Practical Cryptography

Handout 8 – Cryptography Protocols

Kasun de Zoysa
kasun@ucsc.cmb.ac.lk

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

Internet Cryptographic Protocols

- **IPSec :** Packet-Level Encryption, RFC2401
- **DNSSEC :** Domain Name System, RFC2065
- **PCT :** TCP/IP-level Encryption
- **S-HTTP :** Web Browsing, RFC2660
- **SSL :** TCP/IP-level Encryption, Netscape
- **TLS :** TCP/IP-level Encryption, RFC2246
- **SET :** Electronic Funds Transactions
- **Cybercash :** Electronic Funds Transactions, RFC1898
- **PGP :** E-Mail, RFC2015
- **S/MIME :** E-Mail, RFC2311,RFC2634
- **SSH :** Remote Login

Secure Socket Layer History

- SSL 1.0 Netscape 1994
- S-HTTP (web only)
- SSL 2.0 Netscape (buggy)
- PCT Microsoft (loser) 1996
- SSL 3.0 Netscape
- TLS 1.0 IETF 1999
- TLS 1.2 now dominant

TLS: Transport Layer Security

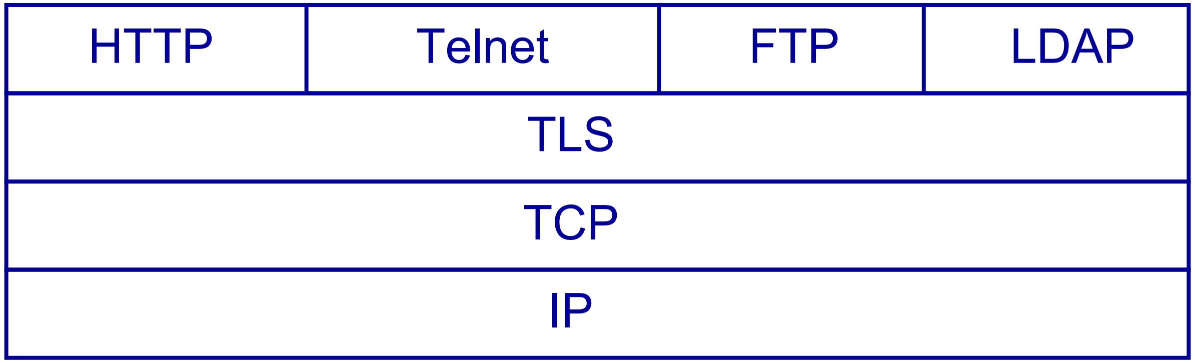
- *formerly known as*
  SSL: Secure Sockets Layer
- Addresses issues of privacy, integrity and authentication
  - What is it?
  - How does it address the issues?
  - How is it used

TLS

- “TLS, more commonly known as SSL”
- RFC2246 : TLS Protocol Version 1.0 1/99
- RFC2487 : SMTP over TLS
- RFC2712 : Adding Kerberos to TLS
- RFC2716 : PPP TLS
- RFC2817 : Upgrading to TLS within HTTP/1.1
- RFC2818 : HTTP over TLS
- RFC2830 : TLS for Lightweight Directory Access Protocol (LDAP)

What is TLS?

- Protocol layer
- Requires reliable transport layer (e.g. TCP)
- Supports any application protocols



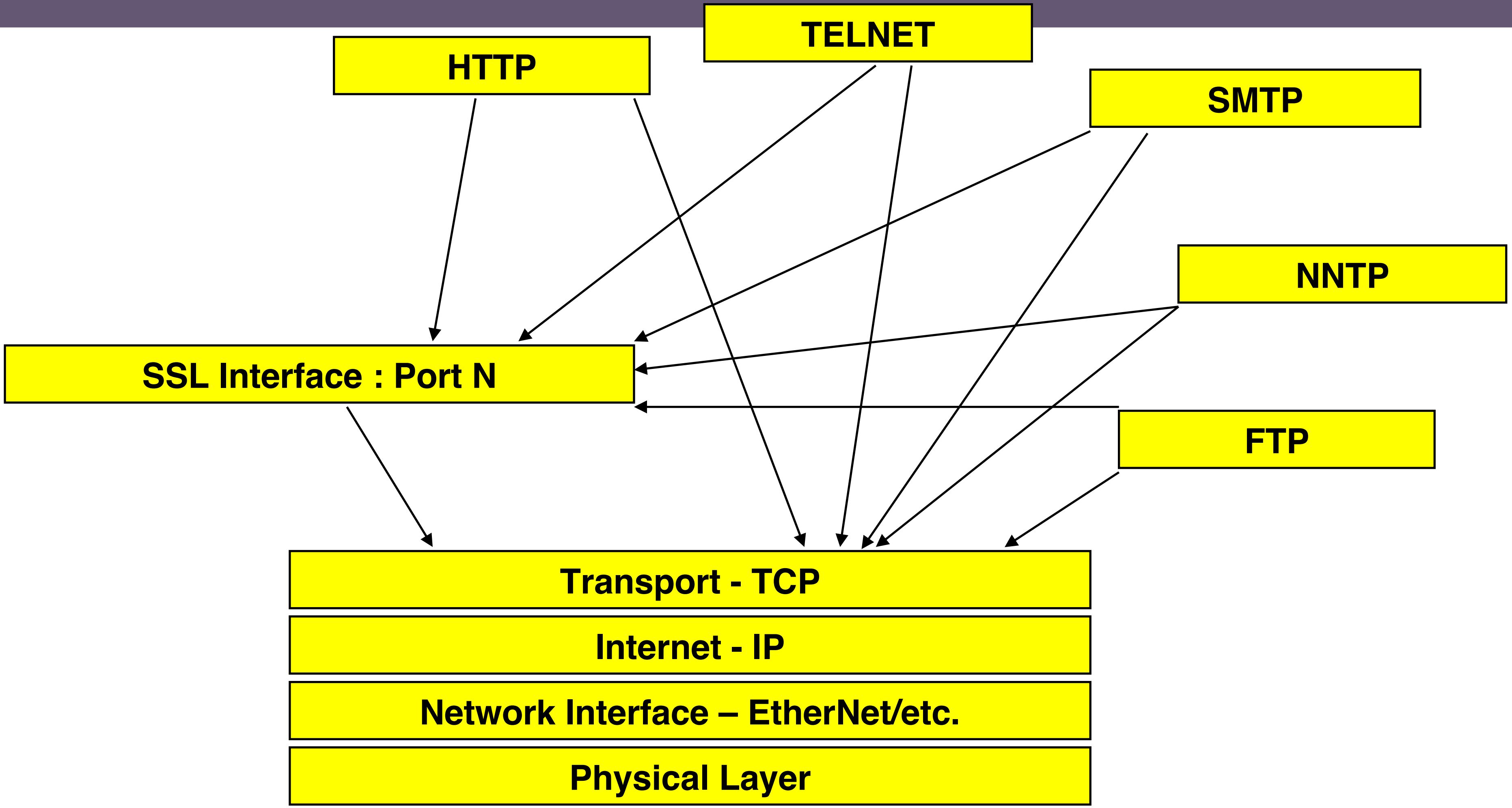
HTTP Telnet FTP LDAP

TLS

TCP

IP

Protocol Stack

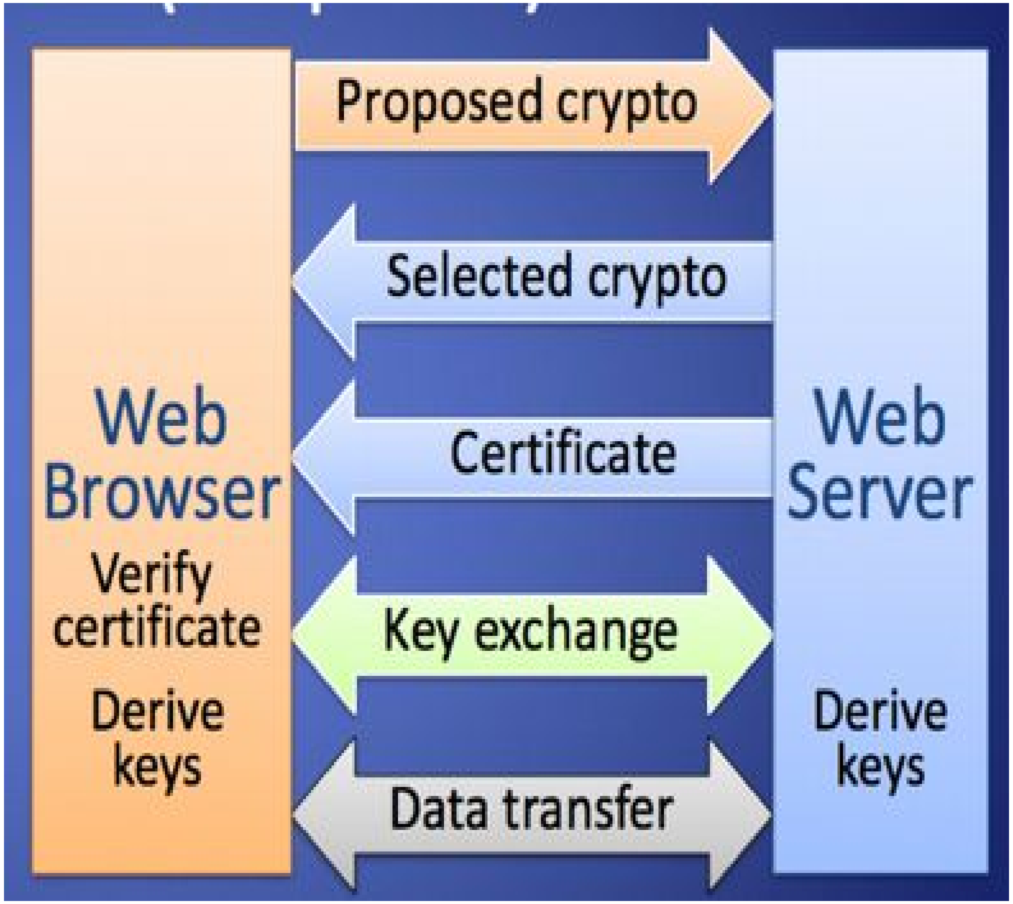


TLS: Overview

- Establish a session
  - Agree on algorithms
  - Share secrets
  - Perform authentication
- Transfer application data
  - Ensure privacy and integrity

TLS Overview

- Browser sends supported crypto algorithms
- Server picks strongest algorithms it supports
- Server sends certificate (chain)
- Client verifies certificate (chain)
- Client and server agree on secret value R by exchanging messages
- Secret value R is used to derive keys for symmetric encryption and hash-based authentication of subsequent data transfer

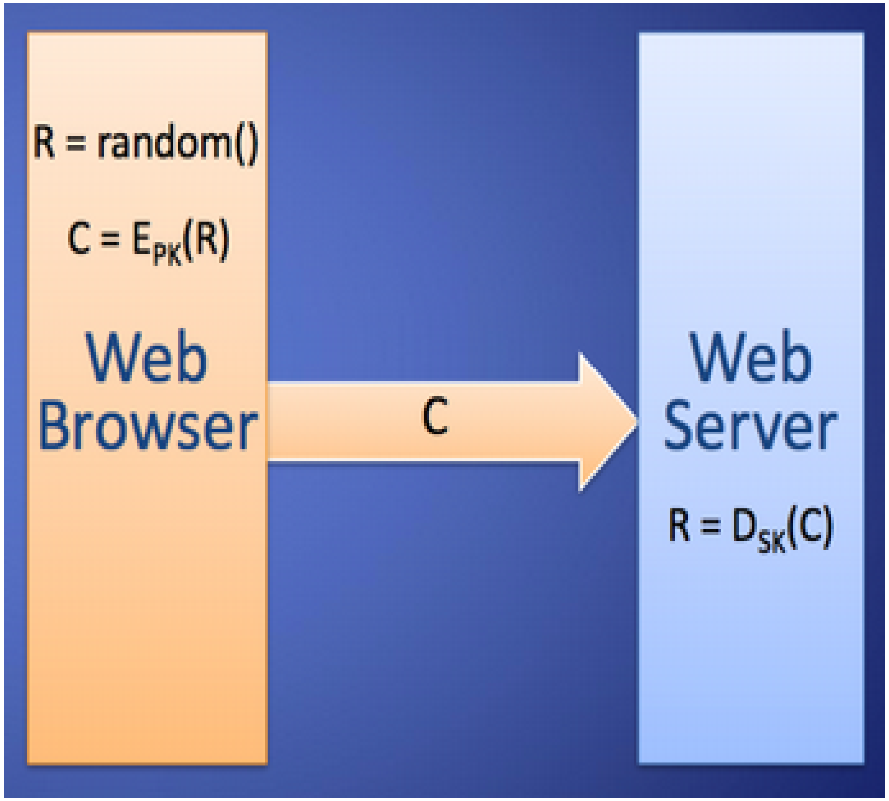


TLS:Key Exchange

- Need secure method to exchange secret key
- Use public key encryption for this
  - “key pair” is used - either one can encrypt and then the other can decrypt
  - slower than conventional cryptography
  - share one key, keep the other private
- Choices are RSA or Diffie-Hellman

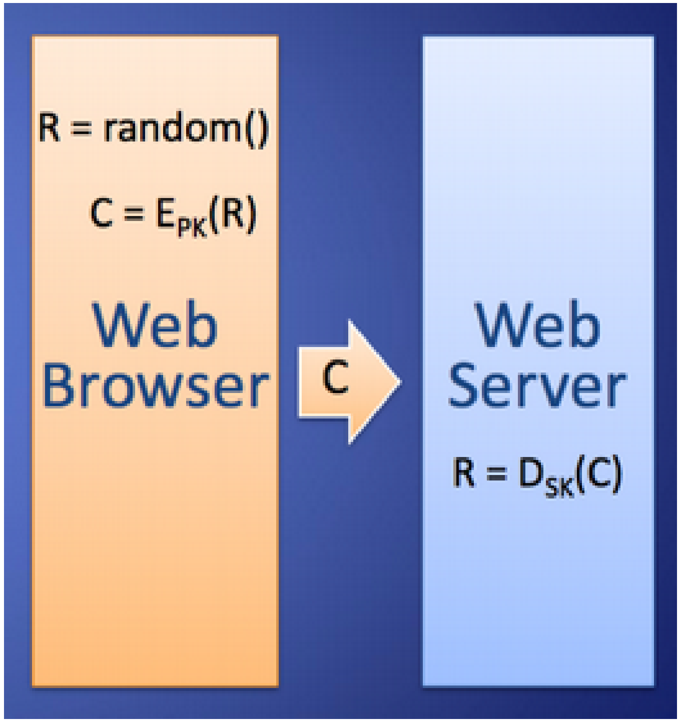
Basic Key Exchange

- Called RSA key exchange for historical reasons
- Client generates random secret value R
- Client encrypts R with public key, PK, of server C = EPK(R)
- Client sends C to server
- Server decrypts C with private key, SK, of server R = DSK(C)

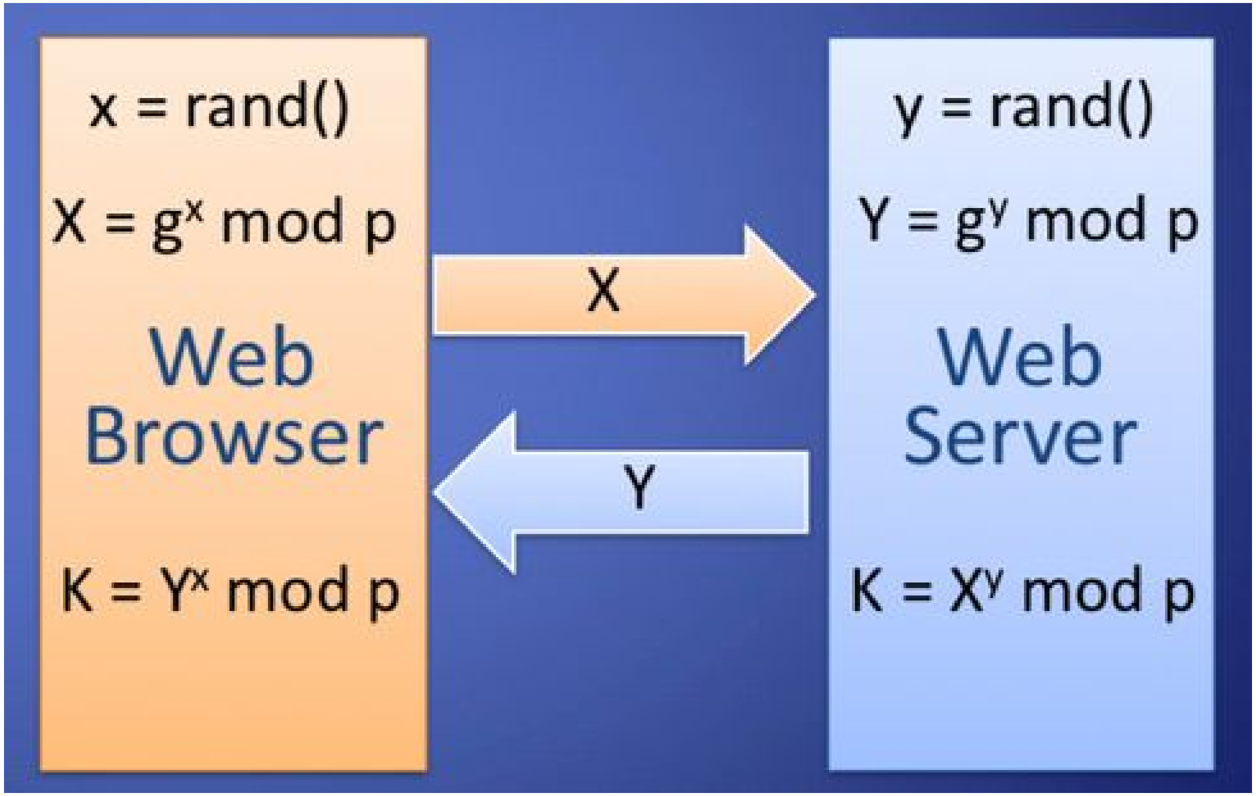


Forward Secrecy

- Compromise of public-key encryption private keys does not break confidentiality of past messages
- TLS with basic key exchange does not provide forward secrecy
- Attacker eavesdrop and stores communication
- If server’s private key is compromised, attacker finds secret value R in key exchange and derives encryption keys

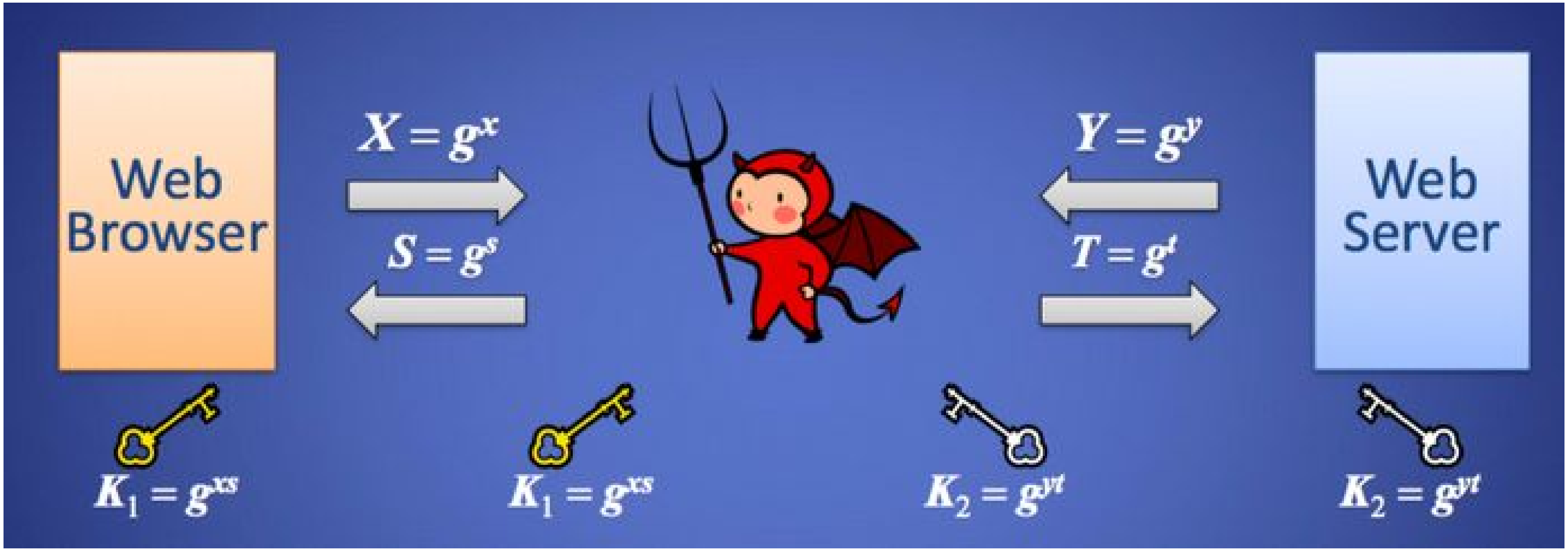


Diffie Hellman Key Exchange



Achieves forward secrecy

Attacker in the Middle



Solution:
Browser and server send signed X and Y respectively
Requires each to know the public key of the other

TLS: Privacy

- Encrypt message so it cannot be read
- Use conventional cryptography with shared key
  - DES, 3DES, AES
  - RC2, RC4
  - IDEA

A

B

Message

$%&#!@

Message

TLS Encrypts

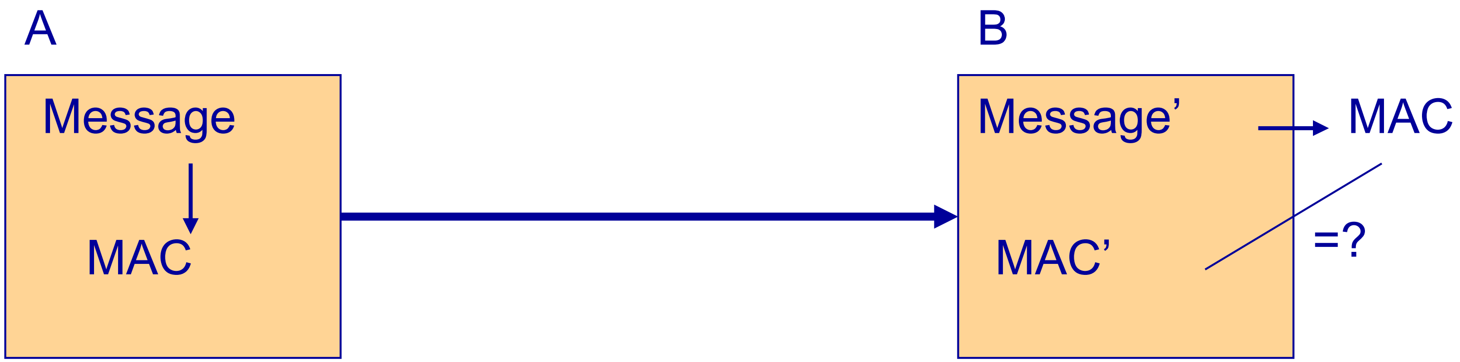
- ALL Browser-Server and Server-Browser except which-browser is talking to which-server
- URL of requested document
- Contents of requested document
- Contents of any submitted form fill-outs
- Cookies sent from browser to server
- Cookies sent from server to browser
- Contents of HTTP header
- Javascript communications
- Etc.

TLS: Integrity

- Compute fixed-length Message Authentication Code (MAC)
  - Includes hash of message
  - Includes a shared secret
  - Include sequence number
- Transmit MAC with message

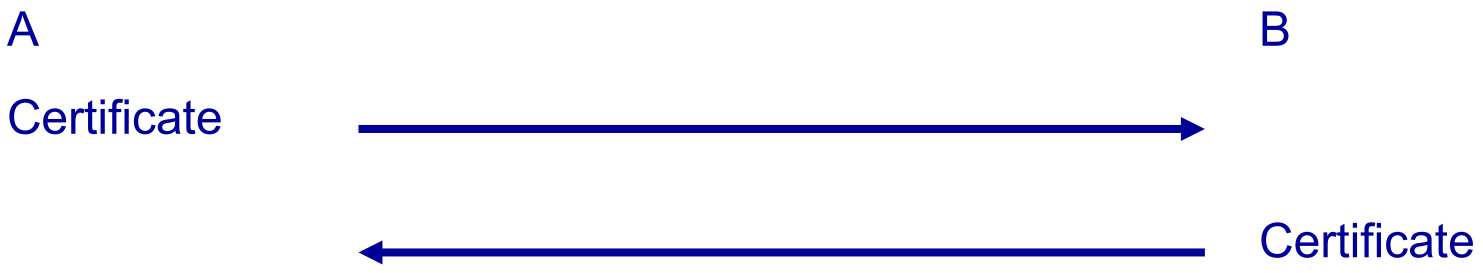
TLS: Integrity

- Receiver creates new MAC
  - should match transmitted MAC
- TLS allows MD5, SHA-1



TLS: Authentication

- Verify identities of participants
- Client authentication is optional
- Certificate is used to associate identity with public key and other attributes



A

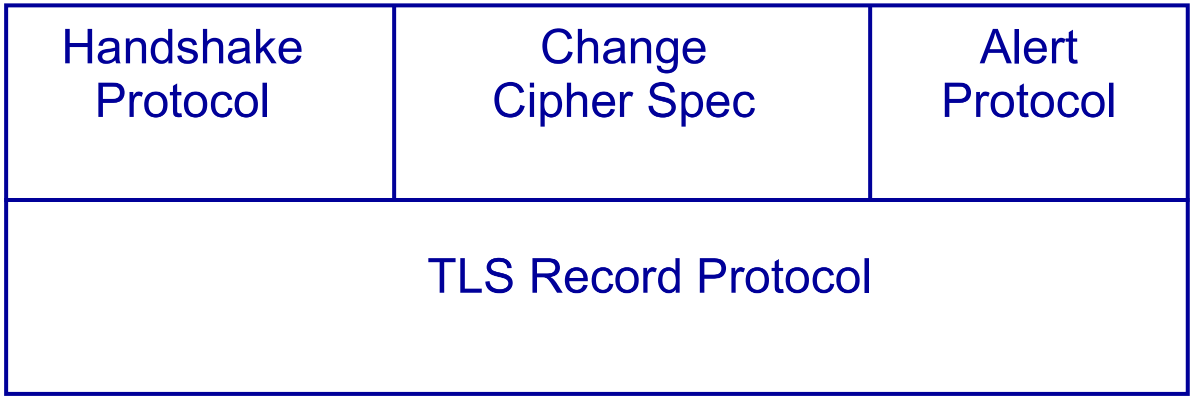
B

Certificate

Certificate

TLS: Architecture

- TLS defines Record Protocol to transfer application and TLS information
- A session is established using a Handshake Protocol



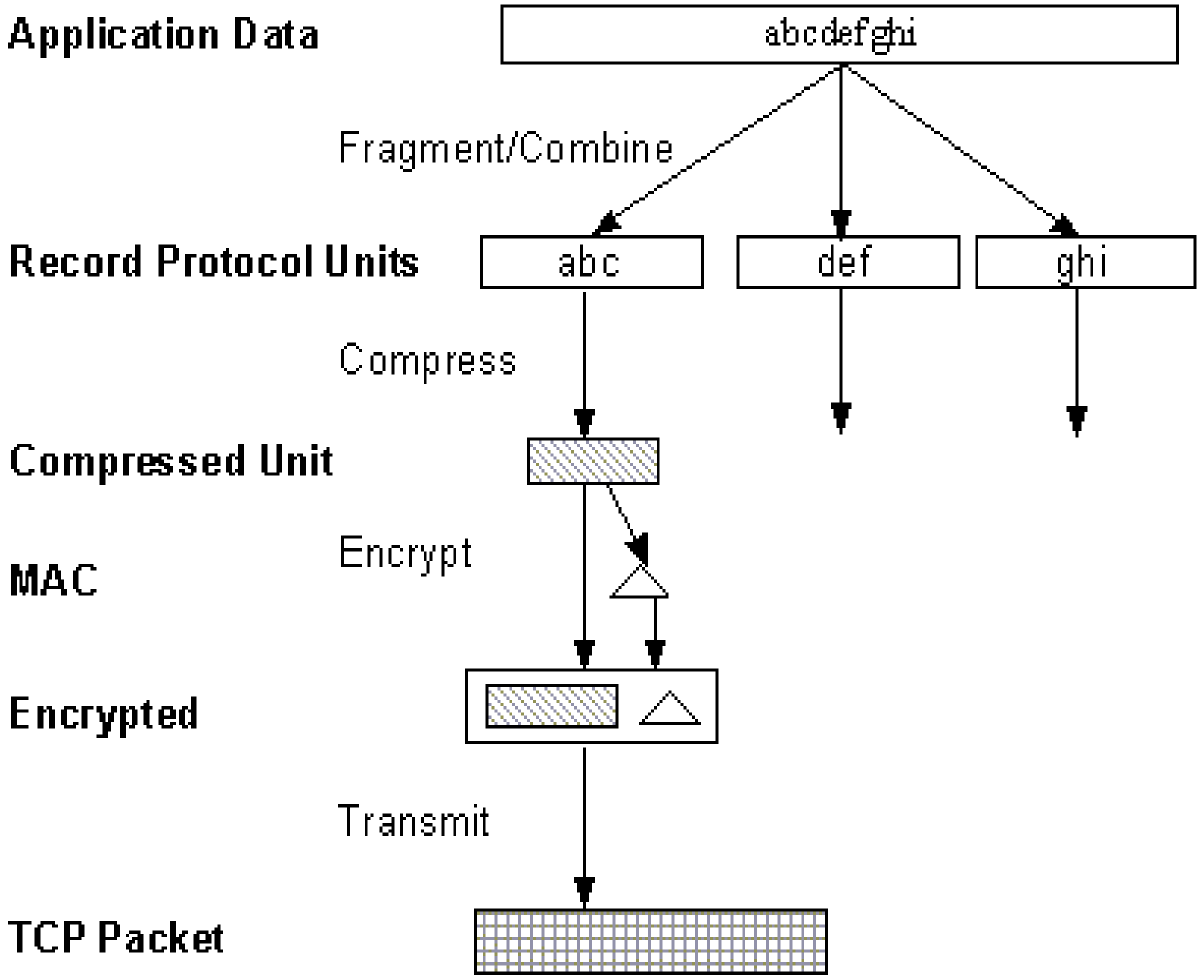
Handshake Protocol

Change Cipher Spec

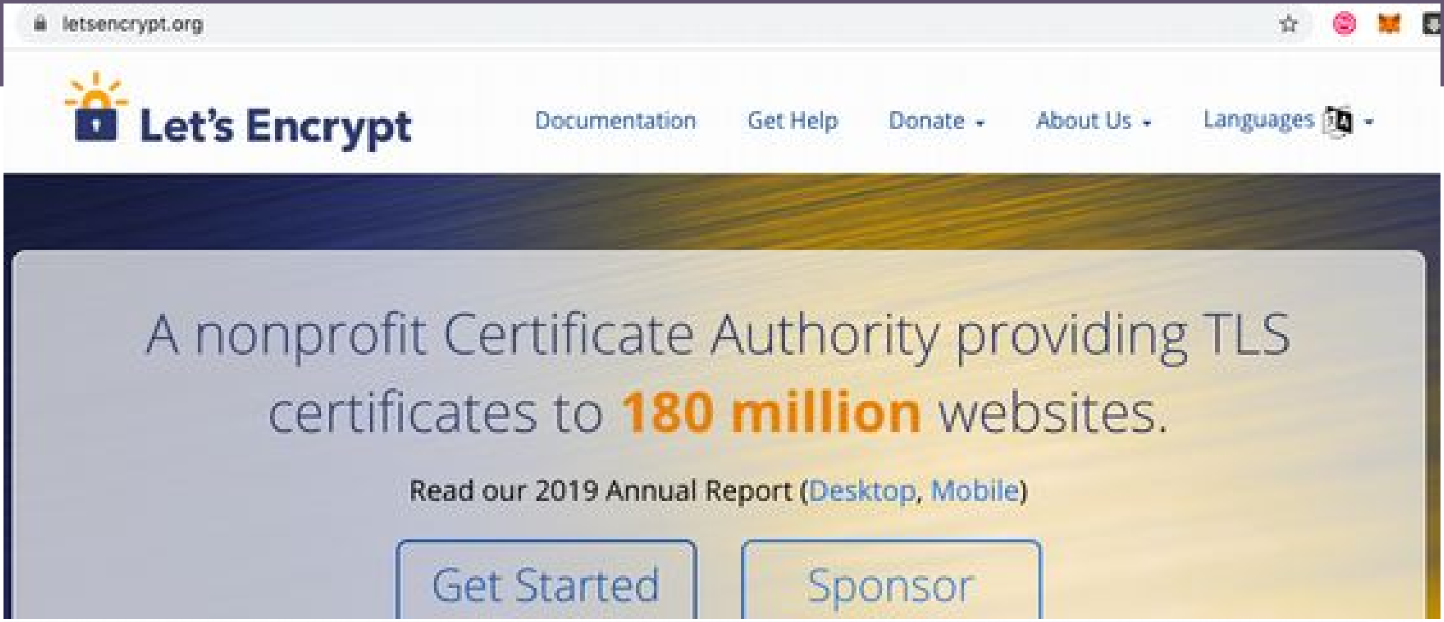
Alert Protocol

TLS Record Protocol

TLS: Record Protocol

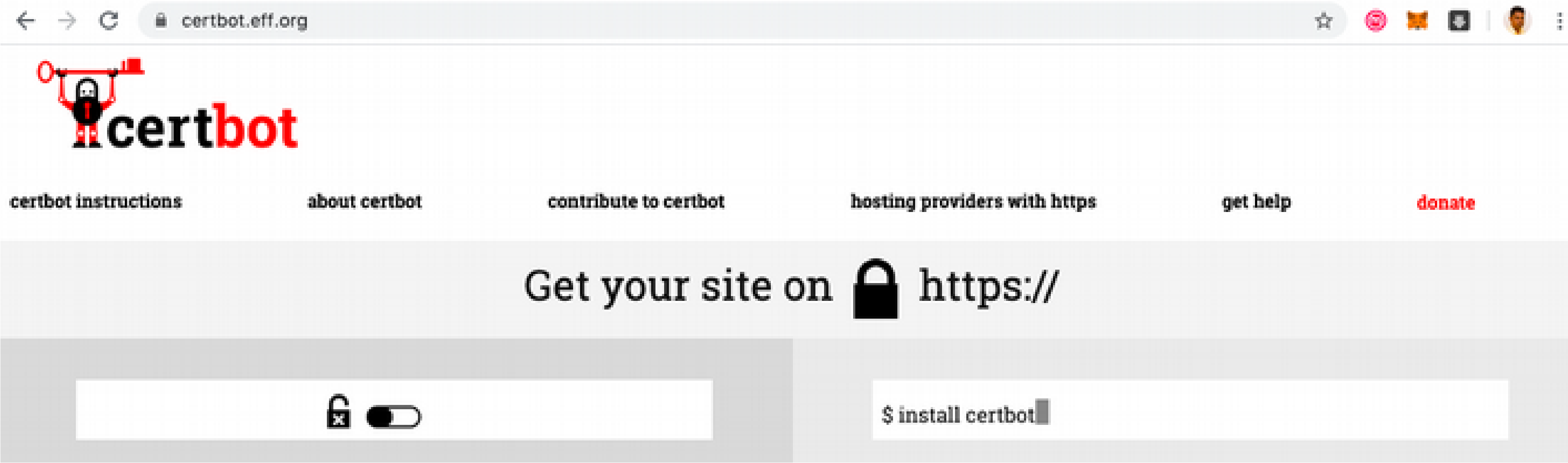


Let’s Encrypt



A nonprofit Certificate Authority providing TLS certificates to **180 million** websites.

Read our 2019 Annual Report (Desktop, Mobile)



Get your site on https://

```
$ install certbot
```

LetsEncrypt – Apache - Ubuntu 18.04

- sudo apt-get install software-properties-common
- sudo add-apt-repository universe
- sudo add-apt-repository ppa:certbot/certbot
- sudo apt install python-certbot-apache
- sudo certbot --apache -d icekubes.center -d www.icekubes.center

LetsEncrypt – Apache - Ubuntu 18.04

- Congratulations! Your certificate and chain have been saved at:
  `/etc/letsencrypt/live/icekubes.center/fullchain.pem`

- Your key file has been saved at:
  `/etc/letsencrypt/live/icekubes.center/privkey.pem`

- openssl x509 -in cert.pem -text

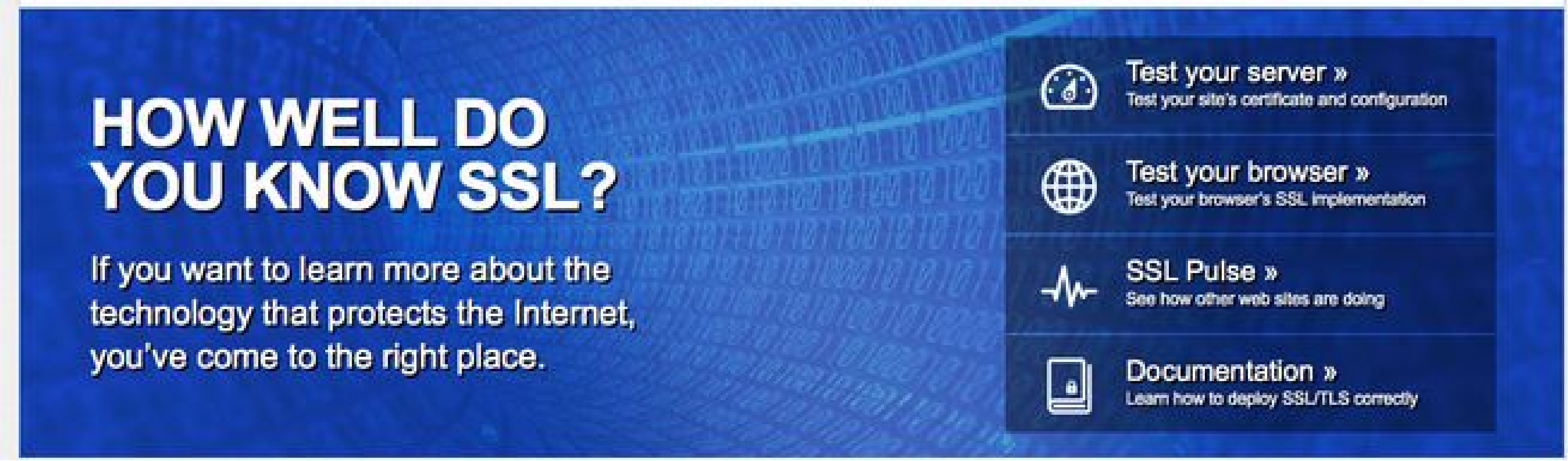
- Your Web root:

  `/var/www/html`

- Your SSL Configuration file:

  `etc/apache2/sites-enabled/000-default-le-ssl.conf`

SSLABS – www.ssllabs.com



HOW WELL DO YOU KNOW SSL?

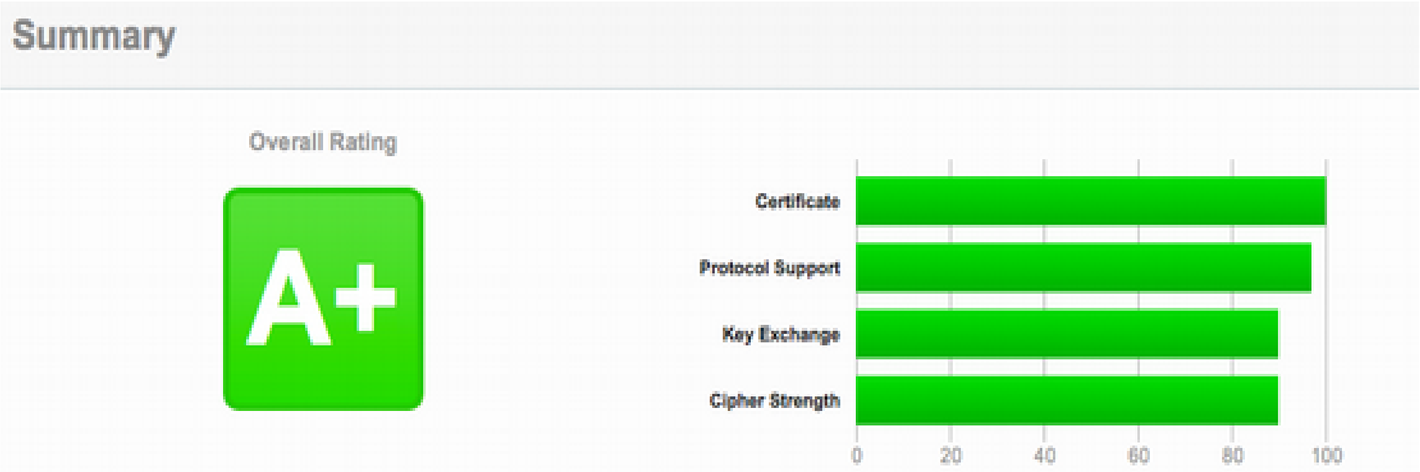
If you want to learn more about the technology that protects the Internet, you've come to the right place.

Test your server »
Test your site's certificate and configuration

Test your browser »
Test your browser's SSL implementation

SSL Pulse »
See how other web sites are doing

Documentation »
Learn how to deploy SSL/TLS correctly



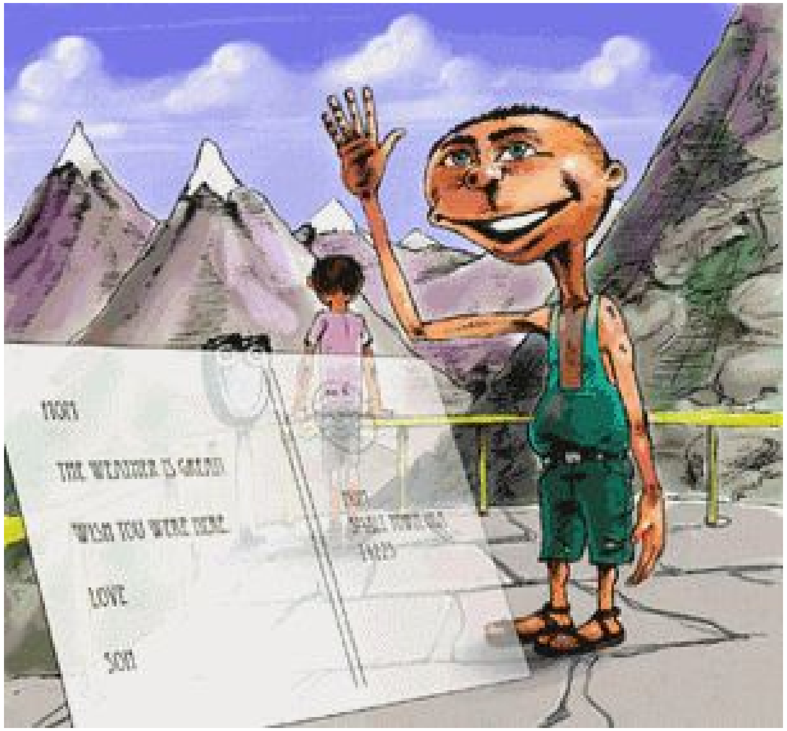
Summary

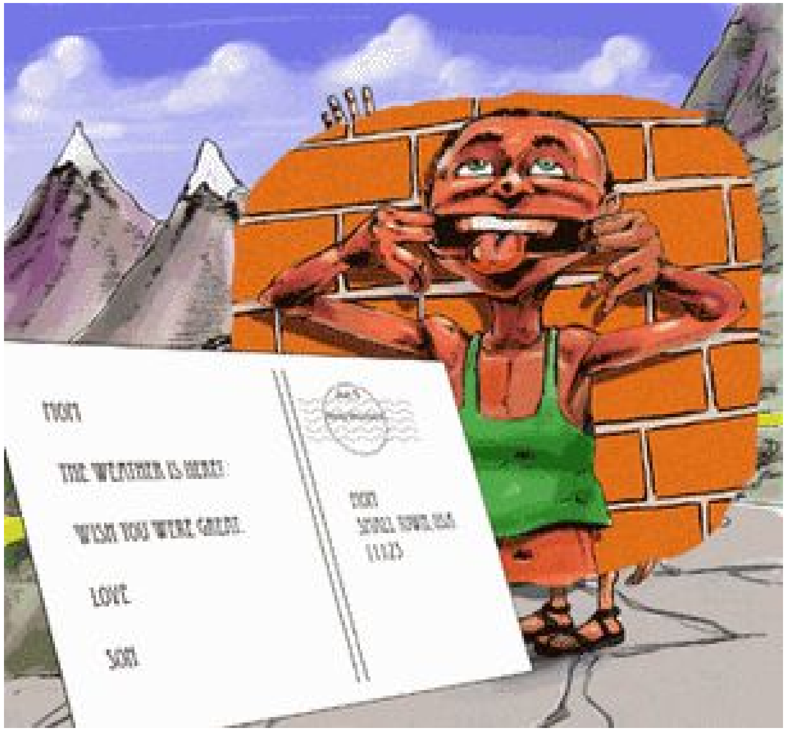
Overall Rating

A+

Email is in the Clear

Email – A Postcard Written in Pencil





http://www.cert.org/homeusers/email_postcard.html

E-mail Security

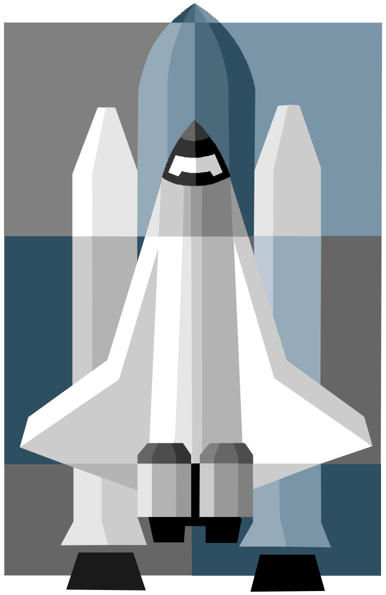
- Pretty Good Privacy (PGP) (www.pgp.com)
  - Philip R. Zimmerman is the creator of PGP.
  - PGP provides a confidentiality and authentication service that can be used for electronic mail and file storage applications.
- S/MIME
  - Secure/Multipurpose Internet Mail Extension
  - S/MIME will probably emerge as the industry standard.
  - PGP for personal e-mail security

Why Is PGP Popular?

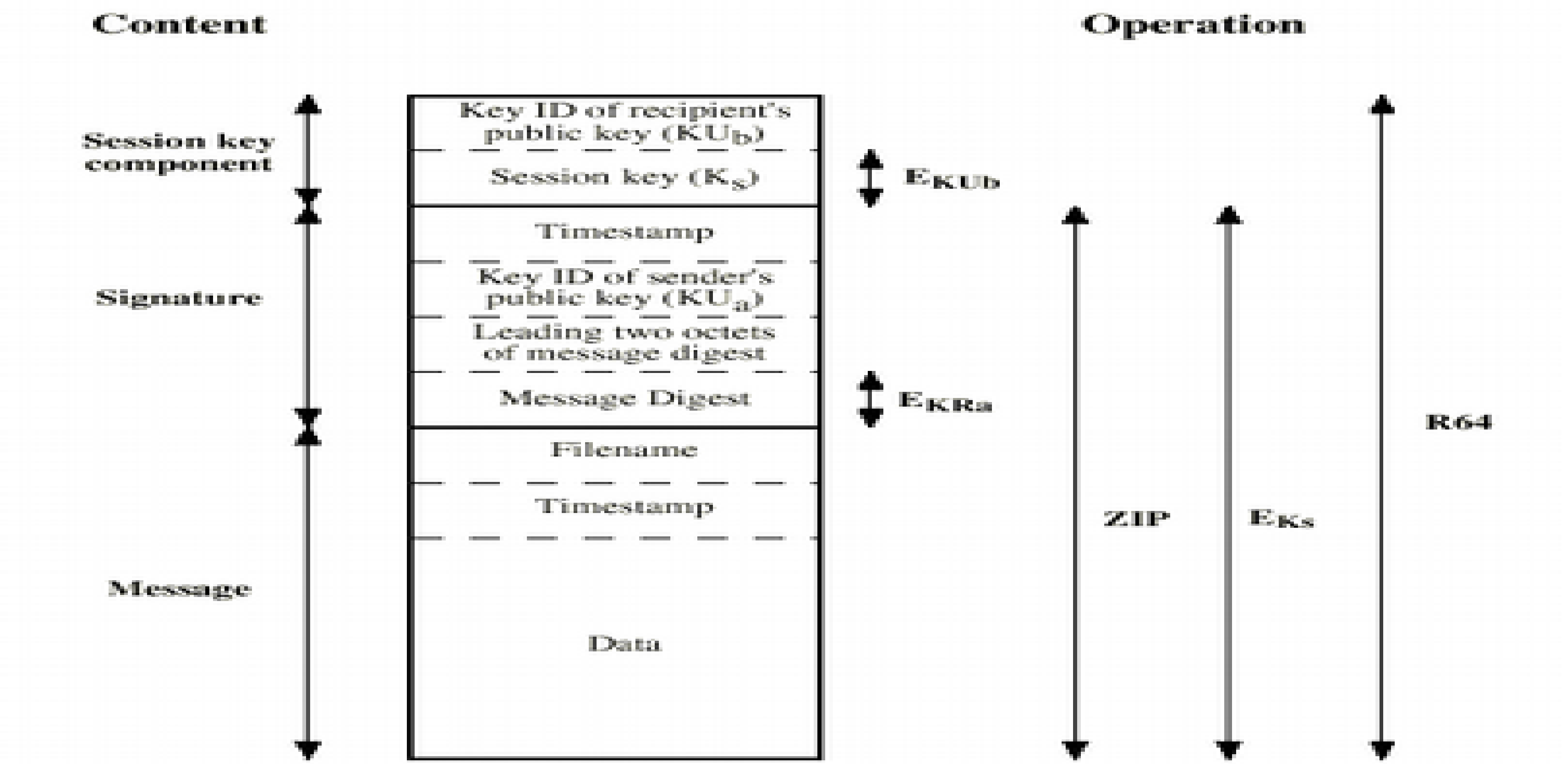
- It is availiable free on a variety of platforms.
- Based on well known algorithms.
- Wide range of applicability
- Not developed or controlled by governmental or standards organizations

Operational Description

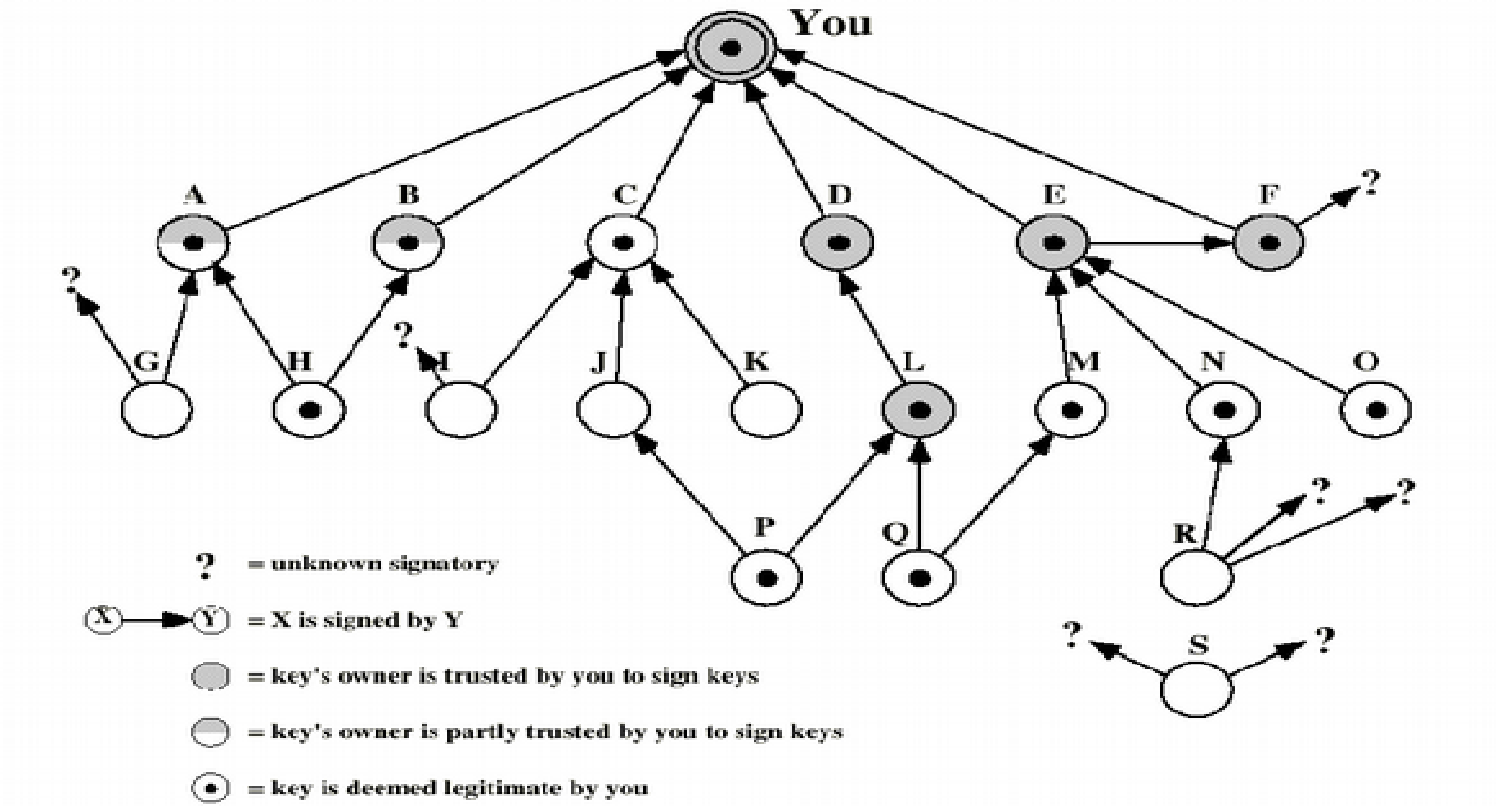
- Consist of five services:
  - Authentication
  - Confidentiality
  - Compression
  - E-mail compatibility
  - Segmentation



Format of PGP



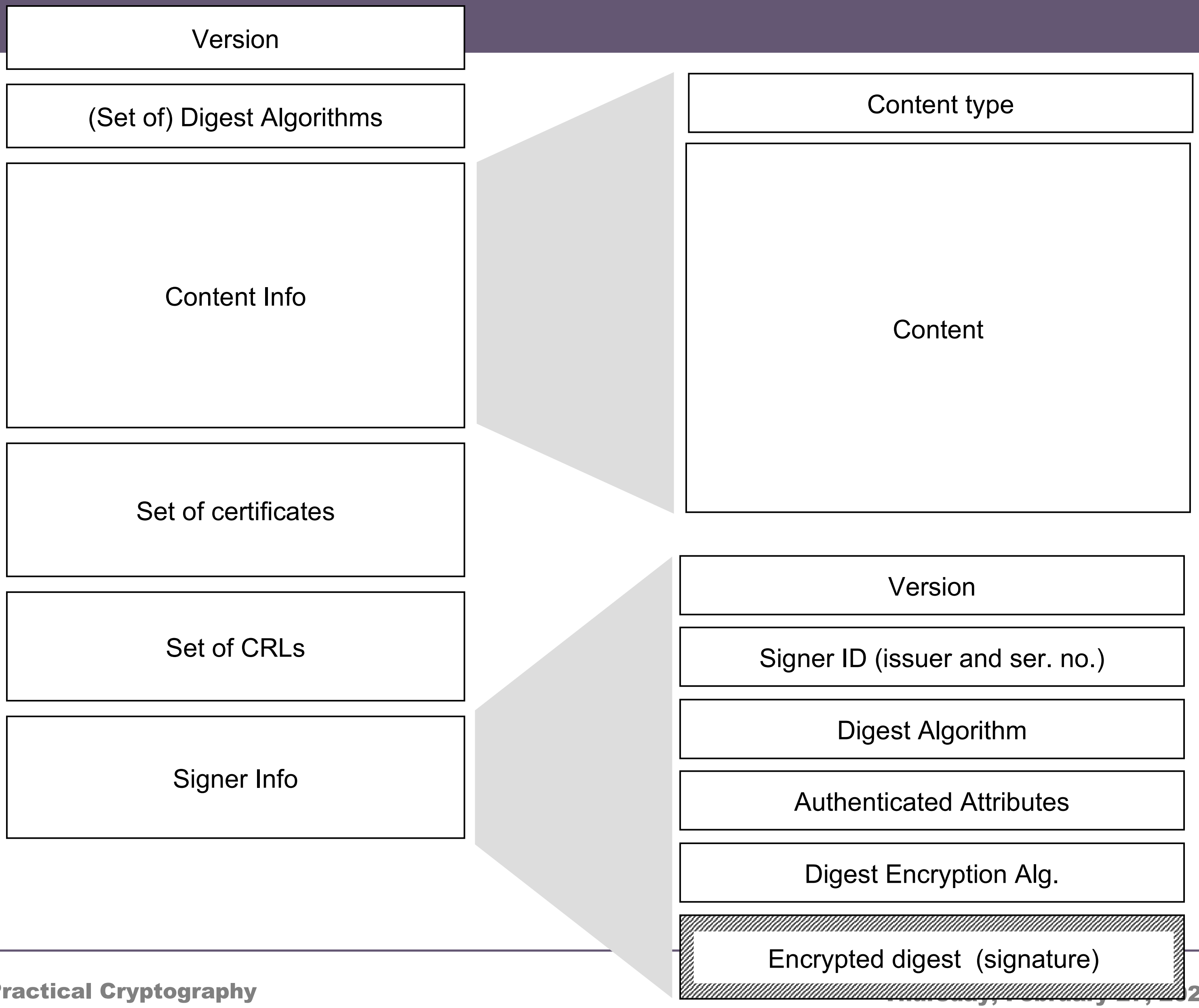
PGP Public Keys


You
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
?
? = unknown signatory
X → Y = X is signed by Y
= key's owner is trusted by you to sign keys
= key's owner is partly trusted by you to sign keys
= key is deemed legitimate by you

Securing a MIME entity

- MIME entity is prepared according to the normal rules for MIME message preparation
- prepared MIME entity is processed by S/MIME to produce a PKCS object
- the PKCS object is treated as message content and wrapped in MIME

PKCS7 “signed data”



Version

(Set of) Digest Algorithms

Content Info

Set of certificates

Set of CRLs

Signer Info

Content type

Content

Version

Signer ID (issuer and ser. no.)

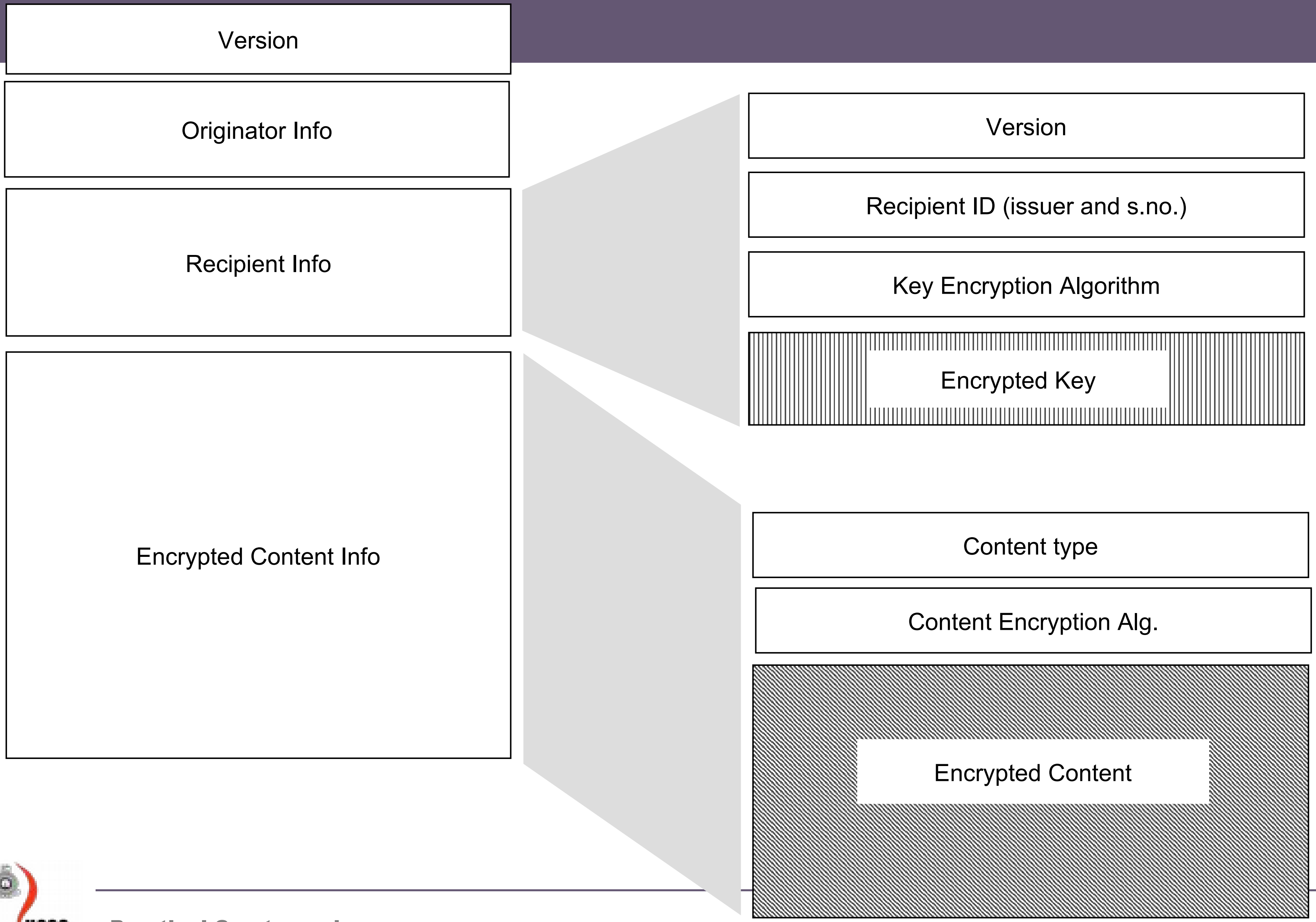
Digest Algorithm

Authenticated Attributes

Digest Encryption Alg.

Encrypted digest (signature)

PKCS7 “enveloped data”



Version

Originator Info

Recipient Info

Encrypted Content Info

Version

Recipient ID (issuer and s.no.)

Key Encryption Algorithm

Encrypted Key

Content type

Content Encryption Alg.

Encrypted Content

Enveloped data – Example

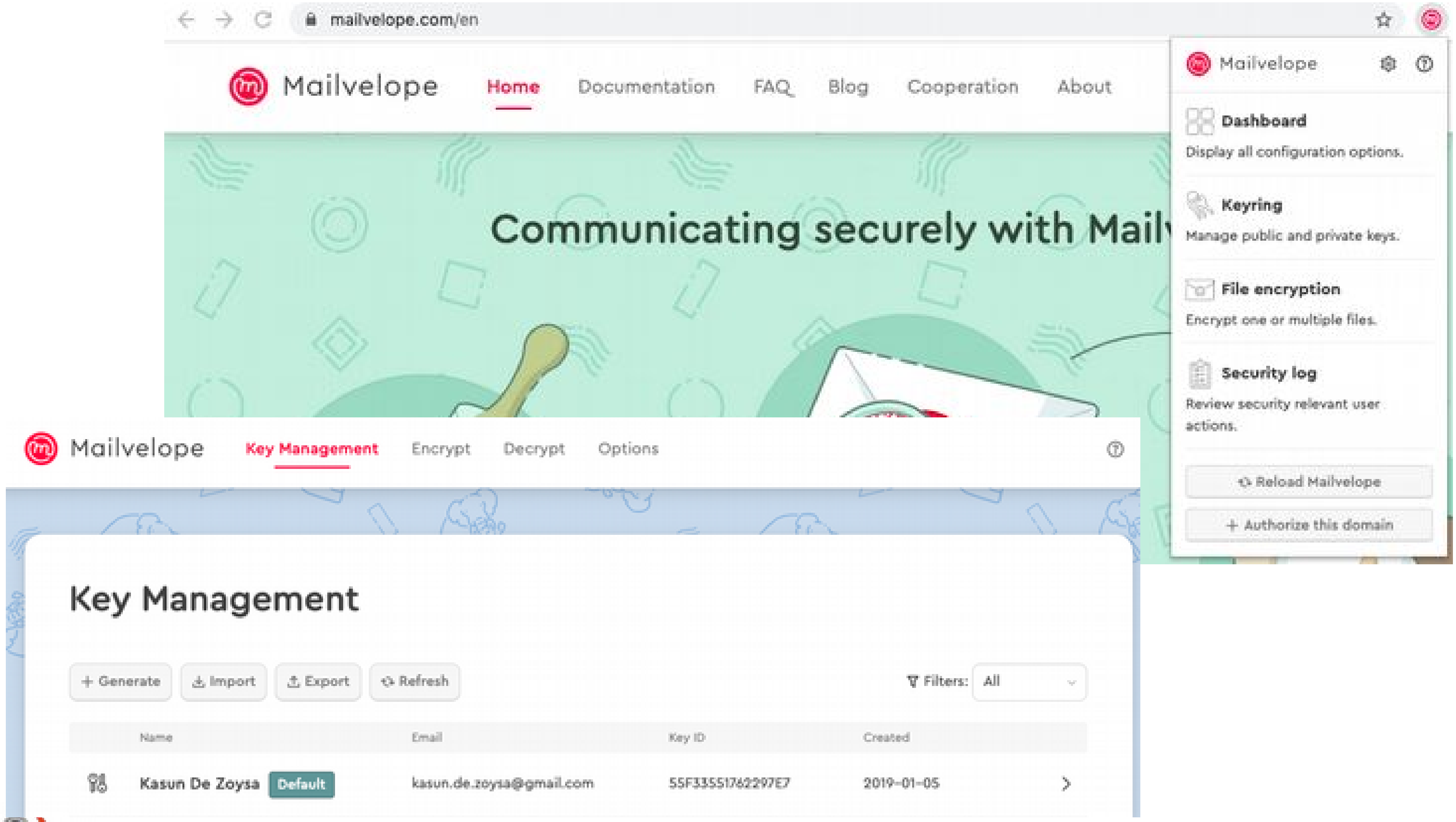
```
Content-Type: application/pkcs7-mime; smime-type=enveloped-data; name=smime.p7m
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7m
```

rfvbnj756tbBghyHhHUujhJhjH77n8HHGT9HG4VQpfyF467GhIGfHfYT6
7n8HHGghyHhHUujhJh4VQpfyF467GhIGfHfYGTrfvbnjT6jH7756tbB9H
f8HHGTrfvhJhjH776tbB9HG4VQbnj7567GhIGfHfYT6ghyHhHUujpfyF4
0GhIGfHfQbnj756YT64V

Clear-signed data – Example

```
Content-Type: multipart/signed; protocol="application/pkcs7-signature";
  micalg=sha1; boundary=boundary42

--boundary42
Content-Type: text/plain

This is a clear-signed message.

--boundary42
Content-Type: application/pkcs7-signature; name=smime.p7s
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7s

ghyHhHUujhJhjH77n8HHGTrfvbnj756tbB9HG4VQpfyF467GhIGfHfYT6
4VQpfyF467GhIGfHfYT6jH77n8HHGghyHhHUujhJh756tbB9HGTrfvbnj
n8HHGTrfvhJhjH776tbB9HG4VQbnj7567GhIGfHfYT6ghyHhHUujpfyF4
7GhIGfHfYT64VQbnj756

--boundary42--
```

www.mailvelope.com



Dashboard
Display all configuration options.

Keyring
Manage public and private keys.

File encryption
Encrypt one or multiple files.

Security log
Review security relevant user actions.

Key Management

	Name	Email	Key ID	Created
	Kasun De Zoysa Default	kasun.de.zoysa@gmail.com	55F33551762297E7	2019-01-05

Discussion