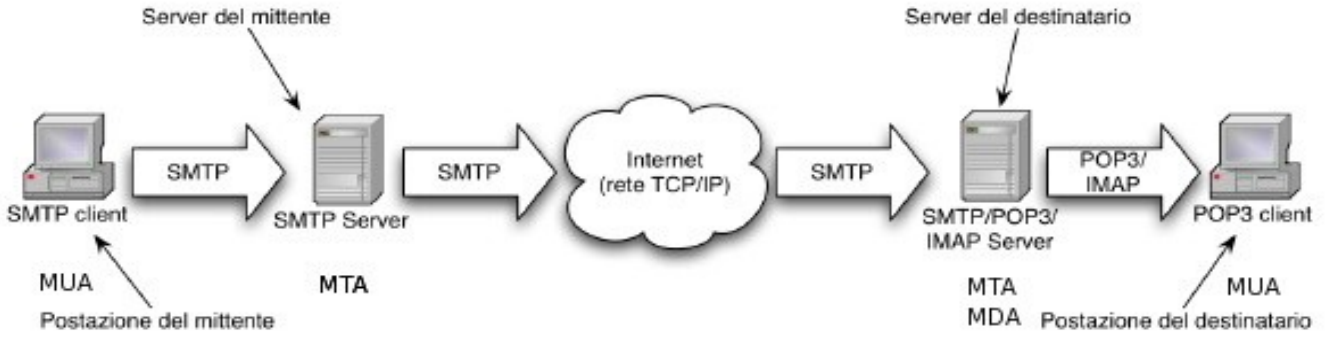


Posta - Relay

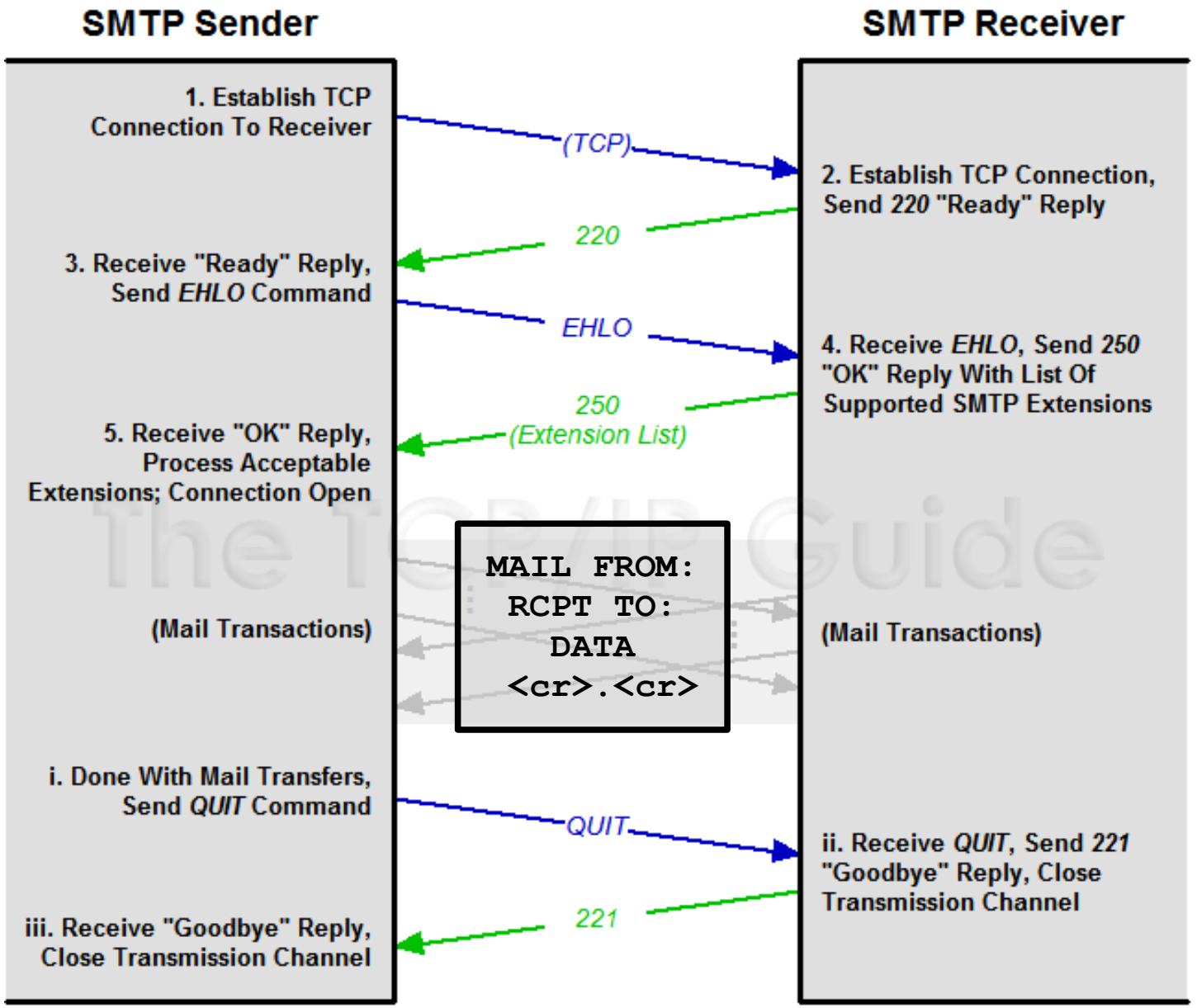
- Abbiamo detto, parlando del flusso, che un MTA può instradare verso terzi una email il cui destinatario non fa parte dei domini da lui serviti (punto 3). Questa operazione è detta **Relay**.
- **Relay**: è un servizio per il quale un server accetta della posta elettronica da un altro server, non perché sia indirizzata ad un suo dominio, ma per spedirla a terzi, a nome del server dal quale l'ha ricevuta.
- Il **relay non è sempre concesso**. Si accetta se la richiesta viene:
  - Da un host conosciuto (ad esempio nella propria rete aziendale)
  - Oppure da un utente autenticato.
- In tutti gli altri casi il messaggio viene rifiutato immediatamente (evitiamo di diventare **spammer** configurando un **Open Relay**).
- Attenzione: il servizio di posta **non** garantisce in alcun modo la consegna delle email né la notifica degli errori, né tantomeno l'identità del mittente o la privatezza della comunicazione.
  - **Per ovviare a questi problemi bisogna ricorrere a sistemi a livello applicazione.**

POSTA – SMTP, POP, IMAP

- Il protocollo usato tra **MUA** e **MTA** e tra **MTA** e **MTA** si chiama **Simple Mail Transfer Protocol (SMTP)** o Extended SMTP (ESMTP).
- La comunicazione tra **MTA** e **LDA** può avvenire sia internamente (ad esempio tramite scambio di file e/o memoria condivisa tra le componenti), oppure tramite il **Local Mail Transfer Protocol (LMPT)**, che è una versione semplificata di ESMTP**.**
- La comunicazione tra il **MAA** e il **MUA(MRA)** avviene tramite il **Post Office Protocol versione 3 (POP3)** o il **Internet Message Access Protocol versione 4 (IMAP4)**.



SMTP - Comunicazione



POSTA – MRA: POP3 vs IMAP4

- **IMAP** conserva le email sul server. La lettura, l'invio e la gestione possono avvenire anche da client desktop ma è il server a mantenere copia delle email inviate, ricevute, scritte.
- **POP**, invece, delega al dispositivo usato per la consultazione il compito di provvedere al salvataggio. Le email vengono scaricate sul dispositivo e la connessione è necessaria solo per inviare e ricevere posta.
- Quindi:
  - Con **imap** le email sono disponibili da qualsiasi dispositivo.
  - Con **pop** solo sul dispositivo da cui vengono scaricate.

- In realtà i **MUA(MRA)** hanno delle opzioni per **pop** che permettono di mantenere una copia dei messaggi sul server (**leave mail on server** o **keep copies**). **Imap** resta comunque preferibile per le performance ottimali anche su reti lente mentre **pop** risulta utile nel caso di connessioni non persistenti.

POSTA – MTA implementazioni

- **Sendmail**: primo vero demone che ha implementato il protocollo SMTP.
- **Qmail**: ideato da Dan Bernstein circa 15 anni fa, è un server di posta scalabile, performante, sicuro e portabile. Si dice sia il più sicuro... Attualmente è il secondo MTA più usato su internet. (Per approfondire: http://www.qmail-ldap.info).
- **Courier MTA**: è un server di posta / groupware integrato che poggia su vari protocolli: ESMTP, IMAP, POP3, SSL e HTTP. Sostanzialmente offre tutti i servizi di posta elettronica compreso anche un sistema di webmail. È quindi una soluzione completa per la gestione della posta, volendo è simile a Exchange. (http://www.courier-mta.org)
- **Exim: MTA** standard di Debian fino a qualche versione fa...
- **Zimbra**: suite completa per la gestione della posta (http://zimbra.org).
- **Postfix**.
- **MS Exchange**.

POSTA - Sendmail

- **Il primo MTA a fare uso di SMTP**.
- Utilizzato fino al 2005 come **MTA** da moltissimi server di posta.
- Purtroppo la progettazione rigida e la complessità della configurazione lo hanno reso via via sempre meno popolare fino all'estinzione.
- **Molti bug e problemi di sicurezza**.
- Stiamo ancora aspettando **sendmail X**.

POSTA - Postfix

- **MTA di riferimento in ambiente Linux**.
- **Facile da configurare, modulare, permette ad esempio diversi tipi di autenticazione (plain, sql, ldap, pam, ecc….)**.
- Permette di interfacciarsi facilmente con altri sistemi di controllo della posta (anti spam):
  - Real-time Blackhole List (**RBL**).
  - Sender Policy Framework (**SPF**).
  - Sistemi di **greylisting**.
  - Sistemi antispam basati sul contenuto come **SpamAssassin**.
  - Sistemi antivirus (**clamav**) ed altri ancora.

POSTA – MS Exchange

- **Software studiato per agevolare la collaborazione online tra vari utenti**.
- Introdotto sul mercato da Microsoft nel 1996 oggi è uno dei più potenti ed utilizzati mail server, soprattutto nelle realtà aziendali che utilizzano infrastrutture e tecnologie basate su prodotti di casa Microsoft.
- Le funzionalità principali di Microsoft Exchange sono la gestione centralizzata della posta elettronica, dei calendari e delle rubriche contatti, che possono essere condivisi tra i vari utenti di una rete aziendale.
- Il client più utilizzato per connettersi ad un server **Exchange** è **Microsoft Outlook** che è disponibile nella suite **Microsoft Office**. Mentre per l'accesso via web è disponibile l'interfaccia **OWA** (**Outlook Web Access**), pressochè identica a livello visuale al familiare Outlook. Esistono inoltre software di terze parti per interfacciarsi con Exchange.

POSTA – MAA Dovecot

- Permette di usare i protocolli **IMAP** e **POP**, e supporta, sia in consultazione che in consegna, diversi formati di memorizzazione della posta, quali `mbox`, `maildir`, `dbox`.
- Permette inoltre, tramite un plugin, di avere un sistema di filtri server side con cui smistare la posta degli utenti in vari folder, nonché inoltrarla ad altri indirizzi. Per gestire questi filtri lato server si usa il protocollo **SIEVE**.
- Permette anche di gestire la quota della posta per ogni utente.
- Permette autenticazione tramite **SQL**, **LDAP**, ecc….
- Altri MAA: `Courier Mail, Cyrus` ecc...

Confronto MTA

Fig. 4

	COURIER-MTA	EXIM	POSTFIX	SENDMAIL	QMAIL
Sicurezza	Media-alta	bassa-media	alta	bassa	alta / molto alta
Difficolta installazione	media	media	facile - media	facile	media-difficile
Difficoltà configurazione	media	facile-media	facile	difficile	facile
Performance	medie	medie	alte	basse	alte
Maturità	bassa	bassa	media	alta	media
documentazione	Poca	molta	media-molta	Molta	Molta
Features	Molte	medie-molte	media	Molte	poche (disponibili patch)


Sent Mail
Spam (372)
Trash

SPAM - Definizione

Uno o più messaggi non richiesti, inviati come parte di un più grande insieme di messaggi, tutti aventi contenuto sostanzialmente identico.

SPAM

- E’ possibile collocare le email indesiderate in **cinque** diverse categorie:
  - **Hoax**, ovvero le bufale e le catene di Sant’Antonio.
  - **Worm**, email mandate da virus.
  - **UCE**, Unsoliticed Commercial Email, email di spam dal contenuto commerciale.
  - **UBE**, Unsoliticed Bulk Email, email indesiderate inviate in grandi quantità.
  - Messaggi derivanti da iscrizioni a mailing list.
- **UCE** e **UBE** sono le maggiori fonti di spam.
- La battaglia contro lo spam è una *guerra infinita*, in cui spesso si vince qualche battaglia, ma in generale la proliferazione di nuove tecniche di spam inficia velocemente le nuove soluzioni trovate.


SYSADMIN
SPAM
SOLUZIONE
imgflip.com

SPAM

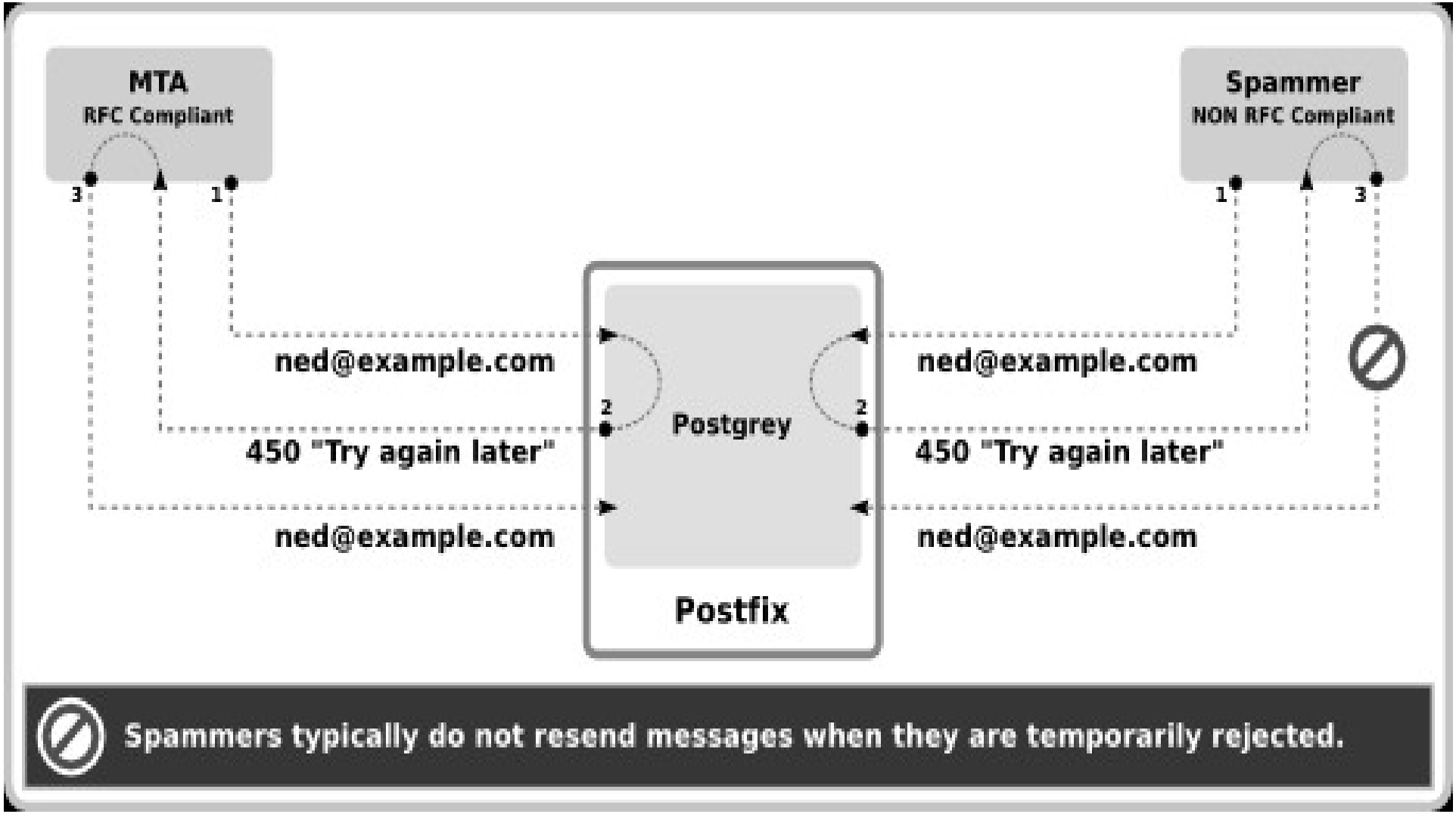
- Quindi ci arrendiamo? No cerchiamo di limitare i danni.
- Il problema **SPAM** **non è arginabile con una policy statica**, per quanto complessa essa sia.
  - *Infatti gli spammer inventano continuamente nuove strategie per aggirare i nostri filtri e, trovato un nuovo impedimento, provvedono ad elaborare un nuovo sistema per aggirarlo.*
- Possiamo solo decidere di **filtrare** le mail in ingresso con vari meccanismi ma tenendo sempre conto che la filtratura **non deve essere totale perché si rischia seriamente di compromettere il servizio scartando mail legittime**: gli utenti vogliono un sistema di filtratura perfetto ma guai a scartare una loro mail legale, importantissima, unica ecc ecc…

SPAM – Tecniche di difesa - Greylisting

1) una email, proveniente da un dominio **sconosciuto** viene **temporaneamente** rifiutata *restituendo un errore di "temporaneamente non disponibile:* `450 "try again later""`, per un numero preimpostato di volte **N**.
2) Ad ogni rifiuto l'MTA sender rimette la mail in coda e dopo un certo periodo di tempo **T** tenta il re-invio.
3) Al tentativo **N** la mail viene accettata dall'MTA receiver e l'MTA sender viene inserito in una **whitelist**, in tal modo tutte le email inviate da tale MTA vengano direttamente accettate.

- Postfix: demoni `postgrey` o `policyd`.

SPAM – Tecniche di difesa - Greylisting



SPAM – Tecniche di difesa - Greylisting

- **Problematiche:**
  - Questa tecnica, **di base**, alla lunga, perde di efficacia. I robot di spam implementano algoritimi che re-inviano più volte le email, aggirando il sistema. Sono stati implementati ulteriori ma più complessi controlli tenendo come base sempre il greylisting.
  - Si possono verificare lievi ritardi nella consegna delle email (**NxT**).
- **Vantaggi:**
  - Nessuna email lecita viene scartata: normalmente tutti i mail server senza fini maliziosi re-inviano la mail fino ad un ricevimento di un errore definitivo o l'accettazione.
  - Si ha comunque un buon filtraggio di base.
- NB: Questa è una **tecnica di base**, che viene modificata ed evoluta a seconda dell'evoluzione dello spam.

SPAM – Tecniche di difesa - RBL

- **RBL: Sono “liste nere” contenenti un elenco di IP che non sono “autorizzati” ad inviare e-mail**.
- **Problematiche:**
  - Difficile capire quali siano gli IP validi e quali no, spesso vengono scartate email legali e fatte passare mail di spam.
  - Gestite da terze parti con criteri “personali” su cui non si può influire, si può solo decidere se usarle oppure no.
- Alcuni criteri per la selezione degli IP da bloccare sono:
  - Tutti gli IP assegnati dinamicamente dai provider;
  - Tutti gli IP che inviano email senza passare da un mail server ufficiale (definito per ogni dominio da un record MX del DNS);
  - **IP segnalati come spammer dagli utenti**.

- Postfix: direttiva `reject_rbl_client` alla voce `smtpd_recipient_restrictions:`
  - `reject_rbl_client zen.spamhaus.org`

SPAM – Tecniche di difesa - SPF

- **SPF (Sender Policy Framework)**: È uno standard che in realtà non ha funzioni antispam nel senso lato del termine**. Si applica in ambito di risoluzione dei nomi (DNS) per cui si può dichiarare, tramite un record TXT (ovvero a testo libero) quali sono gli ip o i nomi che possono inviare mail per il dominio stesso**.
- In pratica si **crea una maschera per cui il mail server ricevente, se il TXT record è formattato nel modo corretto rispetto allo standard, può verificare se il server mittente è abilitato ad inviare mail**.
- **Problematiche:**
  - sono ancora moltissimi i domini che non implementano il record TXT;
  - l'implementazione può essere difficoltosa su strutture di una certa complessità.
  - gestione complicata del processo di forwarding delle mail.
- Esempio: `https://www.achab.it/achab.cfm/it/supporto/knowledge-base/mdaemon/informazioni-generali/KB50181`

- Postfix: nella direttiva `smtpd_recipient_restrictions` si inserisce la voce `check_policy_service spf`.

SPAM – Tecniche di difesa - Spamassassin

- **È un demone che utilizza un sistema di filtri su base euristica(*),** ovvero, il sistema prova ad “indovinare” se la mail è valida oppure no assegnandole un punteggio sulla base di vari aspetti.
- I vari aspetti possono essere:
  - Lingua della mail;
  - Presenza di tag html;
  - Presenza di parole chiave;
- A seconda del punteggio ottenuto dalla email e dalle soglie impostate, **Spamassasin** può:
  - Far passare tranquillamente la email;
  - **Applicare dei tag al subject segnalandola come spam all’utente**;
  - Scartarla e salvarla in una directory apposita.

(*)Euristico: Approccio che si basa sull’intuizione a partire dall’osservazione degli eventi da studiare

SPAM – Tecniche di difesa - Spamassassin

- Può essere “istruito” in base alle mail ricevute in precedenza e quindi migliorare la sua efficacia.
- **Problematiche:**
  - Per essere correttamente mantenuto necessita di continui aggiustamenti e una notevole esperienza.
  - Non esistono configurazioni predefinite che vadano bene per tutti: un esempio sono le politiche aziendali!! Esistono aziende che preferiscono ricevere montagne di spam per paura di perdere email dei loro clienti.

- Postfix: demone `spamassassin`.

SPAM – Tecniche di difesa – Controlli MTA

- Un **MTA** **può implementare dei controlli per ridurre il flusso dello spam robotico sfruttando il *formato del protocollo smtp***.
- Ad esempio in postfix vi sono delle direttive che possono essere usate nel file di configurazione **`main.cf`**:
  - `smtpd_helo_required = yes:` # controlla che il sender inzi la # comunicazione con il comando ehlo.
  - `smtpd_helo_restrictions:` # controlla chi può inziare un diagolo tramite # il comando ehlo. Opzioni:

    `permit_sasl_authenticated, permit_mynetworks, reject_non_fqdn_hostname, permit`
  - `smtpd_sender_restrictions:` # controlla chi, dopo essersi indentificato # con ehlo possa inviare email. Opzioni:

    `permit_sasl_authenticated, permit_mynetworks, reject_non_fqdn_sender, permit`
  - `smtpd_recipient_restrictions:` # effettua controlli sul destinatario. # Opzioni:

    `reject_non_fqdn_recipient, check_policy_service inet:127.0.0.1:10031, permit`

SPAM – Tecniche di difesa – AMAVIS

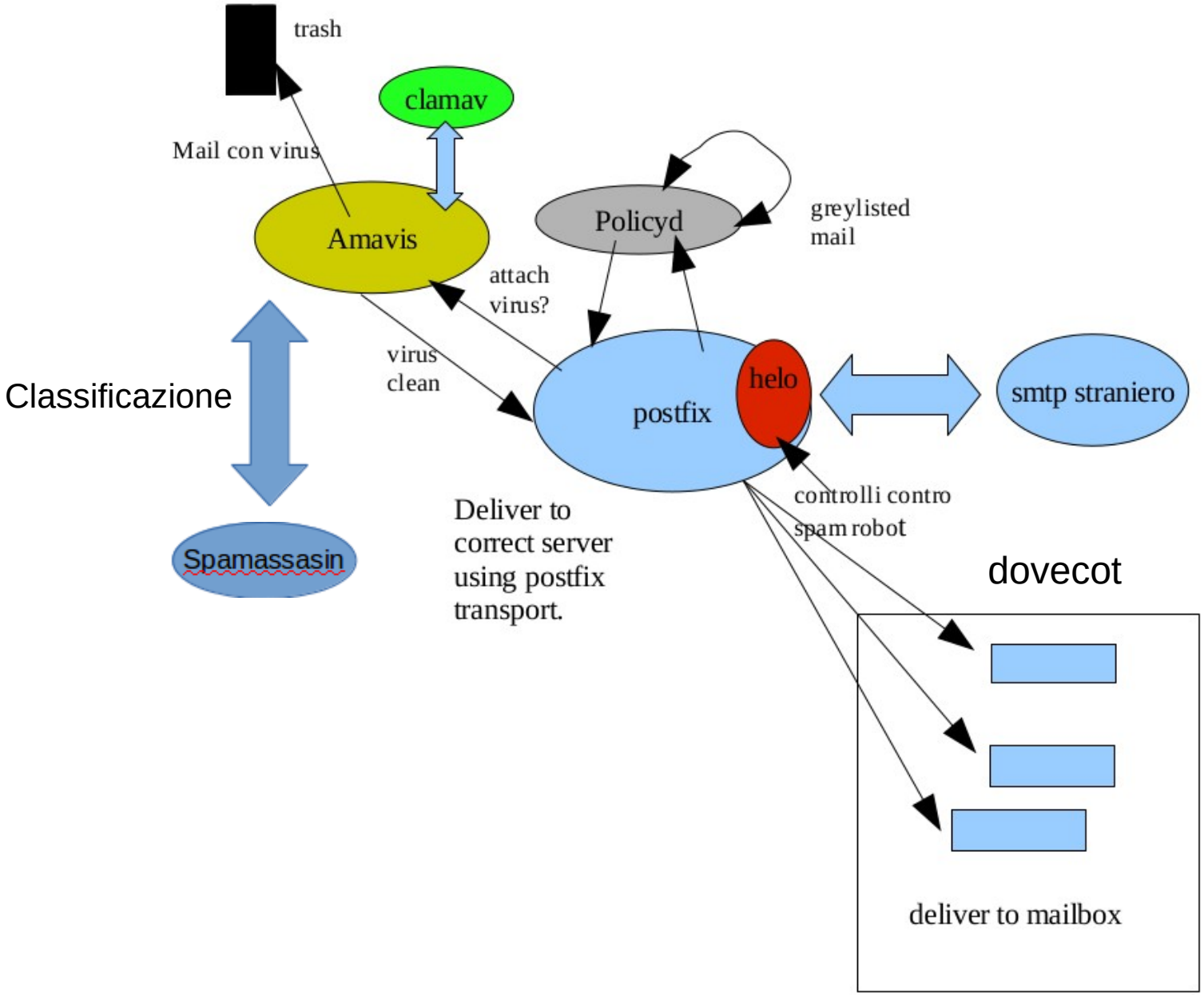
- È sostanzialmente un **super-demone** **che filtra i contenuti delle mail, implementando il trasferimento, l'elaborazione, la codifica delle stesse ed interfacciandosi con altri sistemi di filtraggio di spam e virus**.
- Essenzialmente può essere visto come un interfaccia tra un **MTA** (quindi usa **SMTP**) e altri sistemi di filtraggio dei contenuti (es. **Spamassasin** o **ClamAV**).
- Può essere usato per:
  - Rilevare virus, spam, contenuti vietati nelle mail ecc.
  - Bloccare, etichettare, redirezionare email a seconda del loro contentuto.
  - Mettere in quarantena o rilasciare messaggi.
  - Eliminare Virus dai messaggi tramite un antivirus esterno.

SPAM – Tecniche di difesa – AMAVIS

- Un utilizzo abbastanza comune di Amavis è quello di un sistema di filtraggio basato su:
  - **Postfix** come MTA;
  - **Spamassasin** come classificatore per lo spam;
  - **Clamav** come antivirus;
  - **Amavis** come gestore delle operazioni di antispam e antivirus.

- Postfix: demone `amavisd-new`

Esempio:postfix+amavis+greylisting+dovecot



Postfix – Installazione e test

- Installare postfix:

```
# apt install postfix
```

- Test:

```
# telnet localhost 25
```

- Se volete leggere il messaggio inviato potete installare `mutt`:

```
# apt install mutt
```

- Configurazione in `/etc/postfix`, in particolare i file `master.cf` e `main.cf`.

```
las@mylas:~$ telnet localhost 25
Trying ::1...
Connected to localhost.localdomain.
Escape character is '^]'.
220 mylas.localdomain ESMTP Postfix (Ubuntu)
ehlo localhost
250-mylas.localdomain
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-DSN
250 SMTPUTF8
MAIL FROM: las@mylas
250 2.1.0 Ok
RCPT TO: las@mylas
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Subject: Prova invio mail

Ciao questa è una prova.

Saluti
Las

.
250 2.0.0 Ok: queued as 5B4A3E9
```

Postfix – `master.cf`: descrive la struttura di postfix

```
#
==========================================================================
# service type  private unpriv  chroot  wakeup  maxproc command + args
#               (yes)   (yes)   (no)    (never) (100)
#
==========================================================================
smtp      inet  n       -       y       -       -       smtpd
smtps     inet  n       -       y       -       -       smtpd
pickup    unix  n       -       y       60      1       pickup
cleanup   unix  n       -       y       -       0       cleanup
qmgr      unix  n       -       n       300     1       qmgr
```

Postfix – `main.cf`: descrive il comportamento di postfix

```
smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)

# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache


smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated
defer_unauth_destination
myhostname = gundam.dsi.unive.it
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
mydestination = $myhostname, gundam, localhost.localdomain, localhost
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
```

Dovecot – Installazione

- Installare `dovecot` in Ubuntu/Debian:

```
# apt install dovecot dovecot-imapd
dovecot-pop3d dovecot-lmtpd
```

- Configurazione in `/etc/dovecot`, in particolare il file `dovecot.conf`.

Esempio di dovecot.conf

```
mail_location = mbox:~/mail:INBOX=/var/mail/%u
namespace inbox {
  inbox = yes
  location =
  mailbox Drafts {
    special_use = \Drafts
  }
  mailbox Junk {
    special_use = \Junk
  }
  mailbox Sent {
    special_use = \Sent
  }
  mailbox "Sent Messages" {
    special_use = \Sent
  }
  mailbox Trash {
    special_use = \Trash
  }
  prefix =
}
passdb {
  driver = pam
}
protocols = " imap pop3"
ssl = no
userdb {
  driver = passwd
}
```

!!! Warning !!!

- Importanti per un corretto funzionamento di un sistema di posta elettronica sono:
  - Sincronizzazione degli orologi (**NTP**), perchè i ticket hanno una durata temporale.
  - Corretta configurazione dei **DNS** sia diretta che inversa, altrimenti i vari attori possono non riconoscersi tra loro.
  - Correttezza dei certificati **SSL**, se si parla di crittografia è essenziale che i certificati che garantiscono l'autenticità di client e server siano validi. (Vedremo SSL più avanti).
  - **Manutenzione costante** del database utenti/gruppi/client.

POSTA -Progetto

- Realizzare un sistema di posta elettronica con MS Exchange.
- Realizzare un sistema di posta con Courier MTA.
- Studiare la suite Zimbra.
- Realizzare un sistema di posta elettronica su Linux con antispam e antivirus e webmail (roundcube).
- Realizzare un sistema di posta elettronica su Linux con autenticazione a scelta (sql) con antispam e antivirus e webmail (roundcube).
- Potete seguire queste guide:
  - https://www.exratione.com/2016/05/a-mailserver-on-ubuntu-16-04-postfix-dovecot-mysql/
  - https://noviello.it/come-installare-postfix-dovecot-mysql-spamassassin-su-ubuntu-18-04-lts/

Avete un Server di Posta!