Latin American and Caribbean Internet Addresses Registry
Registro de Direcciones de Internet para América Latina y Caribe
Registro de Endereços da Internet para América Latina e Caribe

An Overview of DNSSEC

August 04, 2011.

Carlos Martínez-Cagnazzo
carlos @ lacnic.net

DNSSEC

- Cryptography 101
- DNSSEC
- Where DNSSEC ?
- How does DNSSEC work ?
- New Resource Records
- Trust Chains

CRYPTOGRAPHY

Cryptography

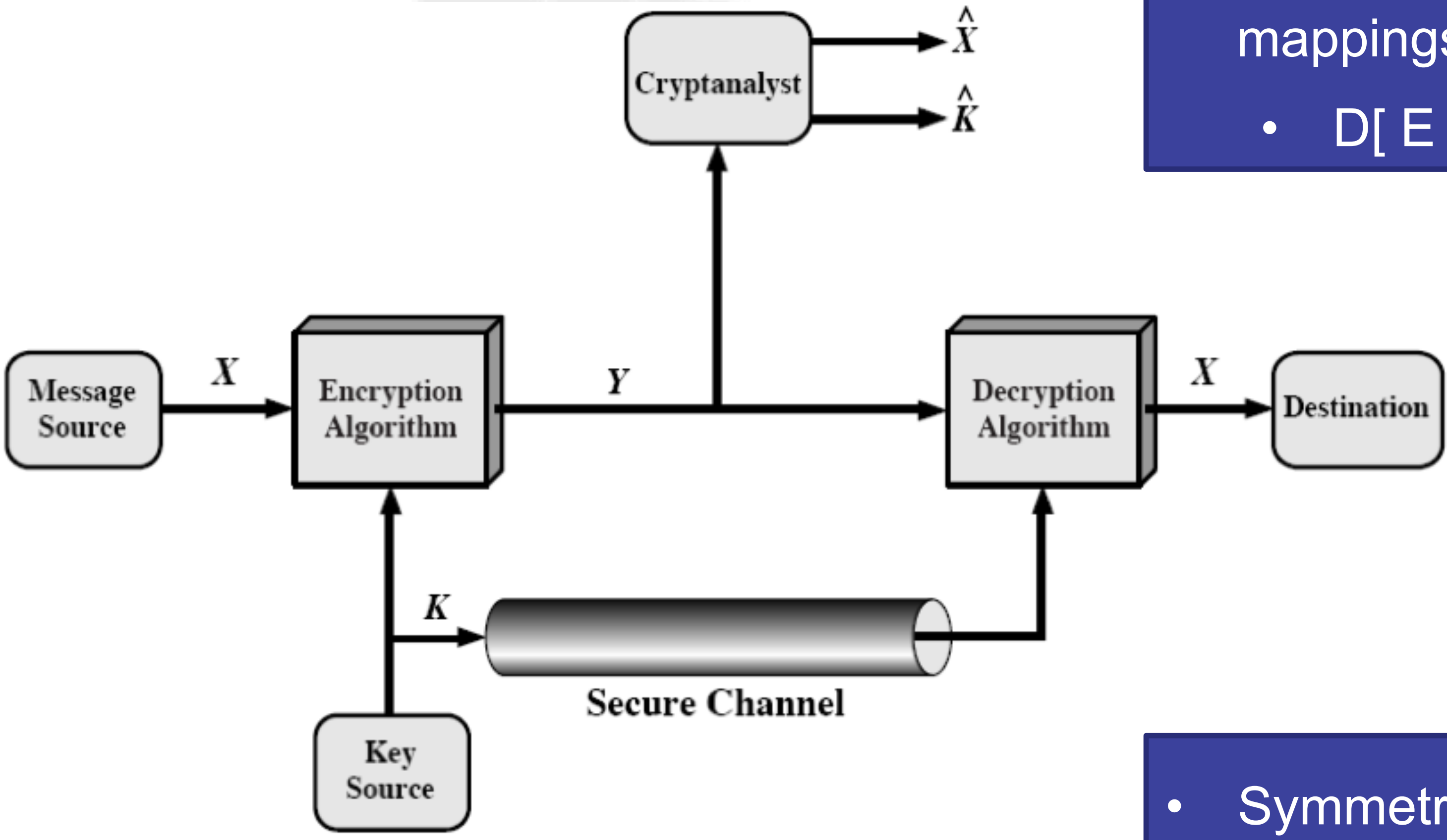
- Cryptography concepts we'll need for DNSSEC
  - Public-key Cryptography
  - Hashing algorithms
  - Digital signatures
  - Trust Chains

Cryptography (ii)

- Let's imagine two parties which need to communicate in a private manner. They will like to see certain **properties** enforced in their data exchanges.
  - They'd like to be sure that no one else has been **able to read** their messages (**privacy** property)
  - They'd like to be sure that no one else has been **able to change or alter** their messages (**integrity** property)
  - They'd like to be sure that the party who sends a message is really who it claims to be (**authentication** property)

Symmetric Cryptography

[Source: Stallings]

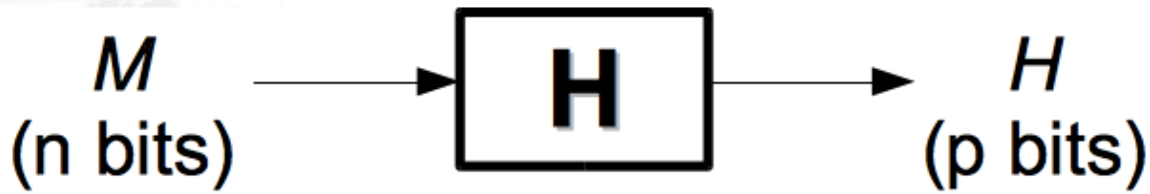


- E[.] y D[.] are two functions both which are inverse mappings of each other
  - D[E [X]] = X

K (the key) is a *parameter* introduced to ease compromise recoveries

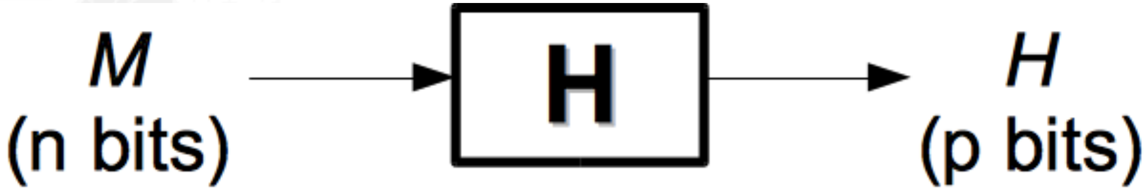
- Symmetric Cryptography
  - $D_K[E_K [X]] = X$

Cryptographic Hashes



- H is a transformation with the following properties
  - p << n
  - For each algorithm ”p” is a given value
    - len(H) is fixed regardless of len(M)
- This means that ***collisions*** do exist
- Collision: If for a pair M1 and M2, H(M1) == H(M2), then M1 and M2 represent a collision
- If H() is chosen and designed carefully then finding collisions is very difficult

Cryptographic Hashes (ii)



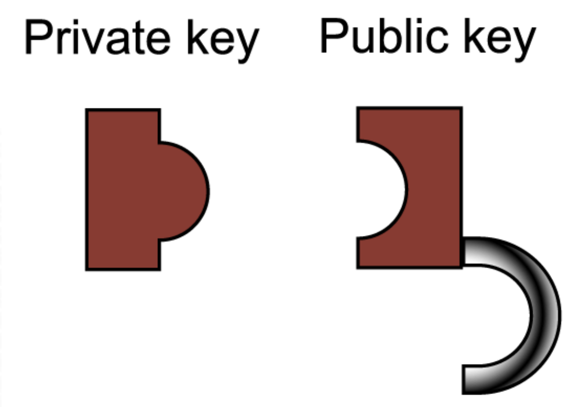
- Intuitively
  - The more "random" the result of a hash "looks", the better it is
- Some well-known algorithms:
- MD5
  - 128 bits
- SHA1 / SHA256
  - 160 / 256 bits

Public-key Cryptography

- Key distribution was always *the* weak point in traditional (symmetric) cryptography
- A lot effort was put to find workarounds and alternatives
- Breakthrough: (*Diffie-Hellman ca. 1976*) "Public-key Cryptography"
- A public-key cryptosystem has the following properties:
  - $D_{K1}[\ E_{K2}\ [X]\ ] = X$
  - D cannot be easily found even if E is known
  - E cannot be broken with a chosen plaintext attack

Public-key Cryptography (ii)

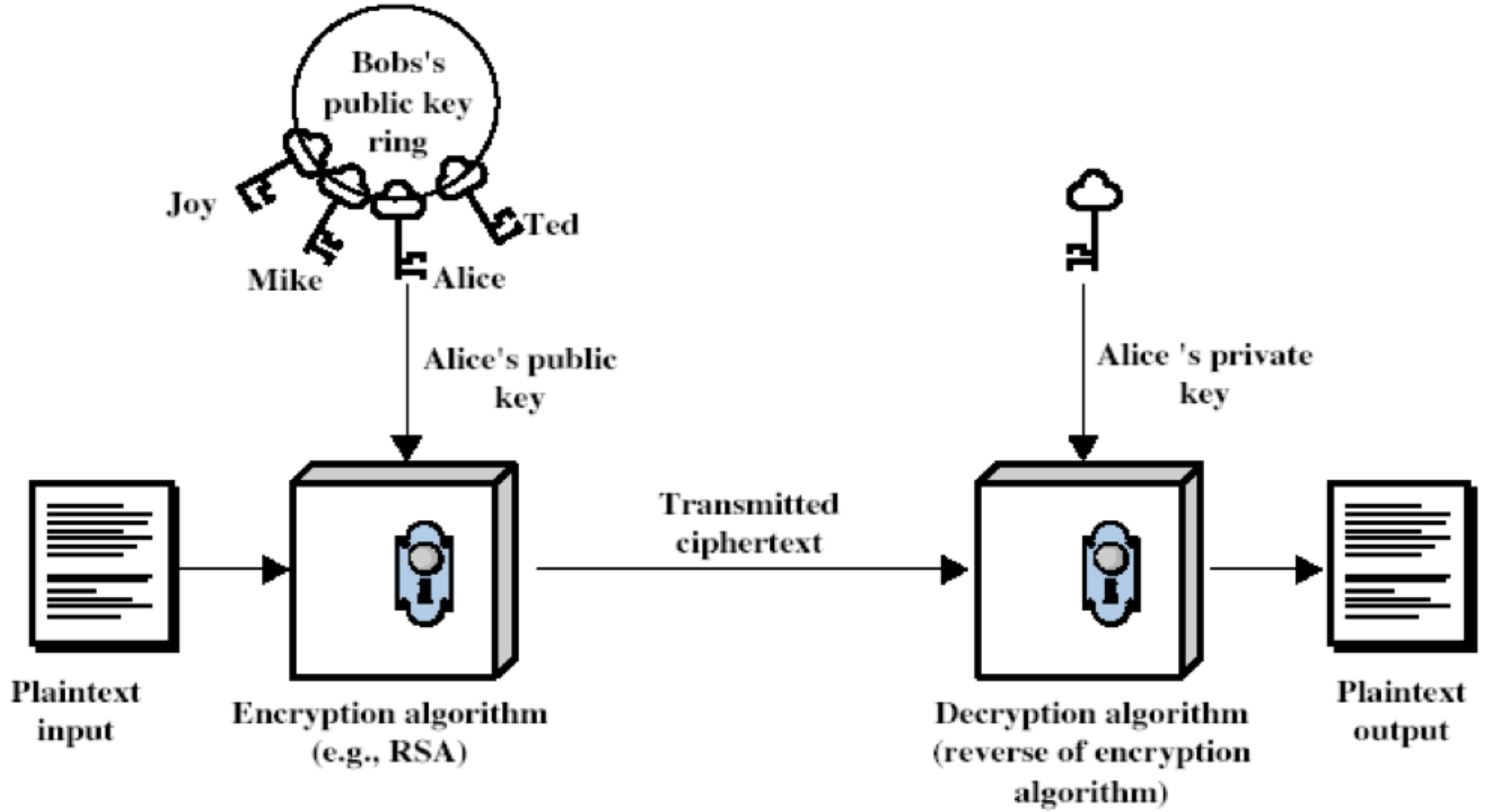
- Each party generates a keypair, that is one public and one secret key
  - Kpub, Kpriv
  - Both keys in the pair are related
    - If one is given the other is also given



- When transmitting a message "X" from A -> B the following computation takes place:
  - $Y = E [Kpub_B, X]$
- When B receives the encrypted message the following computation takes place:
  - $X' = D[Kpriv_B, Y]$

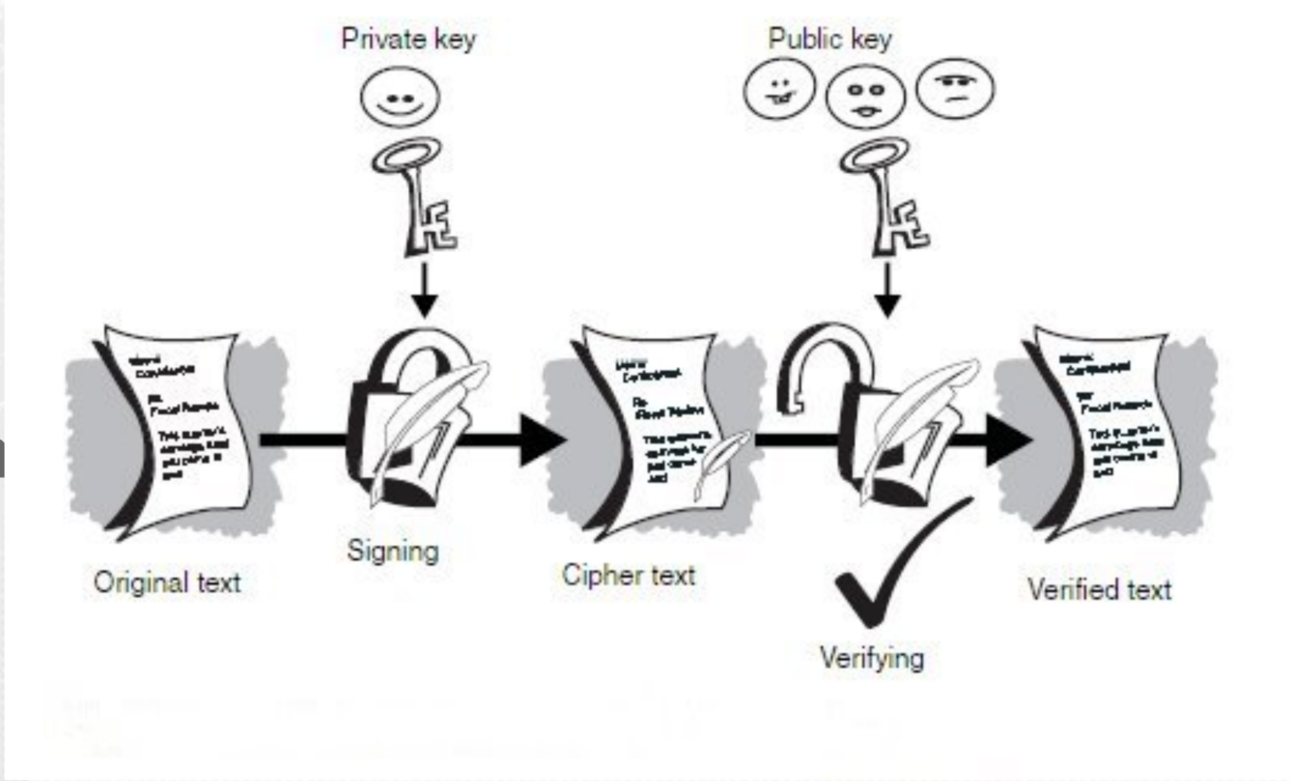
Public-key Cryptography (iii)

- [Source: Stallings]



Digital Signatures

- Goal:
  - *Create integrity proofs of digital documents*
- Usually implemented usin public-key cryptography

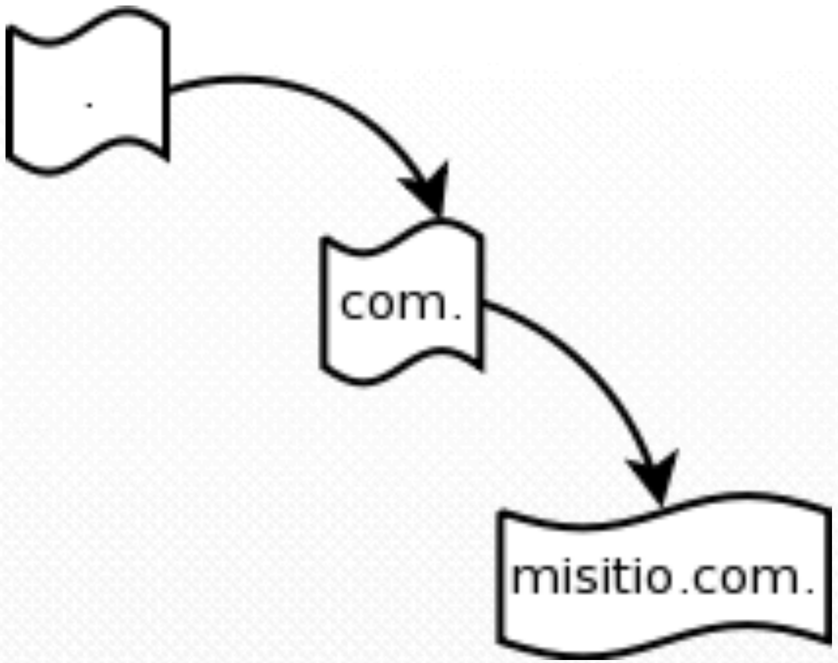


Digital Signatures (iii)

- Given M, a digital document to be signed by party A(lice) to be received by party B(ob)
  - A computes:
    - A hash for M, **H = Hash[M]**
    - A signature for M, **F = E[$Kpriv_A$, H]**
  - A sends the pair {M, F} to B
- When B receives the encrypted message the following computation takes place:
  - A hash for M, **H' = Hash[M]**
  - The hash of the signature is recalculated, **H = D[$Kpub_A$, F]**

Digital Signatures (iii)

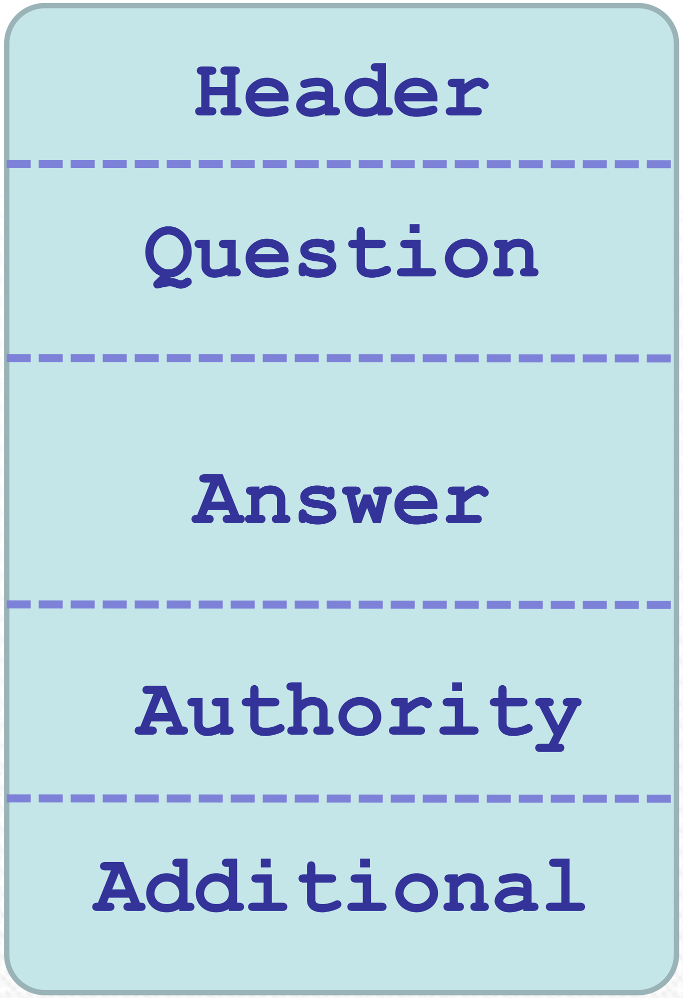
- Trust Chains
  - Each level in the hierarchy signs data in the next one
  - The root needs to be analyzed separately
  - Validation can be either
    - Top down
    - Up down



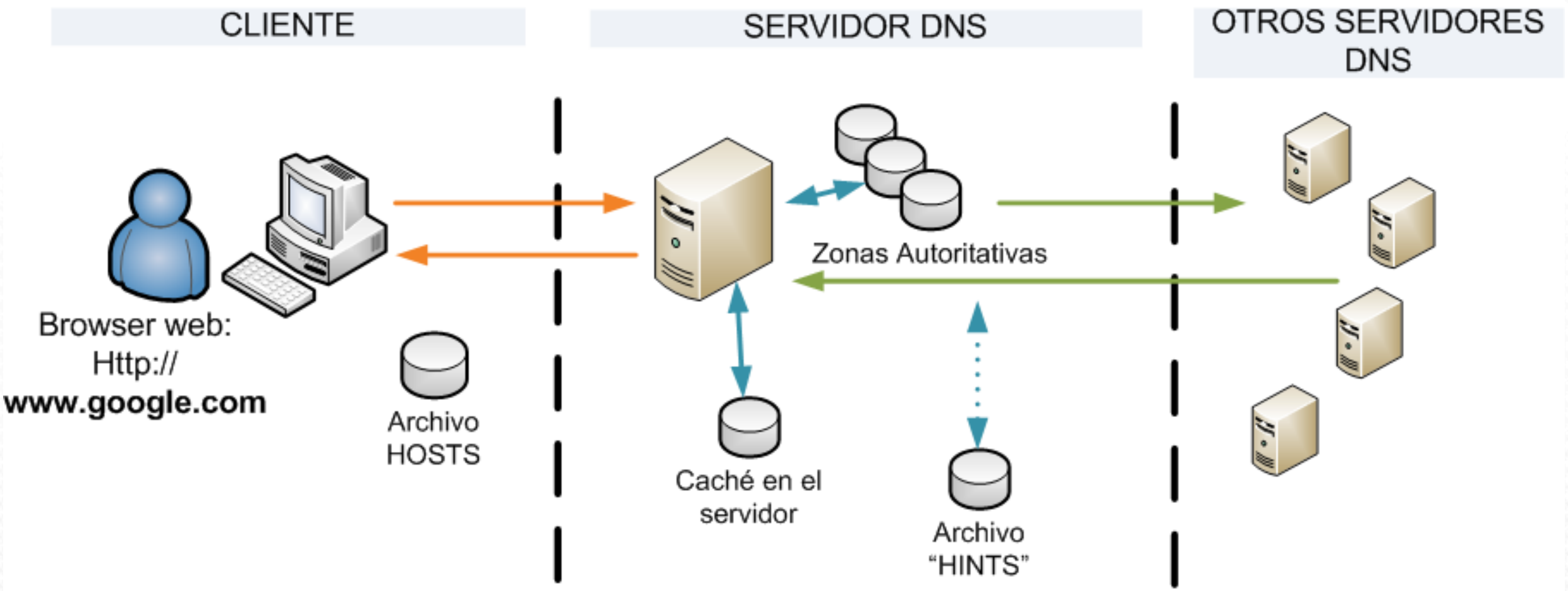
DNSSEC: MOTIVATION

Protocol Specification

- Overview of DNS's wire packet format
  - Header
    - Protocol header
    - Flags (QR, RA, RD,…)
  - Question Section
    - Query we send to the DNS server
      - Tuples (*Name, Type, Class)*
  - Answer Section
    - RRs that answer the query (if any are available), also in (N, T, C) tuple format
  - Authority Section
    - RRs pointing to authoritative servers (optional)
  - Additional Section
    - RRs that may be useful to the querying client (according to the server answering the query)



DNS Queries

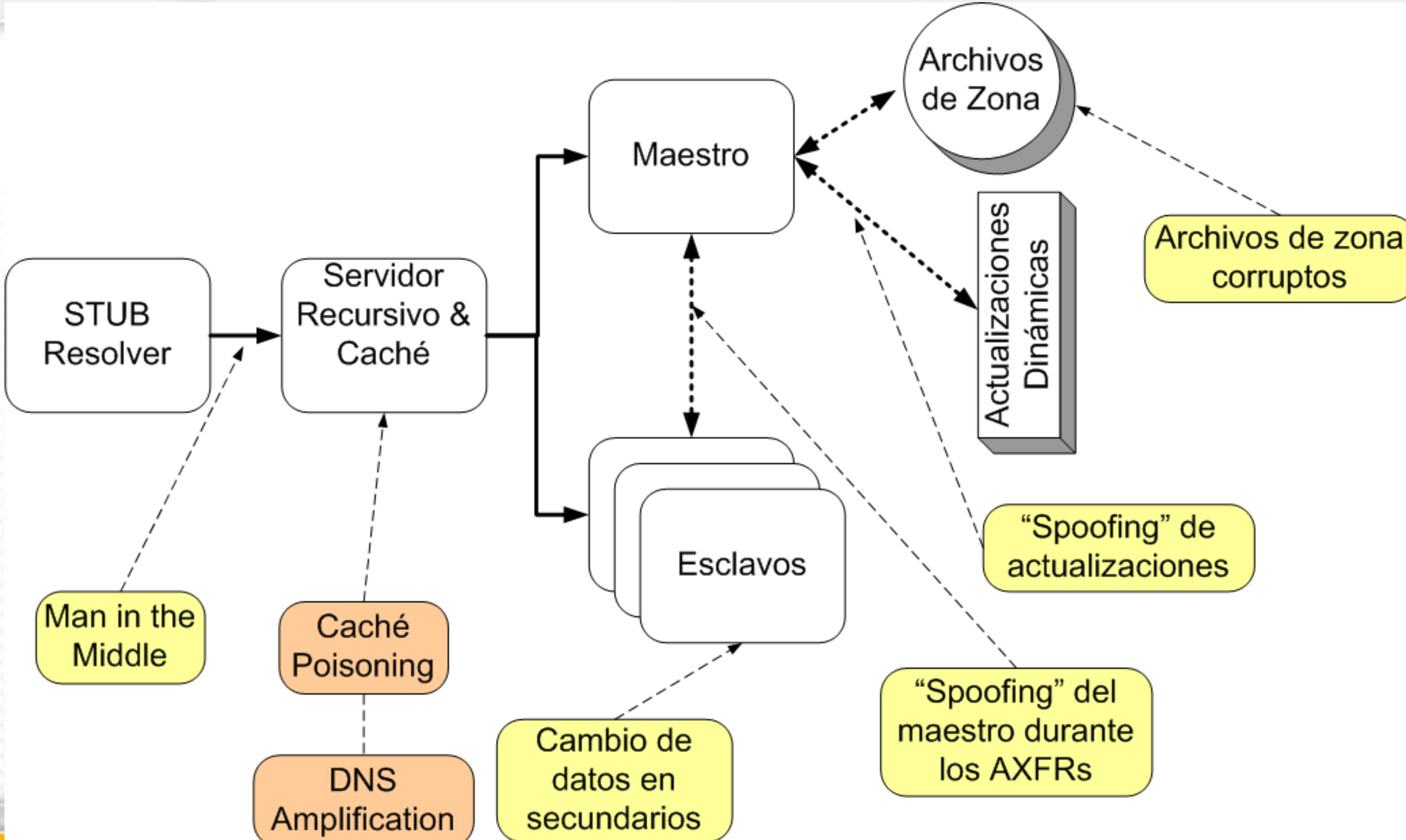


local resolver

local recursive DNS server

Other authoritative servers

Attack Vectors in DNS



Vulnerabilities in DNS

- DNS transmitted data is more prone to spoofing as it is mostly transported over UDP
  - Between master and slaves (AXFR)
  - Between masters and clients (AXFR) "*resolver*"
- Currently the DNS protocol does not have a way to validate information found in a query response
  - Vulnerable to different *poisoning* techniques
  - Poisoned data can be a problem for long periods of time depending on the TTL values of the zones
- Neither do slave servers have a way to authenticate the master servers they're talking to

Introducing DNSSEC

- Threat analysis in the DNS system
  - RFC 3833: "*Threat Analysis of the Domain Name System (DNS)*"
- DNSSEC:
  - "*DNS Security Extensions*"
  - RFC 4033, 4034, 4035
  - ~ May 2005

What does DNSSEC Protect us from?

- DNSSEC will protect us from data corruption and spoofing
  - It provides a way to validate both the integrity and the authenticity of the records contained in a DNS zone
    - DNSKEY/RRSIG/NSEC
  - It provides a way to delegate trust in public keys (trust chains)
    - DS
  - It provides a way to authenticate zone transfers between masters and slaves
    - TSIG

DNSSEC Introduction

- DNSSEC **is not** a new protocol
- Is a set of **extensions** to the DNS protocol as we know it
  - Changes to the wire protocol (EDNS0)
    - Maximum UDP query response extended from 512 to 4096 bytes
  - New resource records added
    - RRSIG, DNSKEY, DS, NSEC
  - New flags added
    - Checking Disabled (CD)
    - Authenticated Data (AD)

DNSSEC Introduction (2)

- New RRs
  - RRSIG: *Resource Record Signature*
  - DNSKEY: *DNS Public Key*
  - DS: *Delegation Signer*
  - NSEC: *Next Secure*
- New Flags:
  - AD: authenticated data
  - CD: checking disabled

DNSSEC Introduction (3)

- A resource record in DNS is a five-value tuple
  - (*name, class, type, TTL, value*)
- The record:
  - `www.company.com. 86400 IN A 200.40.100.141`
  - Is represented by the tuple:
    - Name (www.company.com)
    - Class (IN)
    - Type (A)
    - TTL (86400 seconds)
    - Value (200.40.100.141)

DNSSEC Introduction (4)

- *Resource Record Sets (*RRSets)
  - DNSSEC works by signing RRSets (not individual RRs)
  - An RRSet is a set of resource records that share the same:
    - Class
    - Type
    - Name
- Sample RRSet (TTL omitted for clarity)
  - `www IN A 200.40.241.100`
  - `www IN A 200.40.241.101`

DNSSEC Introduction (5) Zone Signing

- A key pair is created for each zone
  - Each zone has at least one key pair
  - The private key is kept, well, private
    - The private key is used to sign the RRSets in the zone
  - The public key is published in DNS using DNSKEY records
    - The private key is also used to verify the signatures of the RRSets
  - An RRSet can have multiple signatures generated using different key pairs

DNSSEC Introduction (6)

- The digital signature of a RRSet is returned in a special RRSIG record with the query answer
- Example:

```
~ carlosm$ dig +dnssec www.nic.se

;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADDITIONAL: 1

;; ANSWER SECTION:
www.nic.se.            60     IN     A       212.247.7.218
www.nic.se.            60     IN     RRSIG   A 5 3 60 20101021132001
20101011132001 23369 nic.se. HeeUZ5h5iExK5uU1SuNRIf2Dbmh2/
aWV8FkjmzixUzTAVrHv39PfmfnG DHdHoZxoz85hqqYiWb
+t9EZh5+iqxQk8AxRDic9Nn6WxifOoWeS+IUKQ
rVyqXf1NtkZvu1A325vwa8obtbeVGVkhqg6bDIjKYeHixjlQ4cRoFcEW Izk=

;; AUTHORITY SECTION:
nic.se.                2974   IN     NS      ns3.nic.se.
nic.se.                2974   IN     NS      ns2.nic.se.
nic.se.                2974   IN     NS      ns.nic.se.
nic.se.                3600   IN     RRSIG   NS 5 2 3600
20101021132001 20101011132001 23369 nic.se. GSzAUC3SC3D0G/
iesCOPnVux8WkQx1dGbw491RatXz53b7SY0pQuyT1W
eb063Z62rtX7etynNcJwpKlYTG9FeMbDceD9af3KzTJHxq6B+Tpmmxyk
FoKAVaV0cHTcGUXSObFquGr5/03G79C/YHJmXw0bHun5ER5yrOtOLegU IAU=
```

Trust Chains

- How do clients verify a zone's RRSets?
  - It queries for the corresponding DNSKEY
  - The necessary computations are carried out and then compared with the signature in the RRSIG
    - If they match the signatures are valid
- But, how can we trust the DNSKEY? It listed on the same zone we want to verify!
  - We need to validate the **trust chain**

Trust Chains (ii)

- DS Record "*Delegation Signature*"
  - DS records "sign" the keys in their child zones
  - In this way one can also verify the DNSKEY as it is signed when the parent zone is signed
- DS records contain a hash of the public key
  - That is a hash of the DNSKEY's record content
- DS records in the parent zone are signed with the keys of the parent zone
- To complete the full trust chain we also need the **root of the DNS** to be signed

Trust Chains (iii)

- What about the root zone ?
  - The root zone has no parent zone where a DS record could be placed
  - The DNS root has been signed since July 2010
    - *[http://www.root-dnssec.org]*
  - The DS record for "." is obtained out-of-band and installed locally in each server
    - [http://data.iana.org/root-anchors/root-anchors.xml]
    - . IN DS 49AAC11D7B6F6446702E54A1607371607A1A41855200FD2CE1CDDE32F24E8FB5

DNSSEC Introduction (9) Root Zone Signing

- How is the the root trust anchor verified?
- It is verified also out-of-band
  - It can be downloaded using http/https
  - Several validation mechanisms are in place (X.509 certs, PGP signatures)
  - It is locally installed in the same way the root zone itself is configured locally

DNSSEC Introduction (10) Denial of Existence

- "NXDOMAIN" answers
  - Provide “denial of existence” answers via a flag on the “Header” pseudo-section
  - NXDOMAINS are cached in the same way as other responses are
    - Forging NXDOMAINs is a DDoS attack vector
- How can non-existence be signed ?
  - We need an RRSet to sign
    - Remember that DNSSEC always signs RRSets
  - Two different techniques have been proposed:
    - NSEC and NSEC3

LACNIC

Latin American and **Caribbean** Internet Addresses Registry
Registro de Direcciones de Internet para **América Latina** y **Caribe**
Registro de Endereços da Internet para **América Latina** e **Caribe**

Thank You!

Questions?