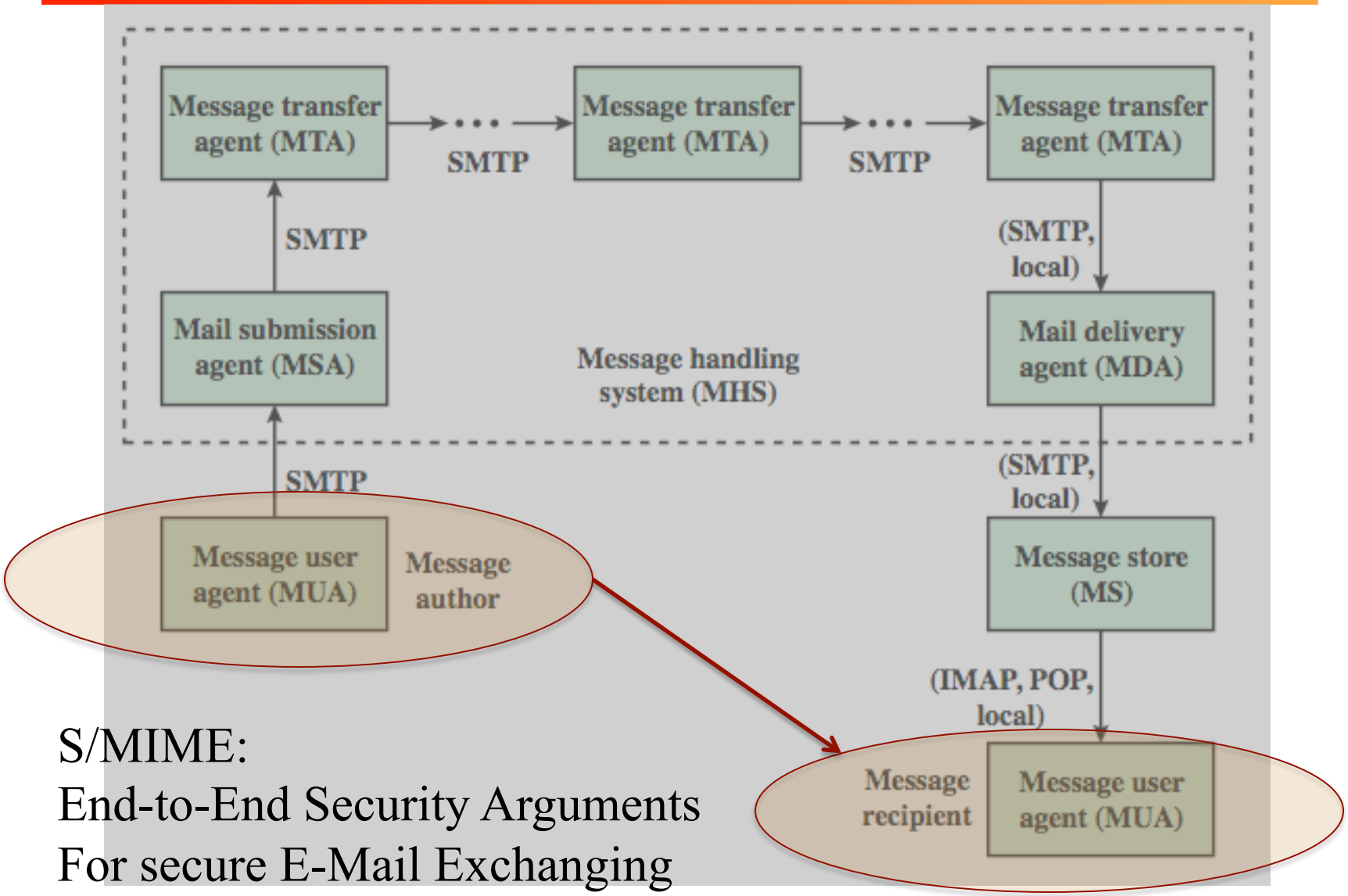


# Segurança do Correio Eletrónico

# S/MIME

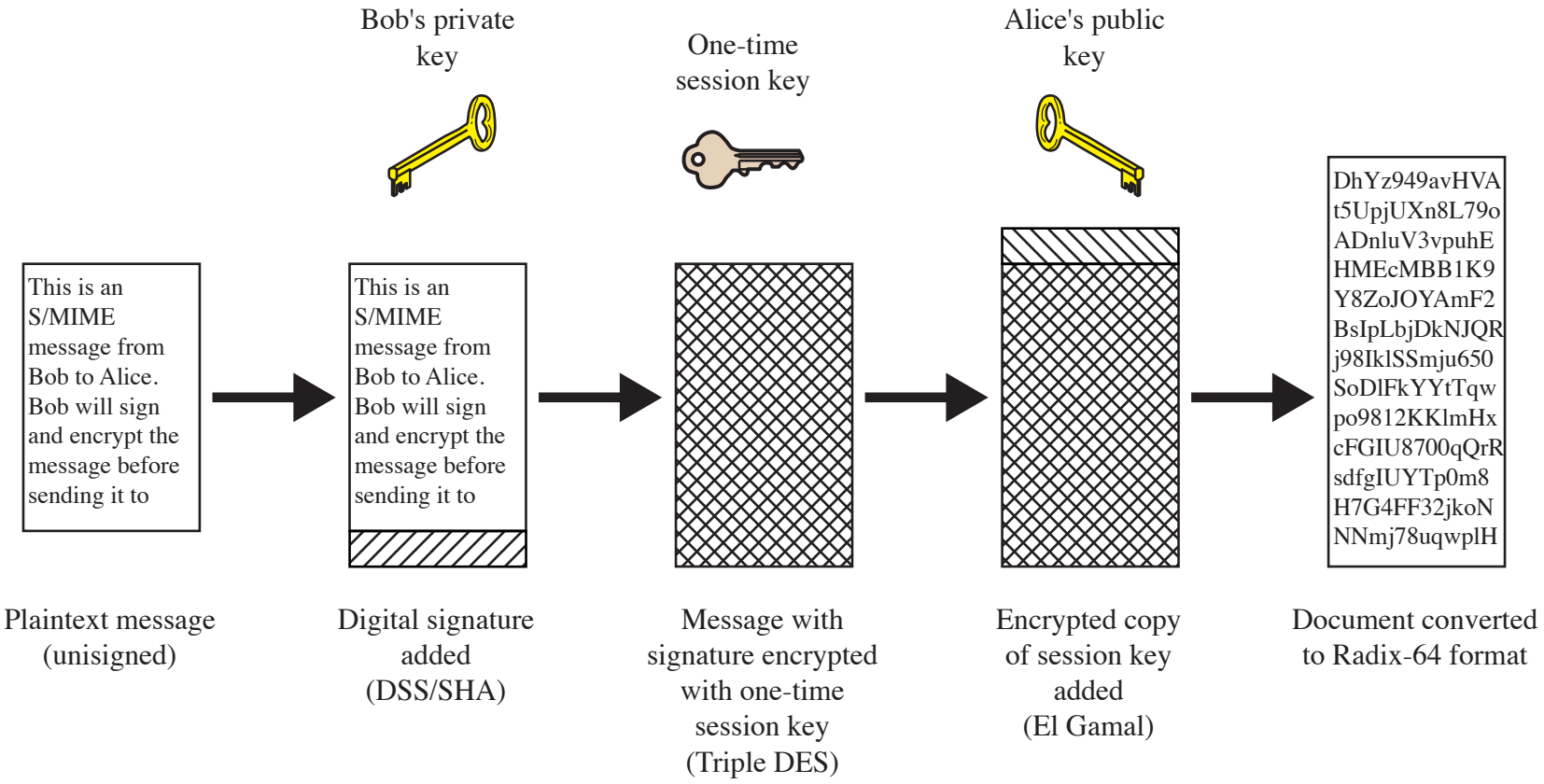
# DKIM

# Arquitetura: Sistema EMail Internet




S/MIME:
End-to-End Security Arguments
For secure E-Mail Exchanging

# Typical S/MIME process




- Cryptographic process very similar to PGP
- Standardization over multipart formats

# Tópicos

- Formato de normalização MIME
- Normalização S/MIME
- DKIM

# Normalização S/MIME (IETF)

- Definido nos:

  RFCs 3370, 3850, 3851, 3852

# Normalização S/MIME

- Extensão de segurança em relação à normalização MIME (Internet E-mail Format)
- Baseado na normalização de esquemas criptográficos estabelecidos pela *RSA Data Security*$^{TM}$
  - Padrões PKCS para representação e utilização de:
    - Assinaturas Digitais de Chave Pública
      - => Autenticação / Assinatura de Mensagens (com conteúdo MIME)
    - Envelopes de chave pública
      - => Confidencialidade / Envelopes de Chave Pública para envio de Chaves (OTK) usadas para cifrar mensagens
    - Conteúdos confidenciais com criptografia simétrica
  - Codificação BASE 64 (com base em carateres ASCII) para representação "*printable*" de informação criptográfica

# S/MIME e sua relevância

- Embora PGP e S/MIME sejam ambos normalizados pelo IETF, S/MIME emergiu (sendo cada vez mais) a normalização industrial "de facto" para segurança de mensagens de correio electrónico
  - Principais aplicações (MUA - *Mail User Agents ou MAIL Readers/Viewers*) disponíveis por parte dos principais intervenientes no mercado
    - Ex., MS Outlook, Apple Mail, …. Etc

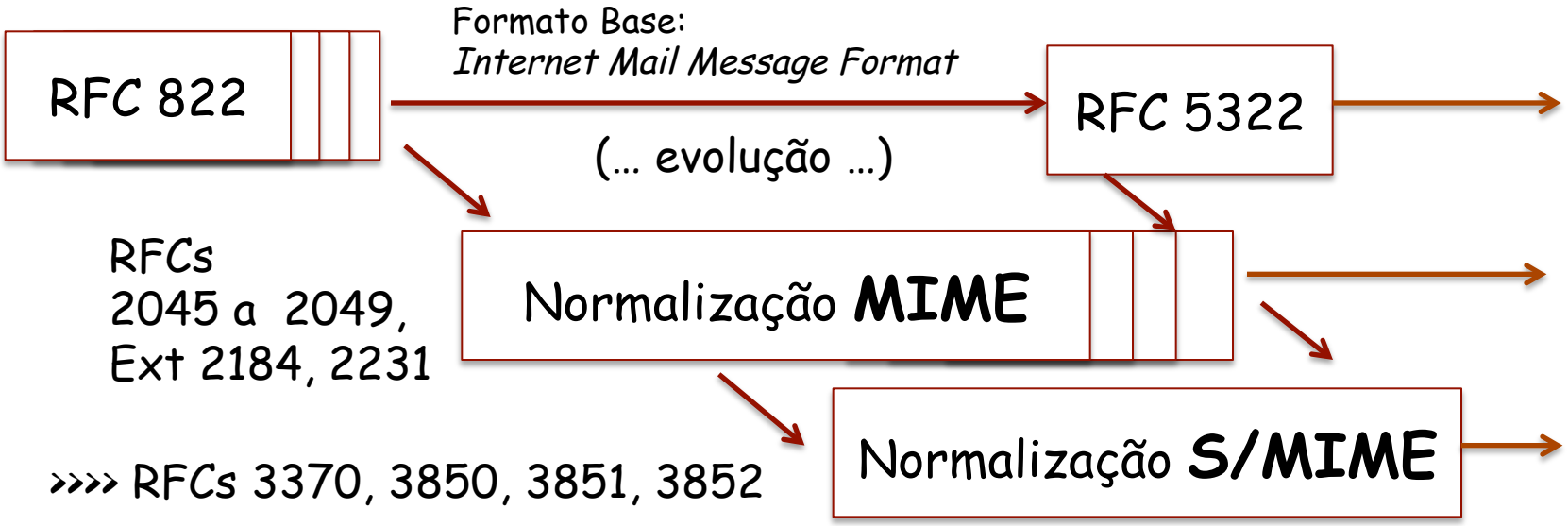
- O sistema PGP é uma alterantiva e escolha para utilização pessoal (ex., implementações em ferramentas de SW aberto)
- Também usado como referência de abordagem pioneira de segurança e privacidade de uso de sistemas Email
  - Base de inspiração para outras abordagens: ex., *Messaging Systems Security*, File-Encryption Tools, …

# S/MIME como extensão MIME

- Para se compreender a abordagem S/MIME, o ponto de partida é a compreensão do formato MIME, como formato de representação de mensagens com partes de conteúdos multimédia
  - ***MIME: Multipurpose Internet Mail Extensions***
  - Atenção: o formato MIME não trata de segurança
    - Apenas trata da normalização da representação e da codificação de mensagens contendo múltiplas partes ou anexos com conteúdos multimédia normalizados: texto ASCII, UNICODE, texto-enriquecido (ex., HTML), imagens, áudio, vídeo, ou até conteúdos visualizáveis por aplicações específicas

# Compreensão da abordagem S/MIME

- O ponto de partida para a compreensão da normalização S/MIME é então:
  - O conhecimento da origem e compreensão da normalização MIME
  - Que por sua vez foi uma normalização sobre a representação inicial de mensagens Email Internet, inicialmente definido pelo IETF RFC 822 e evoluções até Versão mais recente dessa normalização: RFC 5322 (*Internet Mail Message Format* ).



# Tópicos

- Formato de normalização MIME
- Normalização S/MIME
- DKIM

# IETF RFC 5322 (*Base IMF*)

- Formato base: **text messages**
- Estrutura de uma mensagem:
  - **Envelope**
    - Informação de controlo associada ao encaminhamento e entrega da mensagem da origem até ao destino
  - **Conteúdo (Cabeçalho + Corpo)**
    - Informação de controlo e dados (objeto) da mensagem
- RFC 5322: evolução apenas aplicável ao conteúdo
- Mas o conteúdo inclui também um conjunto dos campos do cabeçalho, informação usada para facilitar a aquisição e processamento por parte do SW envolvido na transmissão, e controlo de entrega ou recepção das mensagens

# RFC 5322

- **Cabeçalho:**
  - Um conjunto de linhas de texto (ASCII)
  - Cada linha tem um formato < KEYWORD: VALUE >

- **Corpo**
  - Parte não restritiva que contem no essencial uma representação dos dados da mensagem (em codificação texto ASCII)

- Cabeçalho e Corpo são separados por uma linha em branco

# RFC 5322 format

**Can include other lines:**
**Keyword: Argument Line**
**…**
**Message-ID: 0x31a2341ab87**

```
From: <joseph.neubauer@hp.com>
To: <jones@demo.com>
Subject: Report Status
Date: Tue, 5 Nov 2002 08:45:41 -0500
```

Header

<blank line>

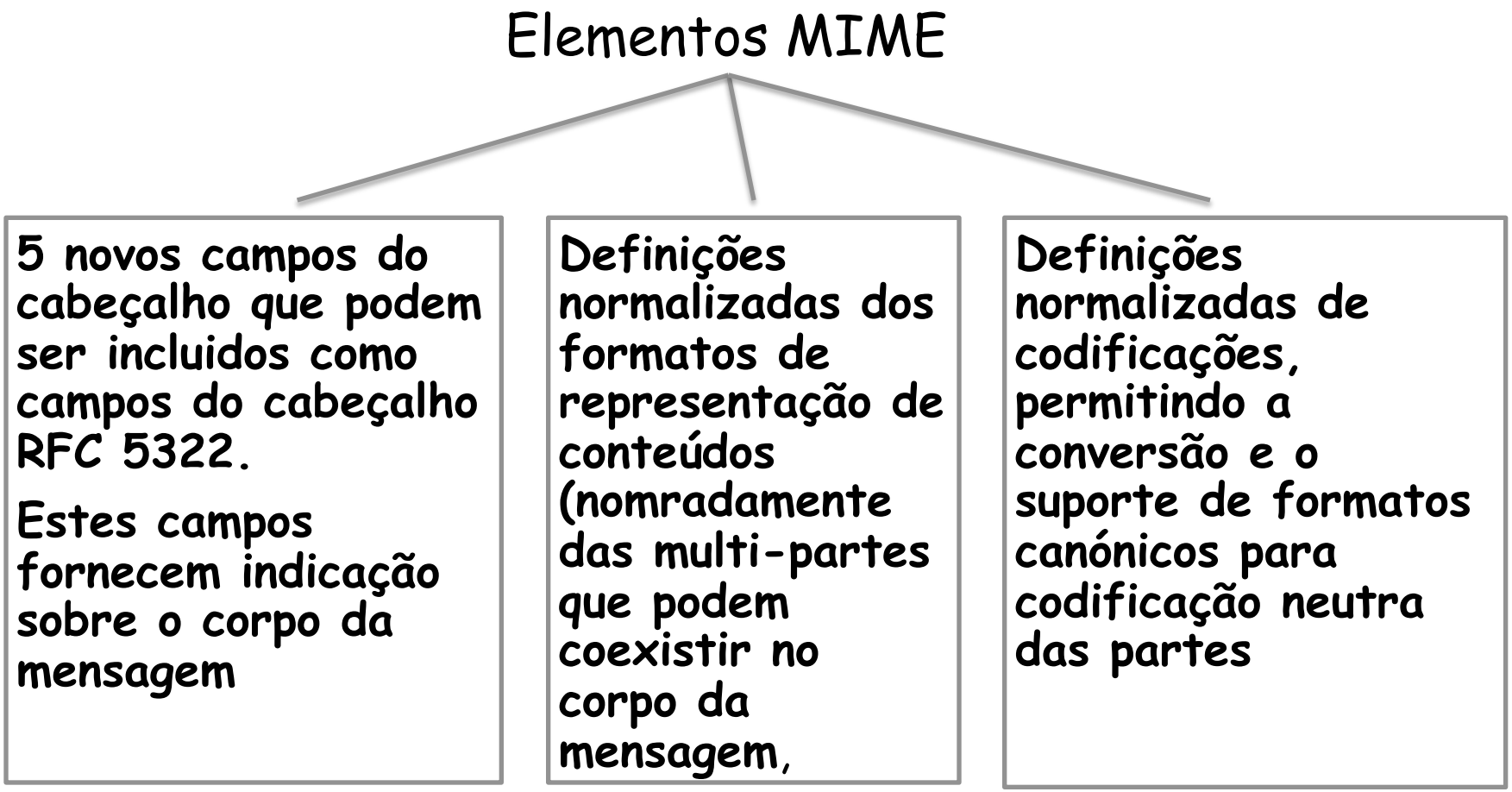
Body
(Contents)

# Extensões MIME

- Extensão de RFC 5322 mantendo a **compatibilidade de envio das mensagens através do protocolo STORE & FORWARD SMTP**
- As extensões MIME são **compatíveis e transparentes face à normalização RFC 5322**

- Apenas trata de:
  - Adicionar mais linhas KEYWORD: VALUE ao cabeçalho
  - E acrescentar linhas de controlo do tipo KEYWORD: VALUE também ao corpo de representação dos dados da mensagem
    - Para incluir a represetação e codificação de diferentes partes em diferentes formatos multimédia

# Quais as extensões introduzidas ?



# Extensões MIME

Extension Header-Fields

- **MIME-Version**
  - Identificação da Versão (ex., 1.0)
  - Indica conformidade com RFCs 2045 and 2046
- **Content-Type**
  - Detalhe do tipo de conteúdo (o que permite reconhecer o conteúdo e visualizar o conteúdo: *MUA built-in functions* ou *external viewers*
  - Ou gestão do conteúdo com base em alguma funcionalidade específica
- **Content-Transfer-Encoding**
  - Indica o tipo de transformação requerida para representar o corpo das mensagens, de forma neutra para o transporte/encaminhamento e entrega da mensagem
- **Content-ID**
  - Identifica as entidades MIME como múltiplos contextos representados (múltiplas partes ou MULTIPARTS)
- **Content-Description**
  - Descrição textual de cada parte no corpo da mensage (informação útil no caso de não ser suportada a visualização automática)
  - **Content-disposition**
  - Informação de controlo de *rendering* (ex., visualização automática da parte ou inclusão como anexo)

# MIME format mail message

```
From: John Doe <example@example.com>MIME-Version: 1.0
Content-Type: multipart/mixed;
              boundary="XXXXboundary text”

This is a multipart message in MIME format.
--XXXXboundary text
Content-Type: text/plain

this is the body text

--XXXXboundary text
Content-Type: text/plain;
Content-Disposition: attachment;
filename="test.txt”

this is the attachment text

--XXXXboundary text--
```

Linha em Branco

# Formato RFC 5322

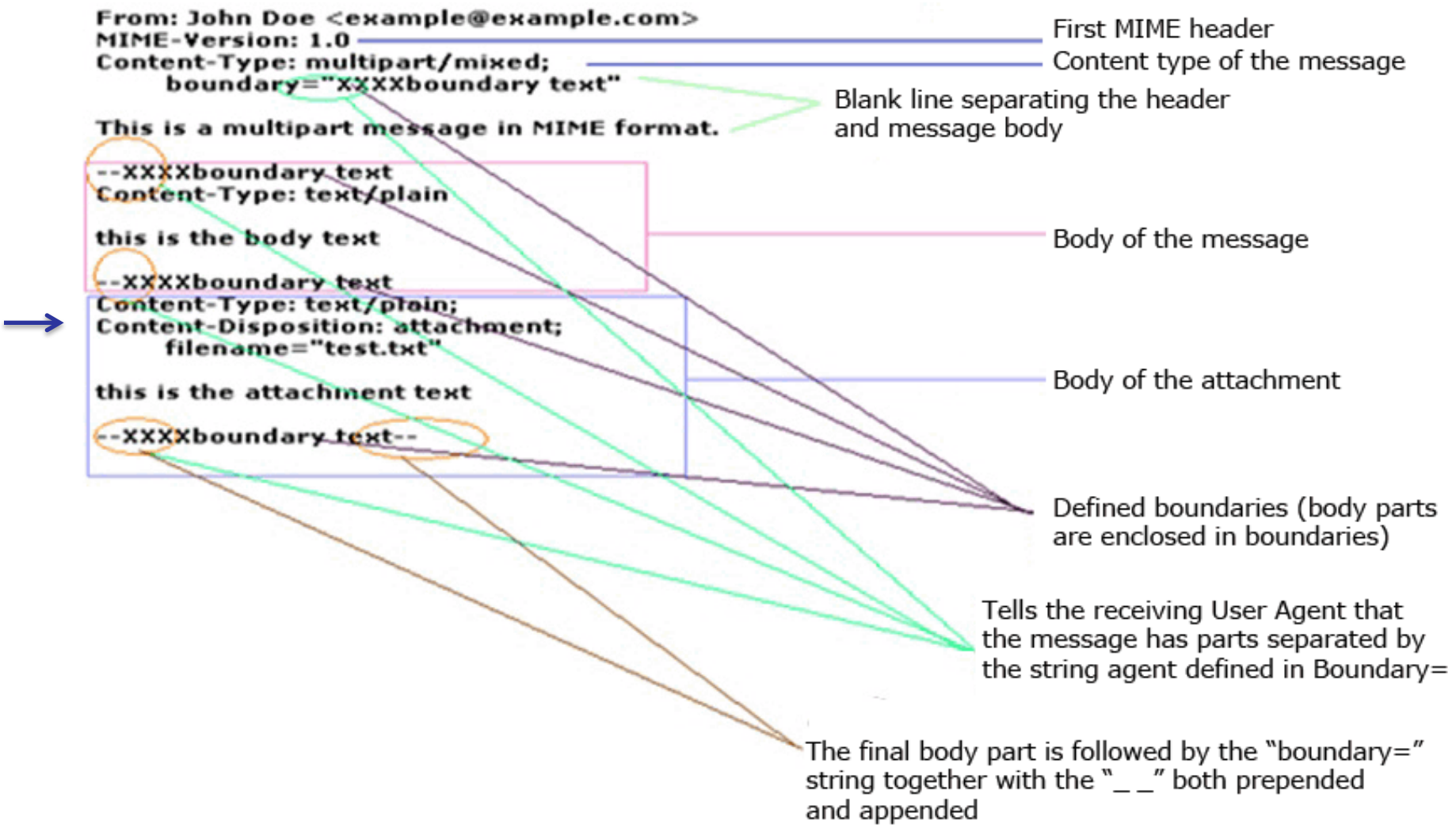
**Can include other lines:**
**Keyword: Argument Line**
**…**
**Message-ID: 0x31a2341ab87**

```
From: <joseph.neubauer@hp.com>
To: <jones@demo.com>
Subject: Report Status
Date: Tue, 5 Nov 2002 08:45:41 -0500
MIME-Version: 1.0
Content-Type: multipart/alternative;

This is a multi-part message in MIME format.
Your email client may not be able to understand MIME.

------=_NextPart_000_0000_01C284A7.B8F453A0
Content-Type: text/plain;
      charset="iso-8859-1"
Content-Transfer-Encoding: 7bit
<blank line>
Update Status Report
-----
neubauerNeubauer, Joe
masonMason, Kevin
jonesJones, Mosby
havensHavens, Mary
There was 1 error
------=_NextPart_000_0000_01C284A7.B8F453A0
Content-Type: text/html;
      charset="iso-8859-1"
Content-Transfer-Encoding: quoted-printable

<h2>Update Status Report</h2>
<hr><p>
<table border>
<tr><td>neubauer</td><td>Neubauer, Joe</td><td><img =
src=3Dhttp://router/images/up.gif></td></tr>
<tr><td>mason</td><td>Mason, Kevin</td><td><img =
src=3Dhttp://router/images/up.gif></td></tr>
<tr><td>jones</td><td>Jones, Mosby</td><td><img =
src=3Dhttp://router/images/up.gif></td></tr>
<tr><td><b>havens</b></td><td><b>Havens, Mary</b></td><td><img =
src=3Dhttp://router/images/dn.gif></td></tr>
</table>
<p>
<font color=3Dblue size=3D5><b>There was 1 error</b></font>

------=_NextPart_000_0000_01C284A7.B8F453A0--
```

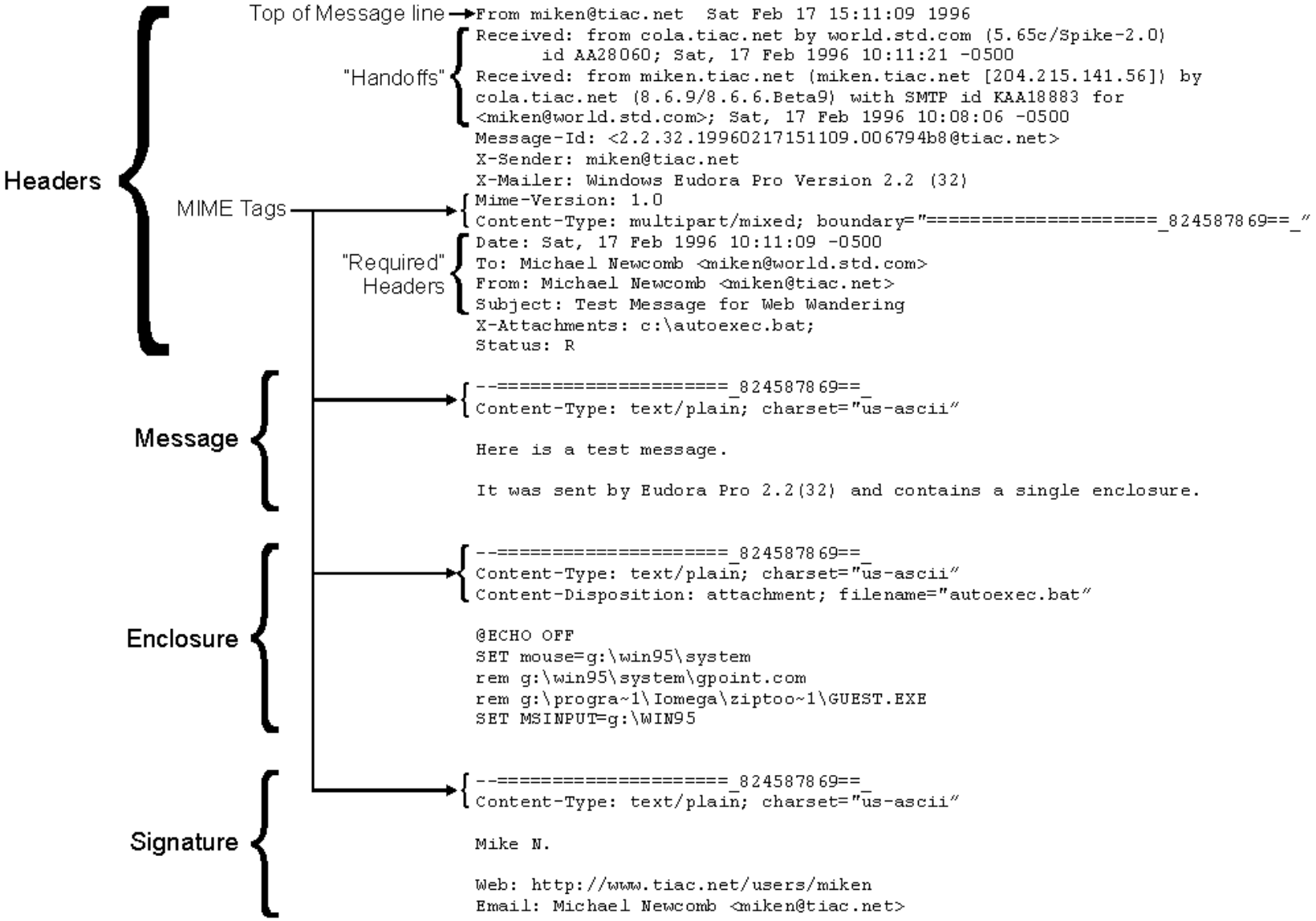
Header

MIME Extensions
Header fields

# Exemplo de formatação MIME



# Outro exemplo de formatação MIME



# Extensões MIME

Extension Content-Types

Main Types

- Text
- Multipart
- Message
- Image
- Video
- Audio
- Application

| Type | Subtype | Description |
|---|---|---|
| Text | Plain | Unformatted text; may be ASCII or ISO 8859. |
| | Enriched | Provides greater format flexibility. |
| Multipart | Mixed | The different parts are independent but are to be transmitted together. They should be presented to the receiver in the order that they appear in the mail message. |
| | Parallel | Differs from Mixed only in that no order is defined for delivering the parts to the receiver. |
| | Alternative | The different parts are alternative versions of the same information. They are ordered in increasing faithfulness to the original, and the recipient's mail system should display the "best" version to the user. |
| | Digest | Similar to Mixed, but the default type/subtype of each part is message/rfc822. |
| Message | rfc822 | The body is itself an encapsulated message that conforms to RFC 822. |
| | Partial | Used to allow fragmentation of large mail items, in a way that is transparent to the recipient. |
| | External-body | Contains a pointer to an object that exists elsewhere. |
| Image | jpeg | The image is in JPEG format, JFIF encoding. |
| | gif | The image is in GIF format. |
| Video | mpeg | MPEG format. |
| Audio | Basic | Single-channel 8-bit ISDN mu-law encoding at a sample rate of 8 kHz. |
| Application | PostScript | Adobe Postscript format. |
| | octet-stream | General binary data consisting of 8-bit bytes. |

# Extensões MIME

Extension Transfer Encodings

Main Encodings

| | |
|---|---|
| 7bit | The data are all represented by short lines of ASCII characters. |
| 8bit | The lines are short, but there may be non-ASCII characters (octets with the high-order bit set). |
| binary | Not only may non-ASCII characters be present but the lines are not necessarily short enough for SMTP transport. |
| quoted-printable | Encodes the data in such a way that if the data being encoded are mostly ASCII text, the encoded form of the data remains largely recognizable by humans. |
| base64 | Encodes data by mapping 6-bit blocks of input to 8-bit blocks of output, all of which are printable ASCII characters. |
| x-token | A named nonstandard encoding. |

# Exemplo de mensagens complexas

Próxima mensagem: (defnição com base no RFC 2045), mostra como se pode endereçar formatos complexos numa mesma mensagem

A mensagem em causa compreende:

- 5 partes que serão visualizadas em série :
  - Duas partes introdutórias (plain text, US-ASCII)
  - Uma parte que é por sua vez uma mensagem com multipartes embebidas
  - Uma parte de formato *richtext* (HTML)
  - Uma parte de fecho que encapsula dados com várias codificações de texto (US ASCII e ISO 8859-1)

- A mensagem com multipartes será mostrada em paralelo, sendo no caso:
  - Uma imagem JPEG codificada em BASE 64
  - Um fragmento áudio (MU LAW) codificado em BASE 64

# Example (cont)

Two Introductory Text Parts

Embedded Multipart
- Audio
- Picture

Rich Text Part

closing encapsulated text message in a non-ASCII character set.

```
MIME-Version: 1.0
From: Nathaniel Borenstein <nsb@bellcore.com>
To: Ned Freed <ned@innosoft.com>
Subject: A multipart example
Content-Type: multipart/mixed;
    boundary=unique-boundary-1
This is the preamble area of a multipart message. Mail readers that understand multipart format should ignore this pre
  If you are reading this text, you might want to consider changing to a mail reader that understands how to properly c
  multipart messages.

  --unique-boundary-1
   ...Some text appears here...
[Note that the preceding blank line means no header fields were given and this is text, with charset US ASCII.  It cou
  been done with explicit typing as in the next part.]

--unique-boundary-1
Content-type: text/plain; charset=US-ASCII
This could have been part of the previous part, but illustrates explicit versus implicit typing of body parts.

--unique-boundary-1
Content-Type: multipart/parallel;   boundary=unique-boundary-2

--unique-boundary-2
Content-Type: audio/basic
Content-Transfer-Encoding: base64
   ... base64-encoded 8000 Hz single-channel mu-law-format audio data goes here....

--unique-boundary-2
Content-Type: image/jpeg
Content-Transfer-Encoding: base64
   ... base64-encoded image data goes here....

--unique-boundary-2--
--unique-boundary-1
Content-type: text/enriched

This is <bold><italic>richtext.</italic></bold> <smaller>as defined in RFC 1896</smaller>

  Isn't it <bigger><bigger>cool?</bigger></bigger>

--unique-boundary-1
Content-Type: message/rfc822

From: (mailbox in US-ASCII)
To: (address in US-ASCII)
Subject: (subject in US-ASCII)
Content-Type: Text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: Quoted-printable

   ... Additional text in ISO-8859-1 goes here ...

--unique-boundary-1--
```

Viewed in Parralel

# Formatos nativos vs canónicos MIME

Um aspecto importante na normalização MIME é a normalização de formatos canónicos

- **Um formato canónico** é um formato apropriado para um *Content-Type* que será normalizado em diferentes sistemas (diferentes ferramentas, diferentes fornecedores, …)

- Por outro lado, **um formato nativo** será um formato que tende a ser particular num certo sistema ou numa verta implementação

# Native vs. Caninical Forms (RFC 2049)

| | |
|---|---|
| **Native Form** | The body to be transmitted is created in the system's native format. The native character set is used and, where appropriate, local end-of-line conventions are used as well. The body may be a UNIX-style text file, or a Sun raster image, or a VMS indexed file, or audio data in a system-dependent format stored only in memory, or anything else that corresponds to the local model for the representation of some form of information. Fundamentally, the data is created in the "native" form that corresponds to the type specified by the media type. |
| **Canonical Form** | The entire body, including "out-of-band" information such as record lengths and possibly file attribute information, is converted to a universal canonical form. The specific media type of the body as well as its associated attributes dictate the nature of the canonical form that is used. Conversion to the proper canonical form may involve character set conversion, transformation of audio data, compression, or various other operations specific to the various media types. If character set conversion is involved, however, care must be taken to understand the semantics of the media type, which may have strong implications for any character set conversion (e.g. with regard to syntactically meaningful characters in a text subtype other than "plain"). |

# Tópicos

- Formato de normalização MIME
- Normalização S/MIME
- DKIM

# Extensões S/MIME em relação a MIME

- Extensões

  RFCs 3370, 3850, 3851, 3852

- Serviços de Segurança em S/MIME

  No essencial as extensões permitem suportar a mesma funcionalidade e serviços de segurança, como já anteriormente se viu para o caso do PGP

  - Autenticação das mensagens (Usando Assinatiuras Digitais com Métodos de Assimétricos ou de Chave Pública: DSA ou RSA)
  - Integridade (pelas funções de síntese normalizadas)
  - Confidencialidade (com geração de OTKs e criptografia simétrica normalizada)
  - Compressão

- A autenticação baseia-se na normalização X509 (com base em certificados de chaves públicas)

# S/MIME vs. PGP

Diferenças notórias:

## Modelo de confiança dos certificados de chave pública

- A autenticação baseia-se também na normalização X509 (com base em certificados de chaves públicas) ou infraetsruturas do tipo PKI
  - Gestão de certificados digitais emitidos por CAs (de confiança)
  - Ou tidos como confiáveis e assim aceites pelos utilizadores
  - Mas não existe um processo de gestão de confiança nos moldes da gestão WEB-OF-TRUST do modelo PGP e seu processamento de chaveiros de chaves-públicas e chaves-privadas

## Geração de chaves OTK para mensagens confidenciais

- Contrariamente ao sistema PGP, em S/MIME não se define ou não se promove um modelo de processamento normalizado para geração das chaves para cifra das mensagens

# Extensões S/MIME

## S/MIME Content Types

- Enveloped Data
- Signed Data
- Clear-Signed Data
- Signed and Enveloped Data

# *Clear Signing* (ou Assinatura em Claro)

- Suportado por um conteúdo (MIME Content Type) MULTIPART com um S/MIME SubType: SIGNED
- O processamento da assinatura digital não envolve qualquer transformação da mensagem a assinar
- Os destinatários (recipentes) que implementam a norma MIME mas não S/MIME conseguem ler estas mensages (só não conseguem validar ou verificar a assinatura e assim não verificar a sua autenticidade)

# Extensões S/MIME:
- Multipartes (c/ S/MIME *Content-Types*)

**Enveloped data**

- Consiste em conteúdo cifrado de qualquer tipo MIME, contendo envelope de chave OTK usada na cifra (gerada pelo emissor) distribuida a um ou mais destinatários, usando as respetivas chaves públicas (conhecidos os respetivos certificados pelo emissor)

**Signed data**

- Assinatuar digital (DSA, RSA) sobre a síntese SHA-1 do conteúdo, usando a chave privada do emissor
- Conteúdo e assinatura codificados em base64 encoding
- Os dados assinados só podem ser processados e vistos no MUA destino se este implementar S/MIME

**Clear-signed data**

- Parte com assinatuar digital (mas conteºudo em claro)
- Recipientes-destinatários apenas implementando MIME conseguem processar e visualizar o conteúdo (embora não processem a assinatura)

**Signed and enveloped data**

- "Signed-only" e "encrypted-only" entities may be nested
- Dados cifrados podem ser "signed-data", "enveloped data" ou "clear-signed data", de acordo com uma ordem repersentada neste content-type

# S/MIME: Content Types

Multipart

Signed

Application

PKCS7-MIME

Signed Data

Enveloped Data

Degenerate Signed Data

Signed Data

Content Types MIME

Content Types S/MIME

| Type | Subtype | smime Parameter | Description |
|---|---|---|---|
| Multipart | Signed | | A clear-signed message in two parts: one is the message and the other is the signature. |
| Application | pkcs7-mime | signedData | A signed S/MIME entity. |
| | pkcs7-mime | envelopedData | An encrypted S/MIME entity. |
| | pkcs7-mime | degenerate signedData | An entity containing only public-key certificates. |
| | pkcs7-mime | CompressedData | A compressed S/MIME entity. |
| | pkcs7-signature | signedData | The content type of the signature subpart of a multipart/signed message. |

# Funções criptográficas em S/MIME

SHA-1
(ou MD5 for Retro-Compatibility)

DSS (ou DSA)
RSA
(512, 768, 1024 bits)

DH, ElGammal,
RSA
(512, 768, 1024 bits)

3DES
AES
RC2/40 (… e alguns outros)

HMAC (SHA-1)

| Function | Requirement |
|---|---|
| Create a message digest to be used in forming a digital signature. | MUST support SHA-1.<br>Receiver SHOULD support MD5 for backward compatibility. |
| Encrypt message digest to form a digital signature. | Sending and receiving agents MUST support DSS.<br>Sending agents SHOULD support RSA encryption.<br>Receiving agents SHOULD support verification of RSA signatures with key sizes 512 bits to 1024 bits. |
| Encrypt session key for transmission with a message. | Sending and receiving agents SHOULD support Diffie-Hellman.<br>Sending and receiving agents MUST support RSA encryption with key sizes 512 bits to 1024 bits. |
| Encrypt message for transmission with a one-time session key. | Sending and receiving agents MUST support encryption with tripleDES<br>Sending agents SHOULD support encryption with AES.<br>Sending agents SHOULD support encryption with RC2/40. |
| Create a message authentication code | Receiving agents MUST support HMAC with SHA-1.<br>Sending agents SHOULD support HMAC with SHA-1. |

# Processamento de mensagens S/MIME

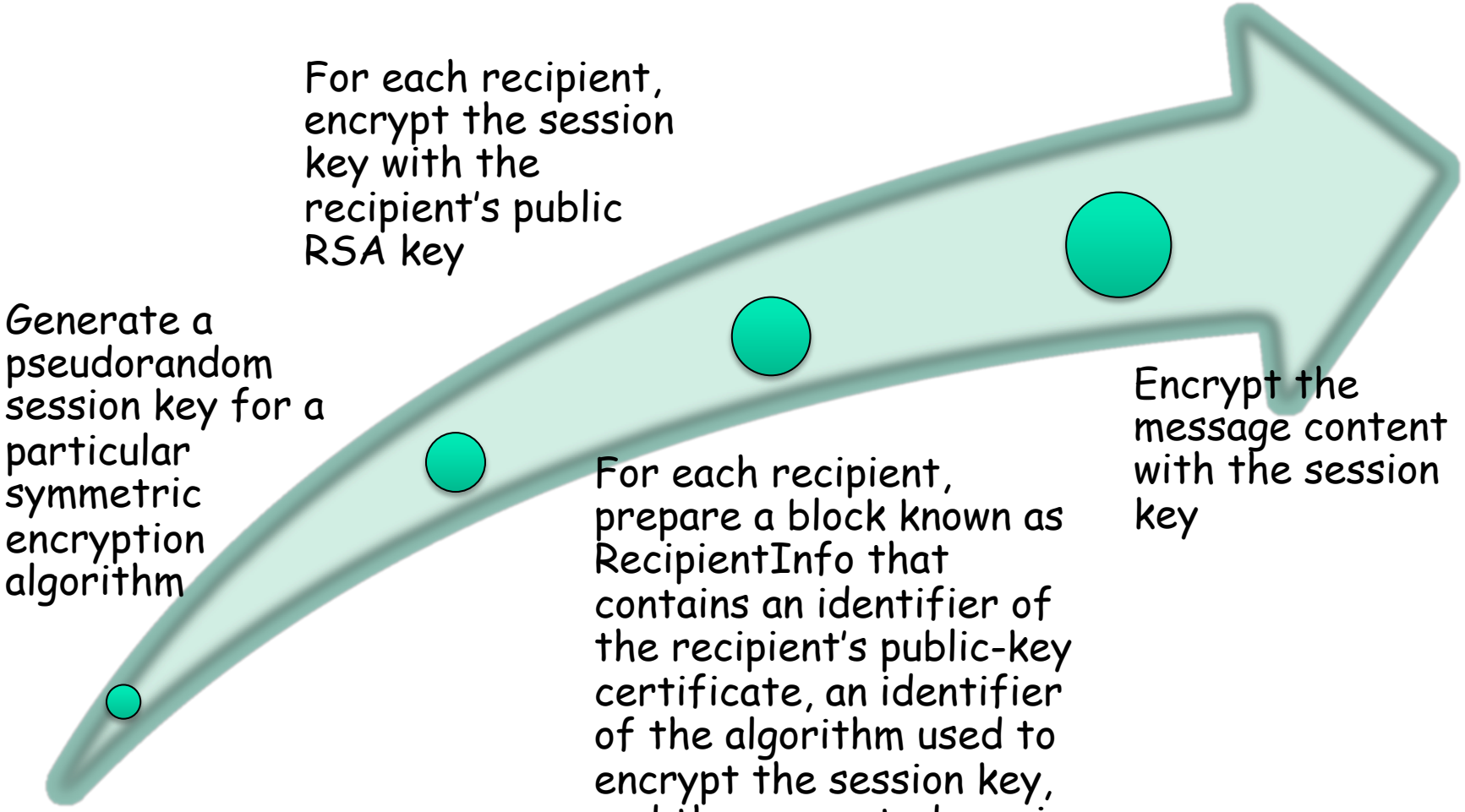
- Dependendo do que se pretende proteger, as mensagens são submetidas ao processamento criptográfico, tendo em conta as chaves e parâmetros dessas operações

- Os conteúdos resultantes desse procassamento criptográfico resulta num objeto normalizado PKCS7 que será incluído como um Content-Type verificável.

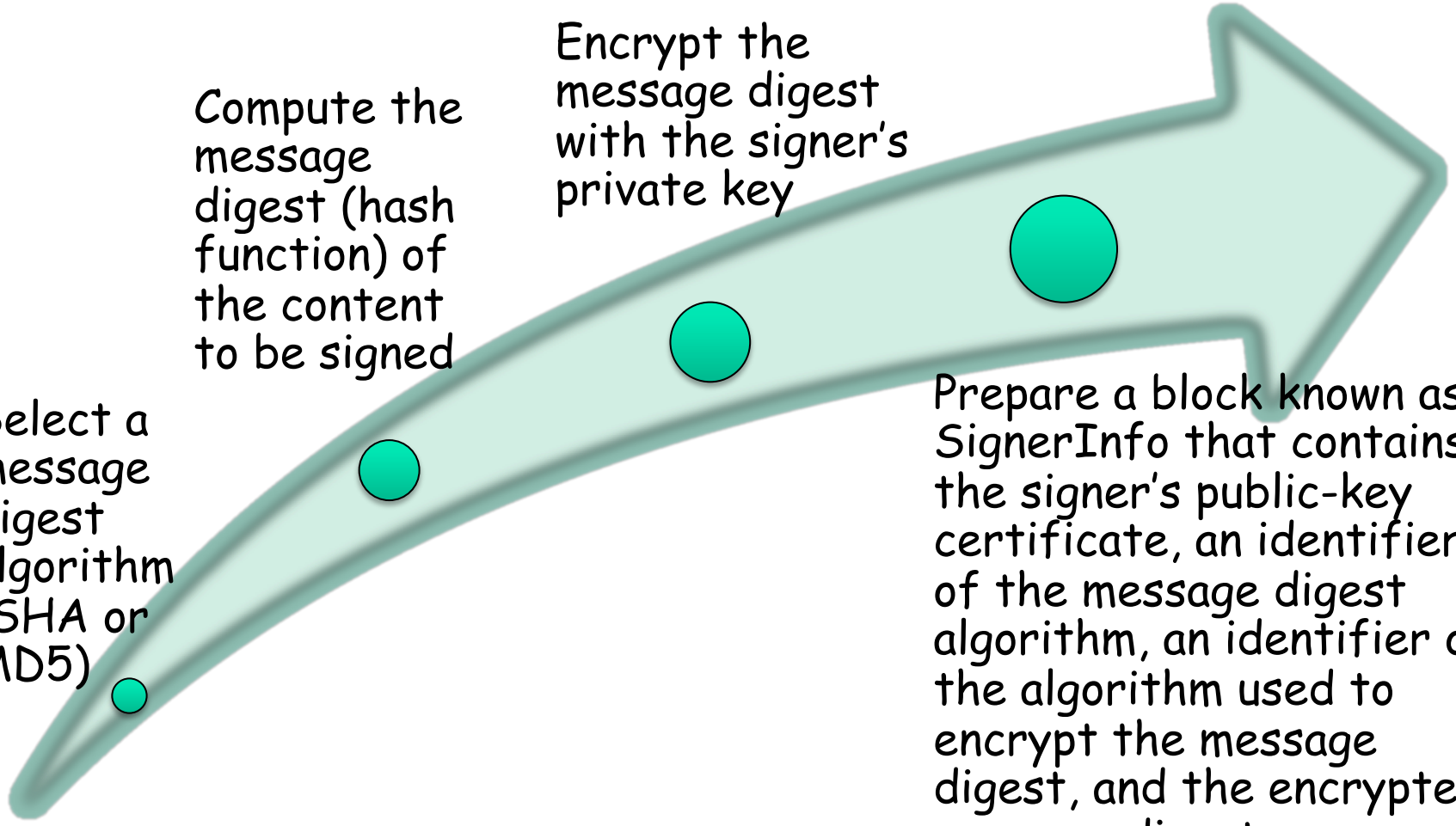
- A representação de acordo com as regras PKCS tem toda a informação (identificadores dos algoritmos criptográficos, tamanhos de chaves, parâmetros, etc)

- Em todos os casos, a messagem a ser enviada será sempre convertida a formatos canónicos.

# Processamento de "Enveloped Data"



# Processamento de "Signed Data"



# Exemplo de mensagem S/MIME

Text Visualizer

Expression: eml

Value:

```
Return-Path: test@testdomain.com
Received: from Main ([127.0.0.1])
        by MAIN
        ; Tue, 10 May 2011 14:09:50 +0200
Message-ID: <88BCB7277A61438C99DB53EFC56BFF15@Main>
From: <test@testdomain.com>
To: <alice2@testdomain.com>
References: <83d6199b-7d0b-4293-9afc-1e26ebc1321e@mail.dll>
In-Reply-To: <83d6199b-7d0b-4293-9afc-1e26ebc1321e@mail.dll>
Subject: Re:
Date: Tue, 10 May 2011 14:07:02 +0200
MIME-Version: 1.0
Content-Type: application/x-pkcs7-mime;
        format=flowed;
        smime-type=enveloped-data;
        name="smime.p7m";
        reply-type=original
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
        filename="smime.p7m"
X-Priority: 3
X-MSMail-Priority: Normal
Importance: Normal
X-Mailer: Microsoft Windows Live Mail 15.4.3508.1109
X-MimeOLE: Produced By Microsoft MimeOLE V15.4.3508.1109

MIAGCSqGSIb3DQEHA6CAMIACAQAxgeQwgeECAQAwSjA2MSQwIgYJKoZIhvcNAQkBFhVhbGljZTJA
dGVzdGRvbWFpbi5jb20xDjAMBgNVBAMTBUFsaWNlAhA6/GZZsf7+skGyLbPXGyDNMA0GCSqGSIb3
DQEBAQUABIGAN6t8J1FZ4k6qyw4WLwqaugArgnGxAzJ3EySSBT6/qzNTKKbPCn3Tvc6GXJx3Vh3f
```

# Outros exemplos… (S/MIME)

S/MIME signed messages >>>

- http://www.jensign.com/JavaScience/verify/smimenote.html
- http://www.jensign.com/JavaScience/verify/advanced/message.txt

S/MIME /compatibilidade MIME

- http://www.jensign.com/JavaScience/verify/advanced/content.txt

# Funções no MUA com S/MIME

- Processamento dos objetos criptográficos associados à interpretação e processamento das multi-partes com S/MIME Content-Types vistos anteriormente
- Uso das respetivas implementações dos algoritmos normalizados (biblioteca criptográfica com funções PKCS e geração de chaves OTK)
- Gestão de certificados e chaves

# Funções de gestão das chaves num MUA S/MIME

**Geração de Chaves OTK**

**The user of some related administrative utility must be capable of generating separate Diffie-Hellman and DSS key pairs and should be capable of generating RSA key pairs**

**A user agent should generate RSA key pairs with a length in the range of 768 to 1024 bits and must not generate a length of less than 512 bits**

**Registo da Chave Pública / "Enrolment do Certificado de Chave Pública)**

**A user's public key must be registered with a certification authority in order to receive an X.509 public-key certificate**

**Armazenamento e Gestão Local de Certificados (como nos browsers)**

**A user requires access to a local list of certificates in order to verify incoming signatures and to encrypt outgoing messages**

# Classes de certificados para S/MIME

- Envolve um esquema (tentativo) de normalização de CLASSES DE CERTIFICADOS definindo níveis de responsabilidades das CAs em relação à emissão de certificados com diferentes níveis de responsabilização.
  - Condições de usufruto do certificado, para efeitos da validação do mesmo tendo em vista o processamento das mensagens S/MIME

# Classes de certificação (Verisign)

| | Summary of Confirmation of Identity | IA Private Key Protection | Certificate Applicant and Subscriber Private Key Protection | Applications implemented or contemplated by Users |
|---|---|---|---|---|
| **Class 1** | Automated unambiguous name and E-mail address search | PCA: trustworthy hardware; CA: trust-worthy software or trustworthy hardware | Encryption software (PIN protected) recommended but not required | Web-browsing and certain e-mail usage |
| **Class 2** | Same as Class 1, plus automated enrollment information check plus automated address check | PCA and CA: trustworthy hardware | Encryption software (PIN protected) required | Individual and intra- and inter-company E-mail, online subscriptions, password replacement, and software validation |
| **Class 3** | Same as Class 1, plus personal presence and ID documents plus Class 2 automated ID check for individuals; business records (or filings) for organizations | PCA and CA: trustworthy hardware | Encryption software (PIN protected) required; hardware token recommended but not required | E-banking, corp. database access, personal banking, membership-based online services, content integrity services, e-commerce server, software validation; authentication of LRAAs; and strong encryption for certain servers |

IA Issuing Authority
CA Certification Authority
PCA VeriSign public primary certification authority
PIN Personal Identification Number
LRAA Local Registration Authority Administrator

# Outros serviços de segurança futuros

- Novos serviços têm sido e estão a ser propostos na evolução gradual da normalização S/MIME. Exemplos de alguns (já suportados em soluções SW já existentes):

### Signed receipt

- Returning a signed receipt provides proof of delivery to the originator of a message and allows the originator to demonstrate to a third party that the recipient received the message

### Security labels

- A set of security information regarding the sensitivity of the content that is protected by S/MIME encapsulation

### Secure mailing lists

- An S/MIME Mail List Agent (MLA) can take a single incoming message, perform the recipient-specific encryption for each recipient, and forward the message

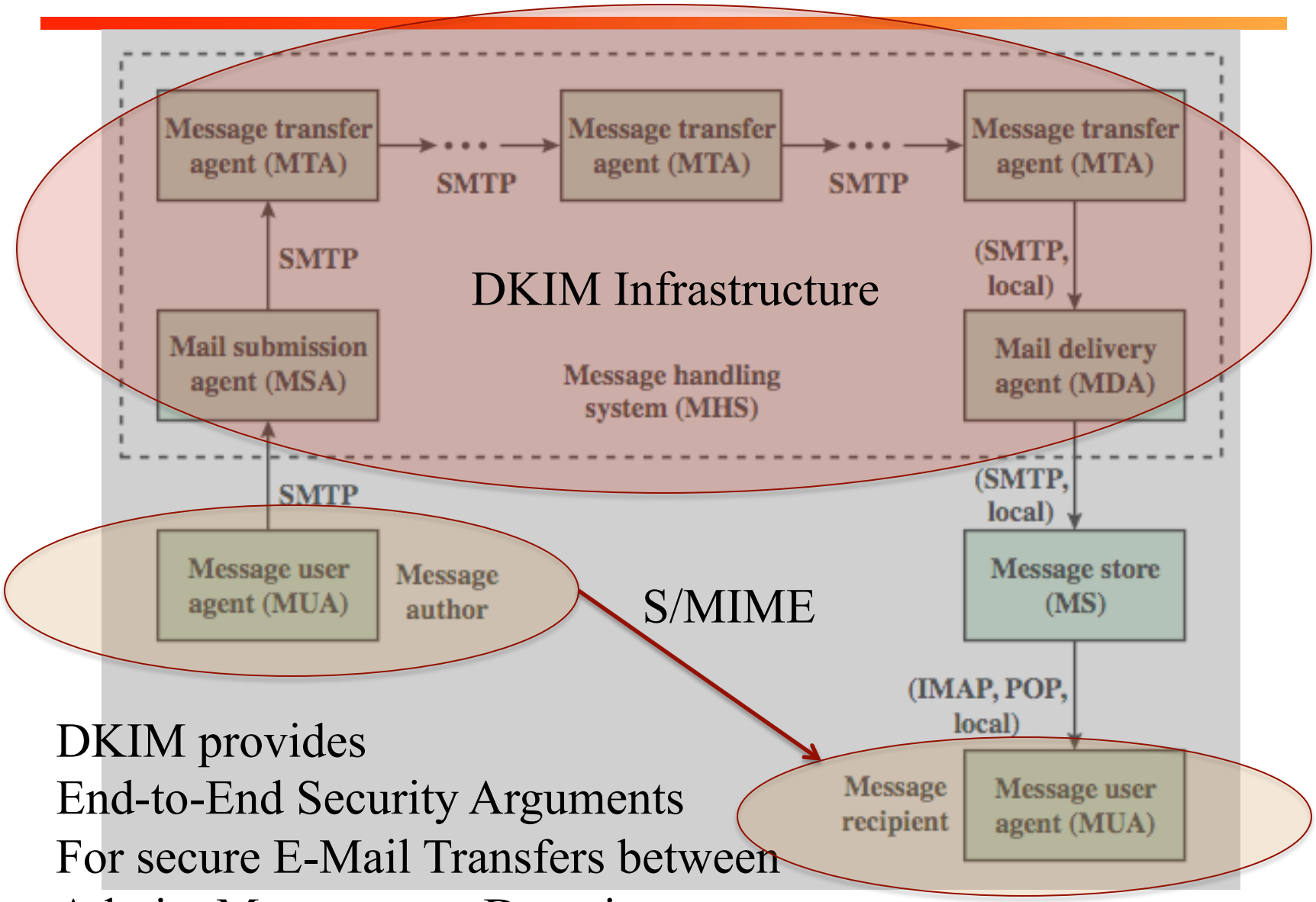
# Tópicos

- Formato de normalização MIME
- Normalização S/MIME
- DKIM

# Motivação DKIM

- Quando se usa PGP ou S/MIME os conteudos das mensagens são protegidos, mas a segurança extremo-a-extremo tb pode ser afectada ou trazer novos problemas, devido a fluxos "blindados" de mensagens que cruzam diferentes domínios de administração
  - Que problemas ? Discussão
- PGP e S/MIME: segurança apenas depende dos endpoints (principais desigandos pelos respetivos endereços Email RFC 822) e depende da correção (commitment) dos utilizadores finais
  - Utilizadores podem defender-se de SPAM ? Ou Induzir SPAM ?
  - Po outro lado a grande maioria das mensagens Email que pssam na Internet não usam ainda hoje S/MIME ou PGP
  - E em muitos casos que usam S/MIME apenas a função de assinatura da parte protegida é usada – logo a informação dos cabeçalhos RFC 5322 pode ser comprometida
- Falta de controlo de outras dimensões do pronlema: *message-hijacking on the fly, no control of message relaying paths, spamming vulnerability, ….*

# Internet Mail Architecture

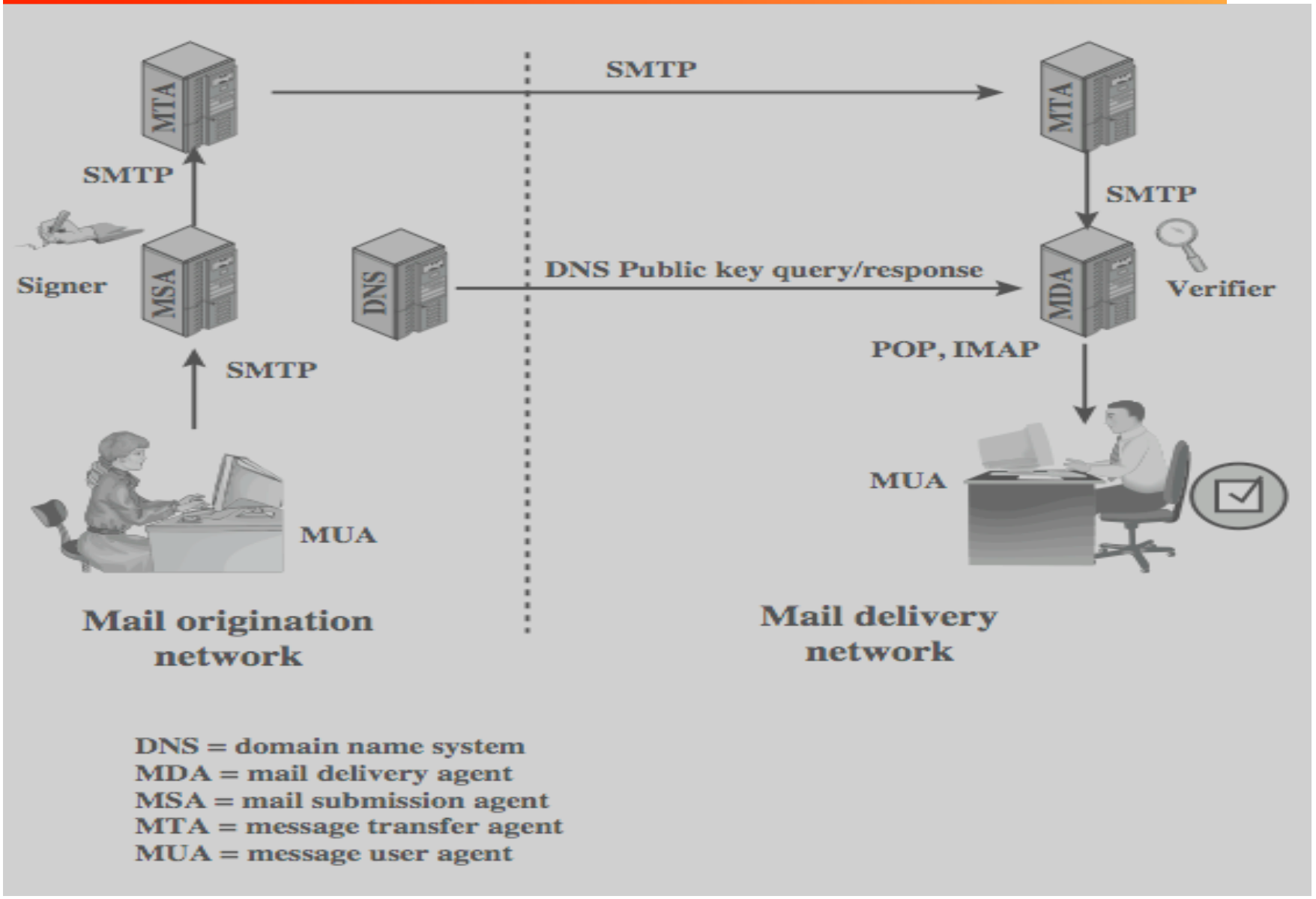



DKIM provides
End-to-End Security Arguments
For secure E-Mail Transfers between
Admin. Management Domains

# DKIM - Domain Keys Identified Mail

- DKIM :especificação para assinatura de mensagens trocadas enter domínios de exchange / store-amd-forward (entre instâncais de encainhamento SMTP)
  - Não implementado nos MUAs (apenas ao nível de MESSAGE RELAYING): transparenet para os utilizadores finais
- Assinatura do domínio inputa um níveld e responsabiliade, entre domíniso cooperantes.
- Recipientes / agentes podem verificar se a assinatura que cobre a mensagem corersponde a um domínio confiável
- Allowing "Good" senders to prove that they did send a particular message and to prevent "forgers" from masquerading "good senders"
- Proposed Internet Standard RFC 4871
- Has been widely adopted in may softwaree products

# Estratégia e infraestrutura DKIM

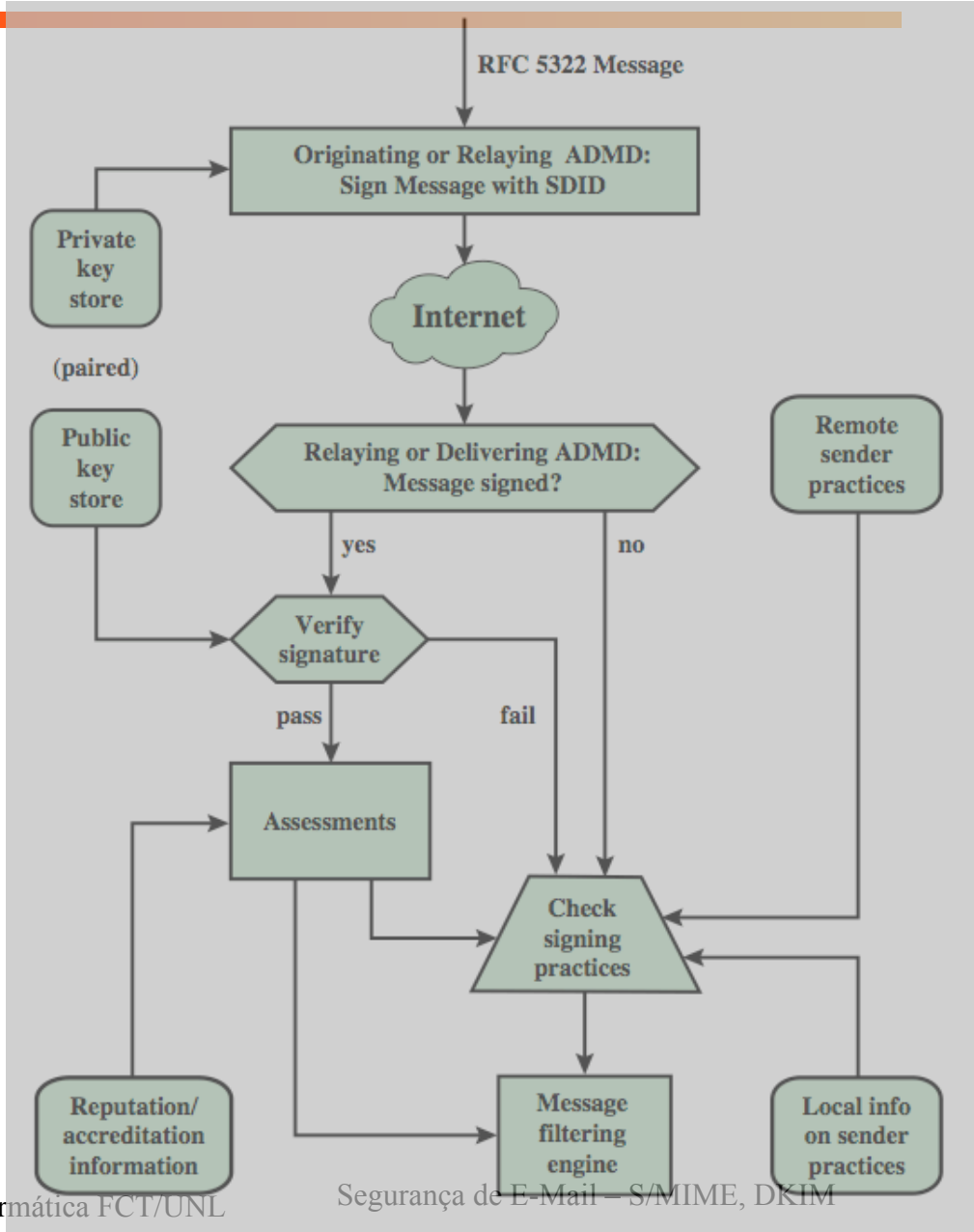


# RFC 4686 principles (DKIM)

- Mensagens submetidas por end-users aos STAs e MSAs nas múltiplas localizações INTERNET (ex., ISPs), via um "E-Mail provider service" num ADMD – *Administrative Management Domain*)
- O provedor assina as mensagens (cobrindo o formato RFC 5322) nos campos sensíveis e considerados críticos - RFC 5322 header fields
  - Impedindo falsificações/alterações entre o domínio ADM origem e destino
  - Assinaturas processadas por um módulo de autorização que assina as mensagens em nome do domínio (ADMD)
  - Chaves privadas do domínio: mantidas em segurança numa Key-Store gerida e apenas acessível ao ADMD
  - Verificação de assinaturas nos MTAs ou MSAs (podendo também ser possúvel num MUA final)
- Assinaturas inseridas nas mensagens RFC 5322 nua entrada identificada pela key: Dkim-Signature:<signature>

# MSA: *Functional Processing Flow*

ADMD:
Administrative
Management
Domain



# DKIM - Domain Keys Identified Mail

- Desta forma, pode estabelecer-se um controlo no encaminhamento de mensagens, sobre:
  - "*Good" senders vs. "Bad Senders*"
  - Políticas: *white-lists vs. black-lists*
    - Apenas exige controlo sobre a confiabilidade dos certificados de chave pública de domínio (X509v3 *Certificates*), respetiva validação usual e gestão adicional de *whitelists* ou *blacklists* ao nvel do contexto de gestão/administração do serviço de Email Relay de cada domínio

- DKIM inicalmente proposto no Internet Standard RFC 4871
- Tem sido rapidamente incorporado na gestão/configuração dos sistemas e soluções (servidores Email / EMail Relaying – eSMTP )

- **Aspetos em aberto:**
  - Como automatizar estes processos sem intervenção (ou reduzindo a intervenção) humana ?
  - Outros aspetos: integração com sistemas IPS, IDS, IRS

# Formato das assinaturas DKIM

O campo de assinatura DKIM tem várias partes separadas por ";" com possíveis sub-atributos que usam o separador ":"

- Ex.,

  v=DKIM1;

  a=rsa-sha256;

  c=relaxed/relaxed;

  d=gmail.com;

  s=gamma;

  h=domainkey-signature:mime-version:received:date:message-id:subject:from:to:content-type:content-transfer-encoding;

  bh=XXXXXXX….XXXX;

  // bh: hashing da representação canónica da parte do de mensagens, possibilitando a verificação adequada da assinatura

  b=YYYYYYYY….YYYYY

  // b: Assinatura com codificação base64

# Exemplo:

http://dkim.org/specs/rfc4871-dkimbase.html

# Ameaças EMail

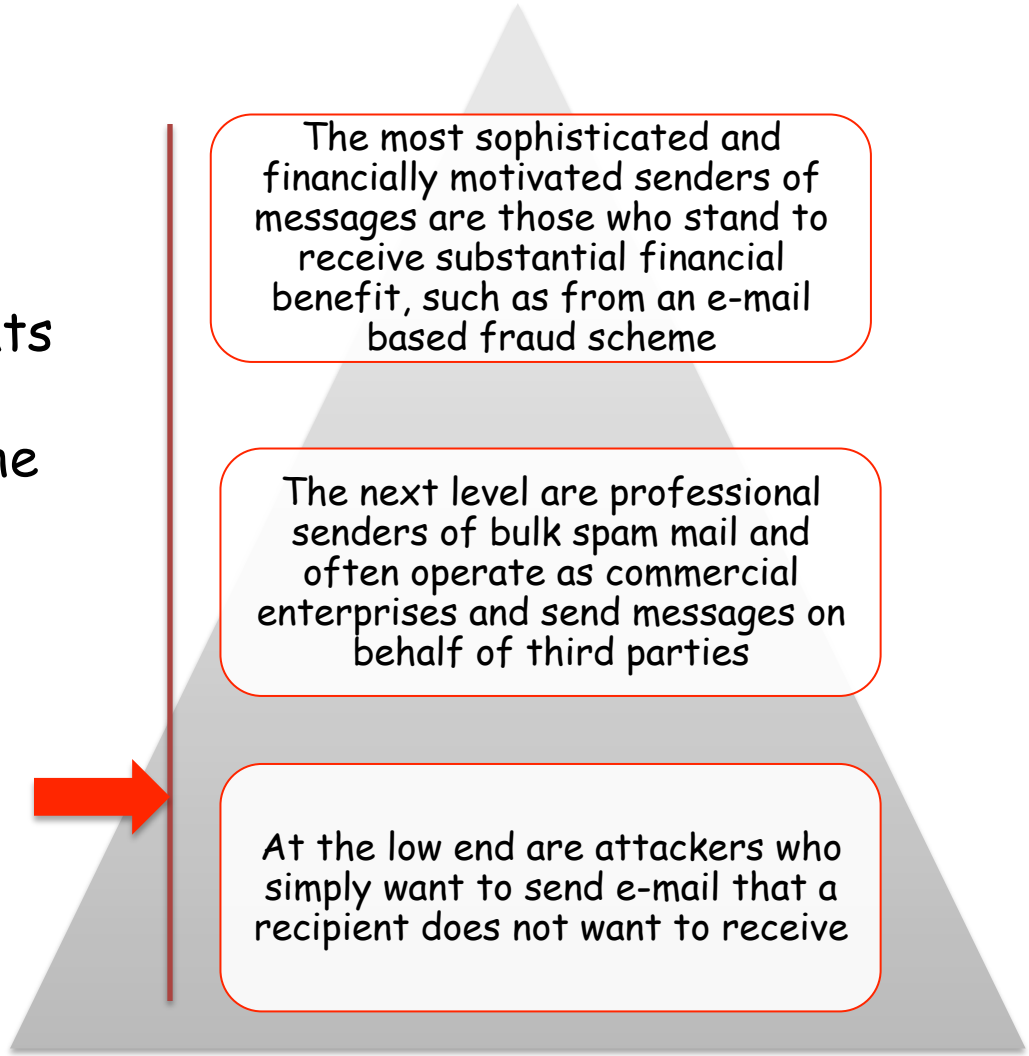
- RFC 4684- *Analysis of Threats Motivating DomainKeys Identified Mail*
  - Descreve a motivação da normalização DKIM e os problemas am aberto
  - *Range: low end, spammers, fraudsters*
  - *Capabilities: where/who/what submitted, signed, volume, routing naming, etc*
  - *Outside located attackers*

# E-mail Threats

- RFC 4684 *(Analysis of Threats Motivating DomainKeys Identified Mail)*

  Describes the threats being addressed by DKIM in terms of the characteristics, capabilities, and location of potential attackers

- Characterized on three levels of threat:



# Tópicos cobertos

- Formato de normalização MIME
- Normalização S/MIME
- DKIM

# Leituras

# Bibliografia

- W. Stallings, Network Security Essentials
  - Cap. 7 – Electronic Mail Security
    - 7.2 S/MIME
    - 7.3 DKIM