ยินดีต้อนรับสู่

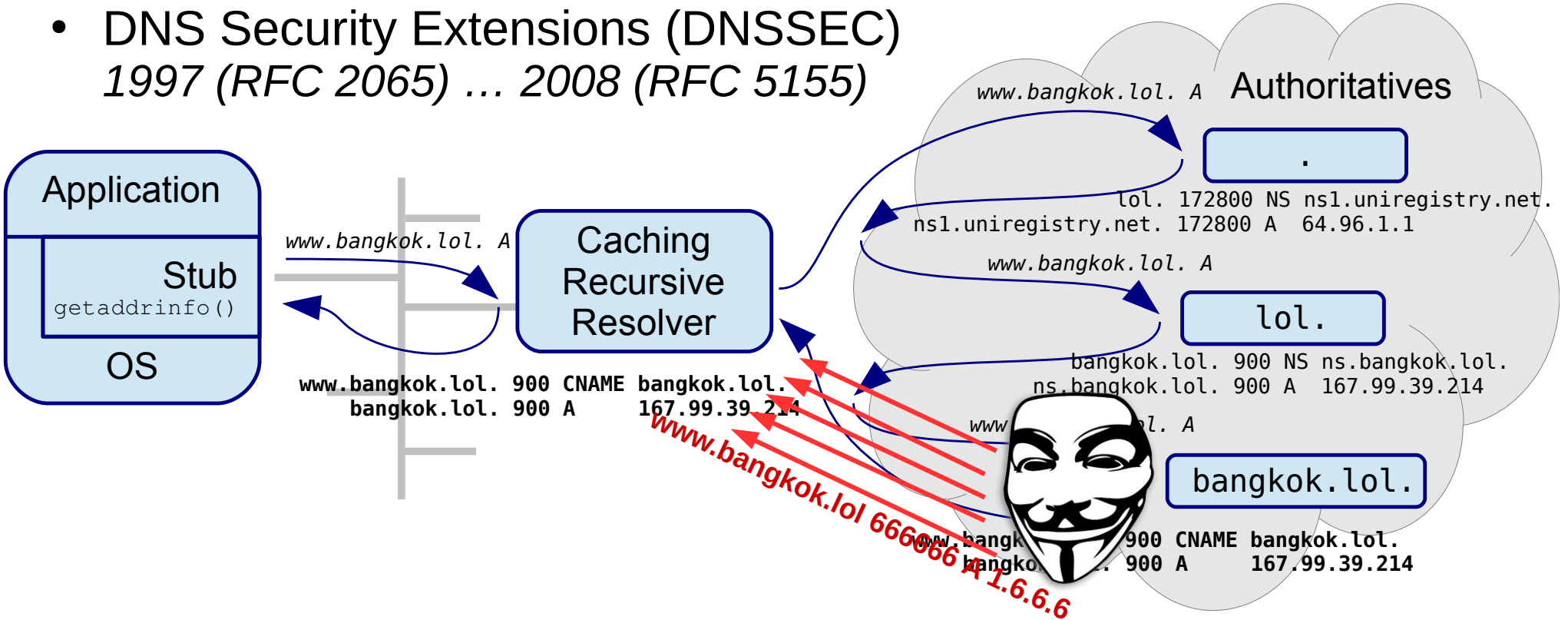
CYBER SECURITY WORKSHOP

Vulnerabilities

Bangkok
8-9 May 2019

Domain Name System - security

- Random bits (65.536 query ID * source ports) & ***Caching*** as security mechanism
- DNS Security Extensions (DNSSEC)
  1997 (RFC 2065) … 2008 (RFC 5155)

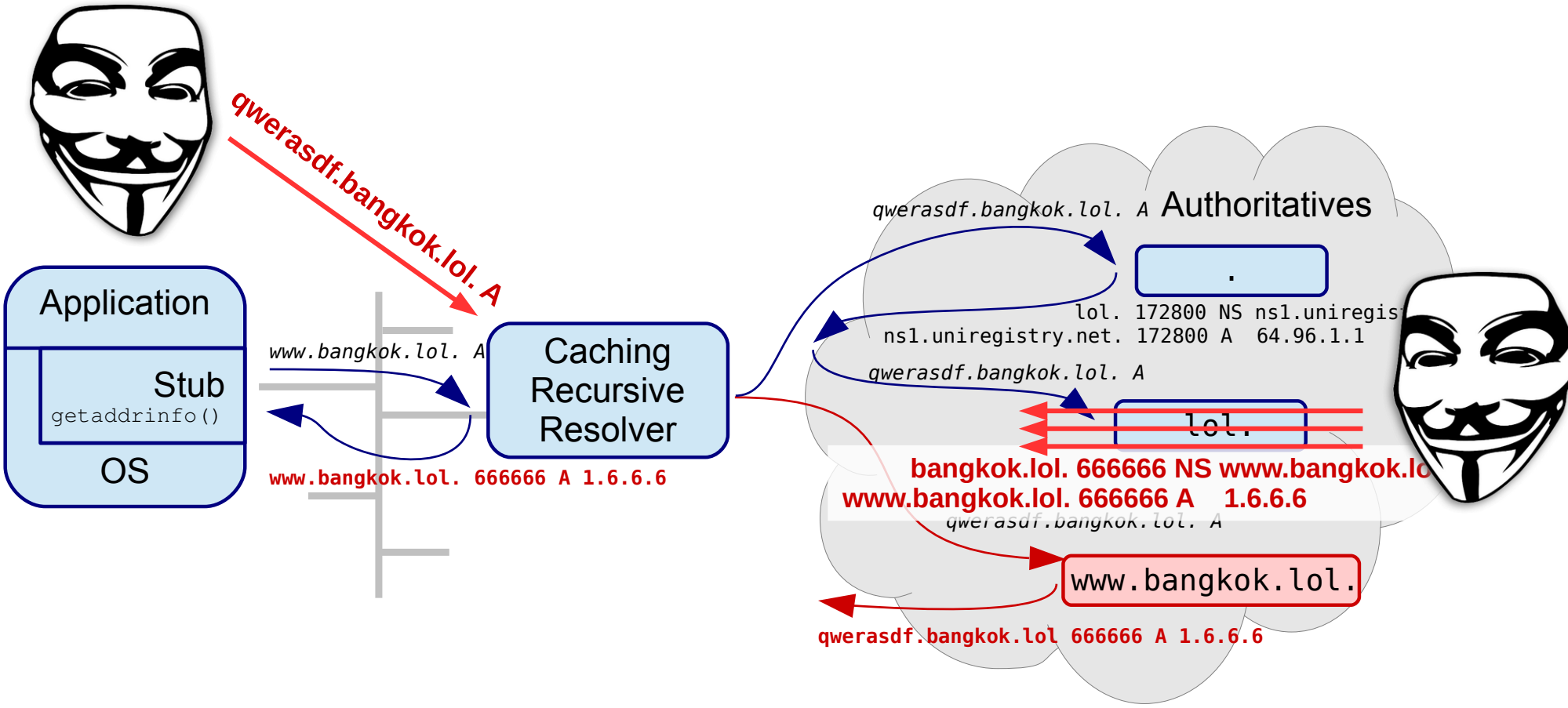


Domain Name System

- Random bits (65.536 query ID * 6
  Caching als s echanis
- DNS Securit ONS
  1997 (RFC FC 5155)

TTL saves you?!?
I don't think so...

Domain Name System - security



Domain Name System - security

# Bits	50% chance	5% chance	Method
16	10 seconds	1 second	Query ID
26	2.8 hours	17 minute	1024 source ports
34	28 days	2.8 days	All source ports + 2 bits server selection
44	288444 days	2844.4 days	0x20 hack

Domain Name System - security

- Help with spoofing DNS responses

Fragmentation Considered Poisonous

Amir Herzberg[†] *and Haya Shulman*[‡]
Dept. of Computer Science, Bar Ilan University
[†]*amir.herzberg@gmail.com,* [‡]*haya.shulman@gmail.com*

Abstract

ent practical *poisoning* and *name-server block-*
s on standard DNS resolvers, by *off-path,*
dversaries. Our attacks exploit large DNS
hat cause IP fragmentation; such long re-
creasingly common, mainly due to the use

narios, where DNSSEC is partially or

sary that is able to send spoofed packets (but not to intercept, modify or block packets). The most well known is Kaminsky's DNS poisoning attack [21], which was exceedingly effective against many resolvers at the time (2008). Kaminsky's attack, and most other known DNS poisoning attacks, allows the attacker to cause resolvers to provide incorrect (poisoned) responses to DNS queries of the clients, and thereby 'hijack' a domain name. We

Domain Name System - security

- Help with spoofing DNS responses

attacker ICMP frag needed→ authoritative

Offsets	Octet	0								1								2								3								
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
0	0	v4				IHL = 20				TOS								Total Length = 56																IP Header
4	32	IPID																x	DF	MF	Frag Offset													
8	64	TTL								Protocol = 1								IP Header Checksum																
12	96	Source IP = 6.6.6.6																																
16	128	Destination IP = 2.2.2.2																																
20	160	Type = 3								Code = 4								ICMP Checksum																ICMP Header
24	192	Unused																MTU = 100																
28	224	v4				IHL = 20				TOS								Total Length = 76																IP Header
32	256	IPID																x	DF	MF	Frag Offset													
36	288	TTL								Protocol = 17								IP Header Checksum																
40	320	Source IP = 2.2.2.2																																
44	352	Destination IP = 7.7.7.7																																
48	384	Source Port = 53																Destination Port = 12345																UDP Heade
52	416	Length = 56																UDP Checksum = 0																

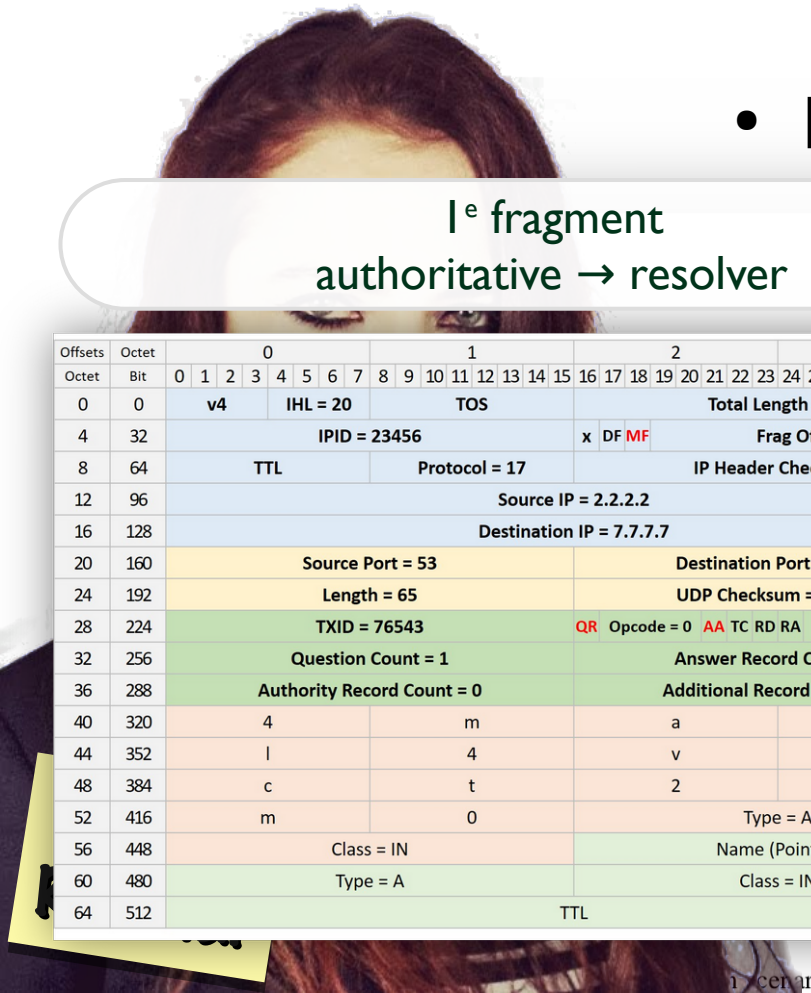
ent practical
s on stand
dversaries.
hat cause
creasingly common, mainly due to the use

narios, where DNSSEC is partially or

poisoning attacks, allows the attacker to cause resolvers to provide incorrect (poisoned) responses to DNS queries of the clients, and thereby 'hijack' a domain name. We

Domain Name System - security

- Help with spoofing DNS responses



1e fragment
authoritative → resolver

Offsets	Octet	0								1								2								3								
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
0	0	v4				IHL = 20				TOS								Total Length = 85																IP Header
4	32	IPID = 23456																x	DF	MF	Frag Offset = 0												IP Header
8	64	TTL								Protocol = 17								IP Header Checksum																IP Header
12	96	Source IP = 2.2.2.2																																IP Header
16	128	Destination IP = 7.7.7.7																																IP Header
20	160	Source Port = 53																Destination Port = 12345																UDP Header
24	192	Length = 65																UDP Checksum = 0x14de																UDP Header
28	224	TXID = 76543																QR	Opcode = 0			AA	TC	RD	RA	Z			RCODE = 0					DNS Header
32	256	Question Count = 1																Answer Record Count = 1																DNS Header
36	288	Authority Record Count = 0																Additional Record Count = 1																DNS Header
40	320	4								m								a								i								Question Section
44	352	l								4								v								i								Question Section
48	384	c								t								2								i								Question Section
52	416	m								0								Type = A																Question Section
56	448	Class = IN																Name (Pointer)																Answer Section
60	480	Type = A																Class = IN																Answer Section
64	512	TTL																																Answer Section

2e fragment
attacker → resolver

Offsets	Octet	0								1								2								3								
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
0	0	v4				IHL = 20				TOS								Total Length = 85																IP Header
4	32	IPID = 23456																x	DF	MF	Frag Offset = 48												IP Header
8	64	TTL								Protocol = 17								IP Header Checksum																IP Header
12	96	Source IP = 2.2.2.2																																IP Header
16	128	Destination IP = 7.7.7.7																																IP Header
20	160	Data Length = 4																IPv4 Address																Answer Section
24	192	= 2.2.2.2																Name = 0								Type								Answer Section
28	224	= OPT																UDP Payload Size = 4096																Additional Section
32	256	Version = 0																DO	Z															Additional Section
36	288	= 0																																Additional Section

sary that is able to send spoofed packets (but not to intercept, modify or block packets). The most well known is Kaminsky's DNS poisoning attack [21], which was exceedingly effective against many resolvers at the time (2008). Kaminsky's attack, and most other known DNS poisoning attacks, allows the attacker to cause resolvers to provide incorrect (poisoned) responses to DNS queries of the clients, and thereby 'hijack' a domain name. We

Domain Name System - security

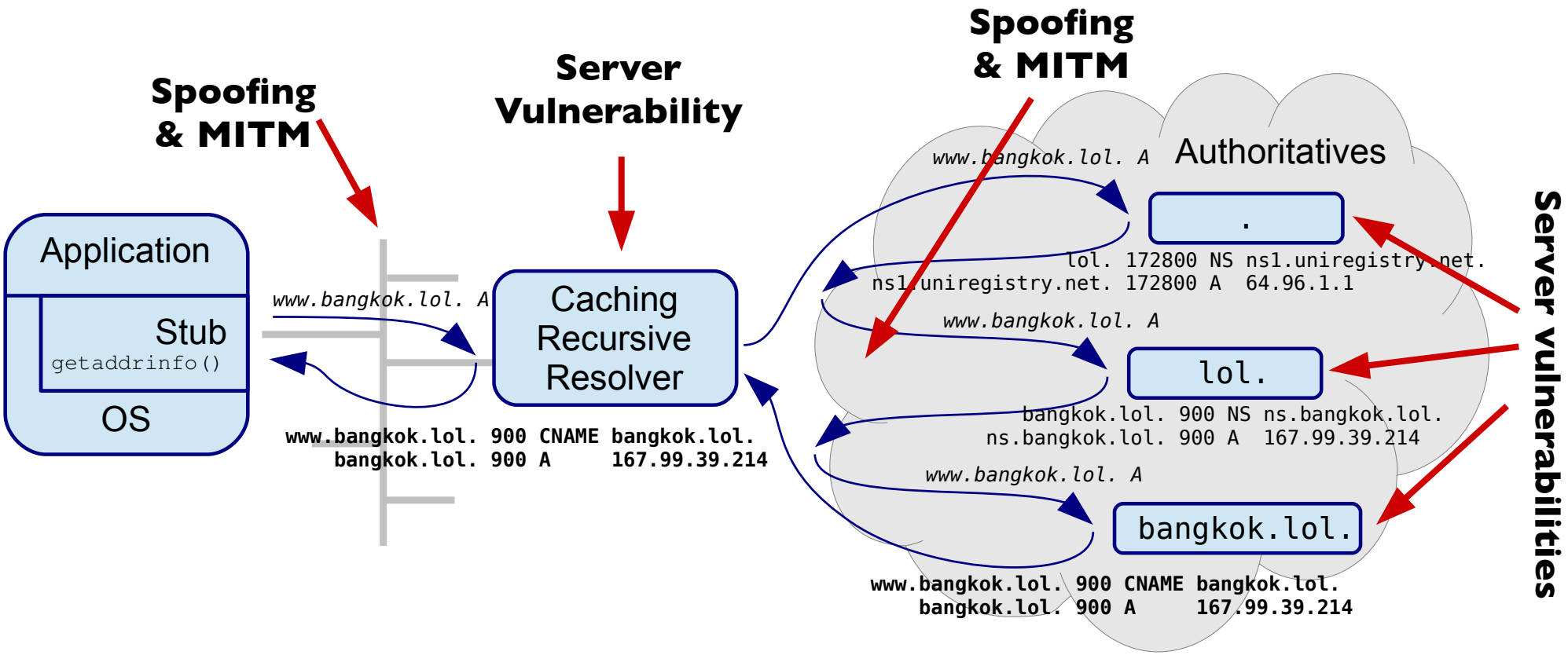
bits	50% chance	5% chance	Method
~~16~~	~~10 seconds~~	~~1 seconde~~	~~Query ID~~
~~26~~	~~2,8 uur~~	~~17 minutes~~	~~1024 source ports~~
2	0 seconds	0 seconds	~~All source ports~~ 2 bits server selection
~~44~~	~~288444 days~~	~~2844.4 days~~	~~0x20 hack~~
5	0 seconds	0 seconds	IP ID

Domain Name System - security

bits	50% chance	5% chance	Method
~~16~~	~~10 seconds~~	~~1 seconde~~	~~Query ID~~
~~26~~	~~2,8 uur~~	~~17 minutes~~	~~1024 source ports~~
2	0 seconds	0 seconds	~~All source ports~~ 2 bits server selection
~~44~~	~~288444 days~~	~~2844.4 days~~	~~0x20 hack~~
5	0 seconds	0 seconds	IP ID
69	2,928,370,544 year	292,837,054 year	IPv6 /64 source address

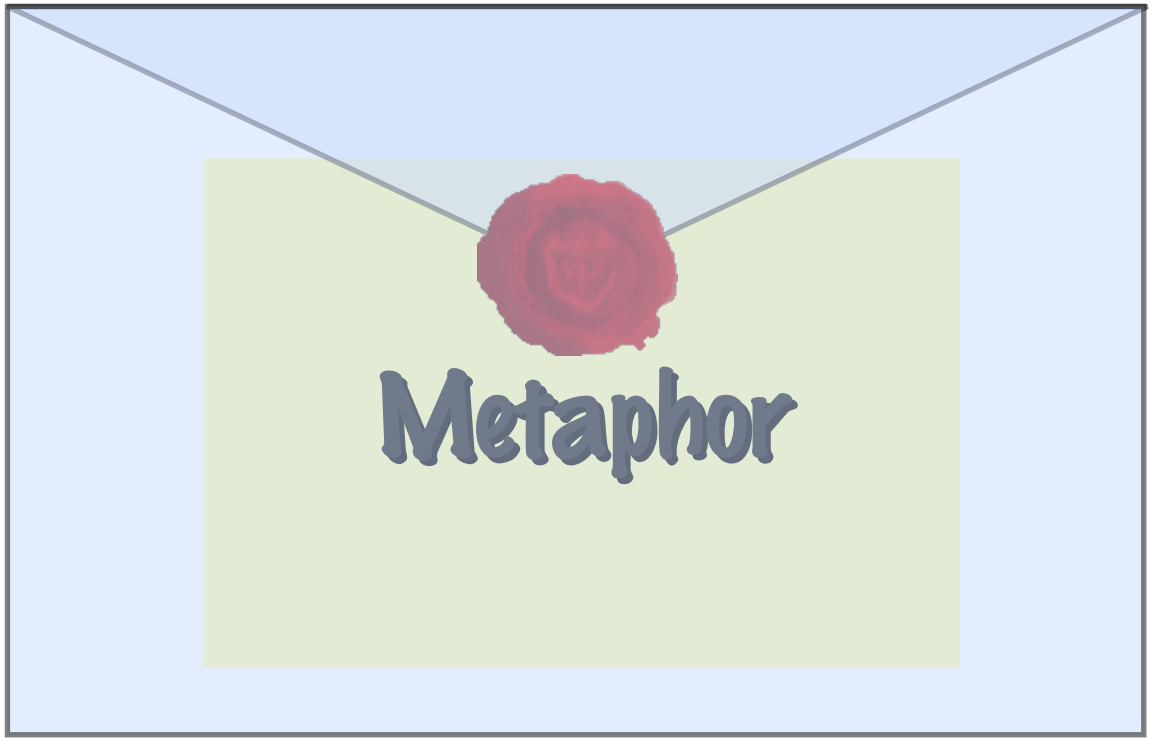
Domain Name System - security

- It's not just spoofing



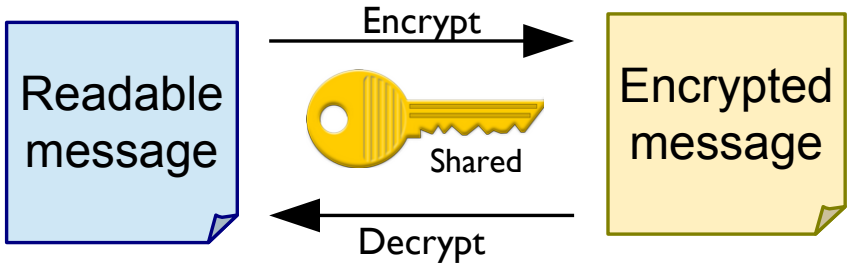
DNS Security Extensions (DNSSEC)

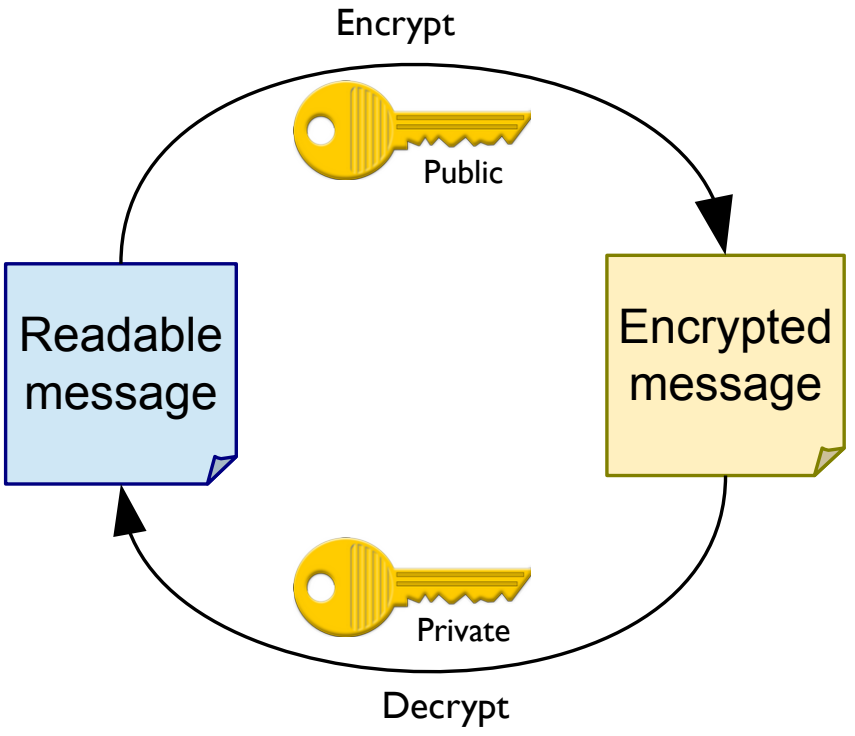
- end-to-end security on top of DNS



DNS Security Extensions (DNSSEC)

Refresher Public Key Crypto

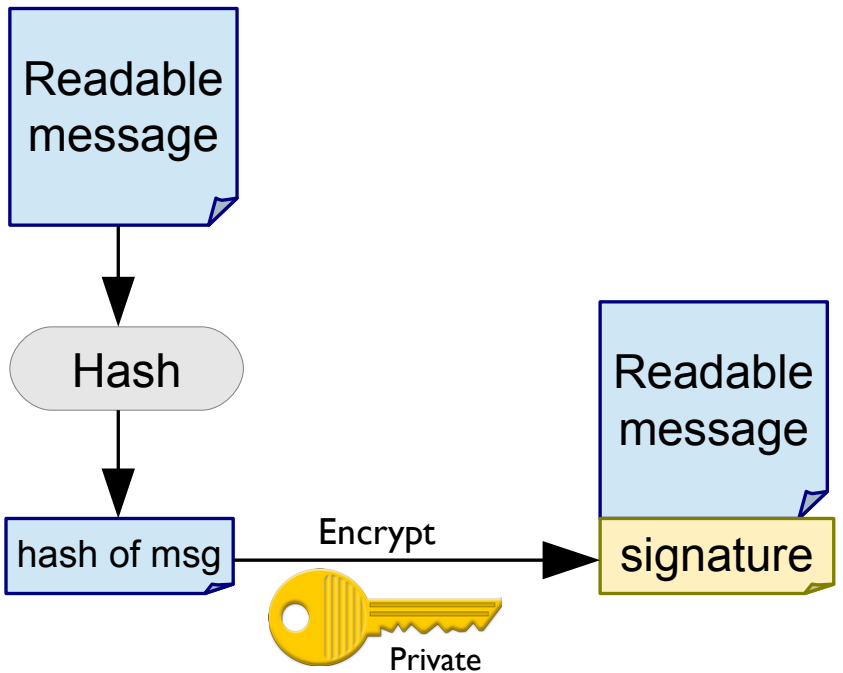


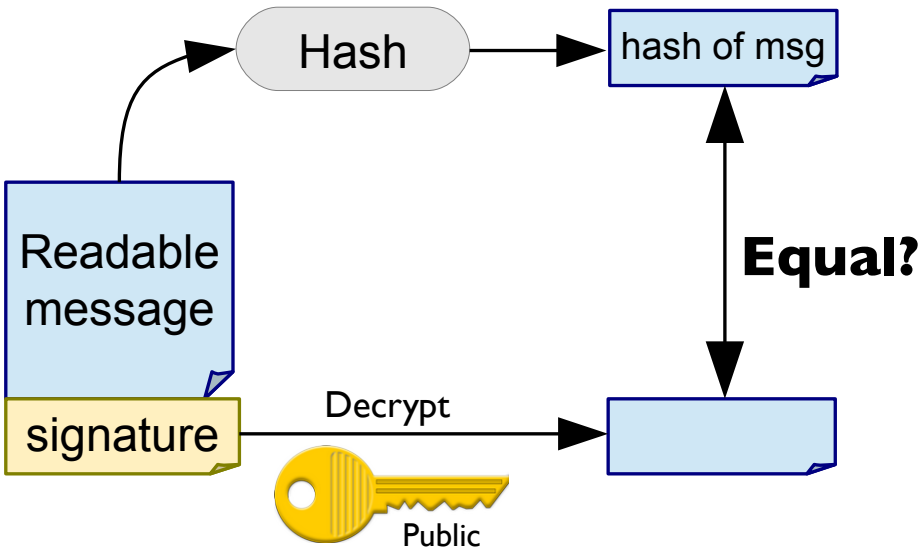


- Symmetric encryption
- Asymmetric encryption

DNS Security Extensions (DNSSEC)

Public Key Crypto – Signing

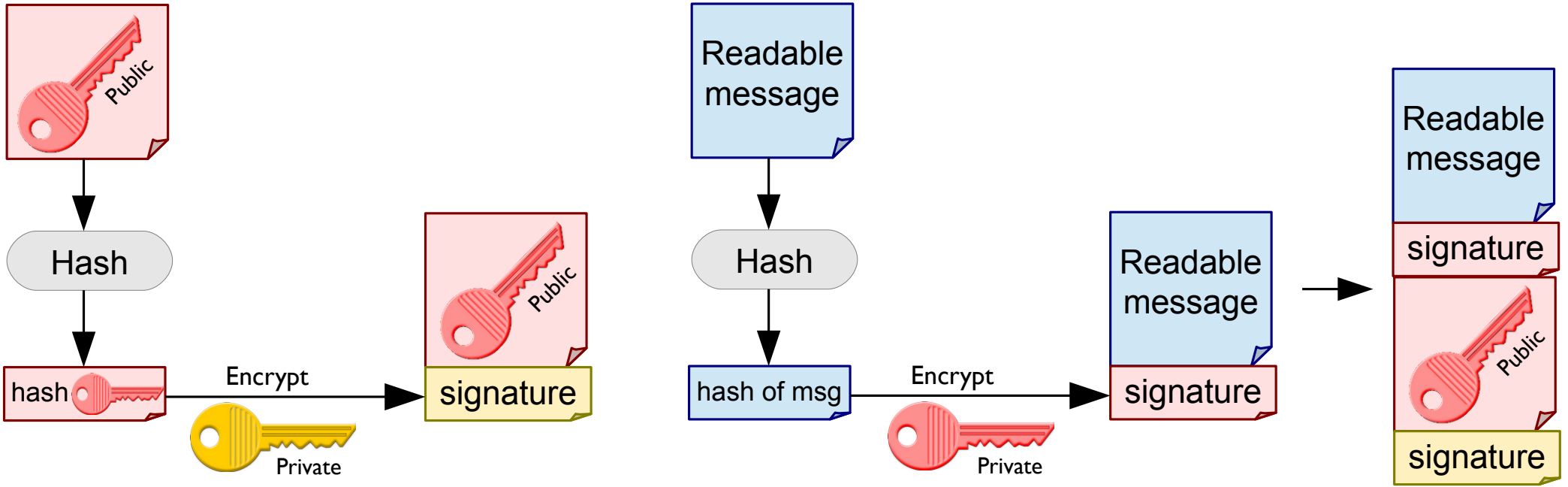


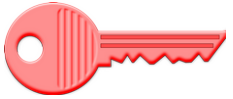
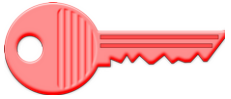


- Create signature
- Verify signature

DNS Security Extensions (DNSSEC)

Public Key Crypto – Delegating Authority



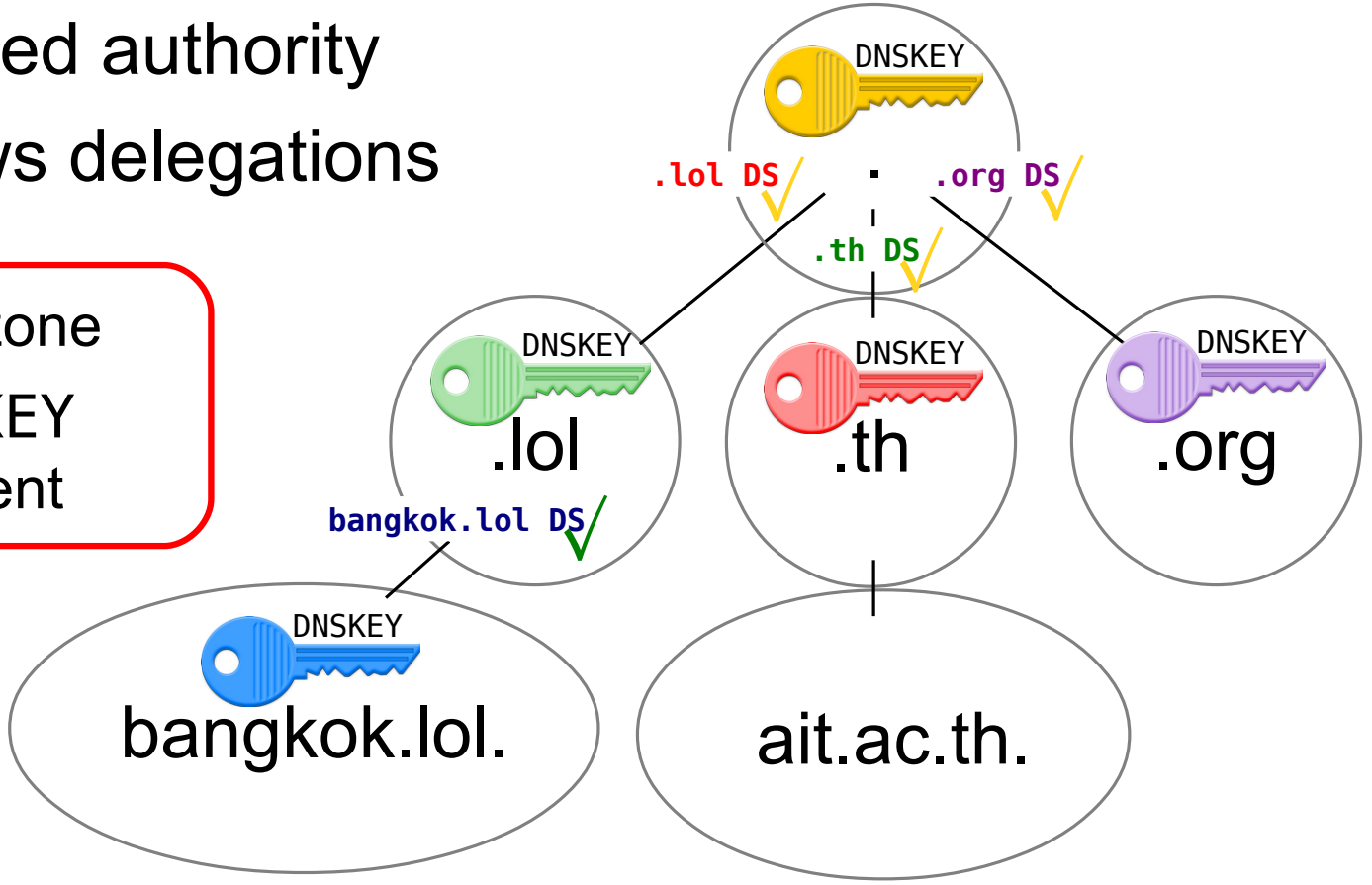
- Building the chain of trust
   authorizes 
-  signs the message

DNS Security Extensions (DNSSEC)

Chain of Trust

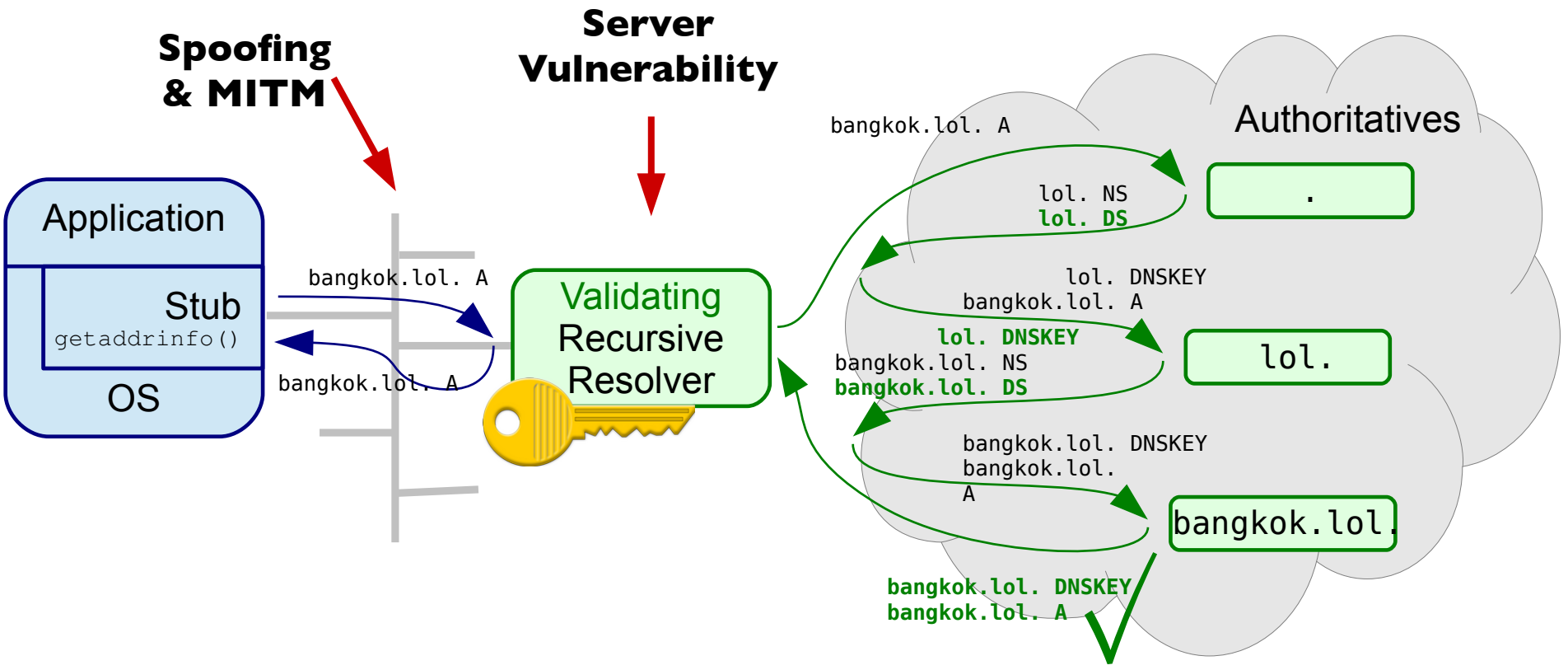
- Zones with distributed authority
- Chain of trust follows delegations

- DNSKEY Public key of zone
- DS Hash of DNSKEY signed by parent



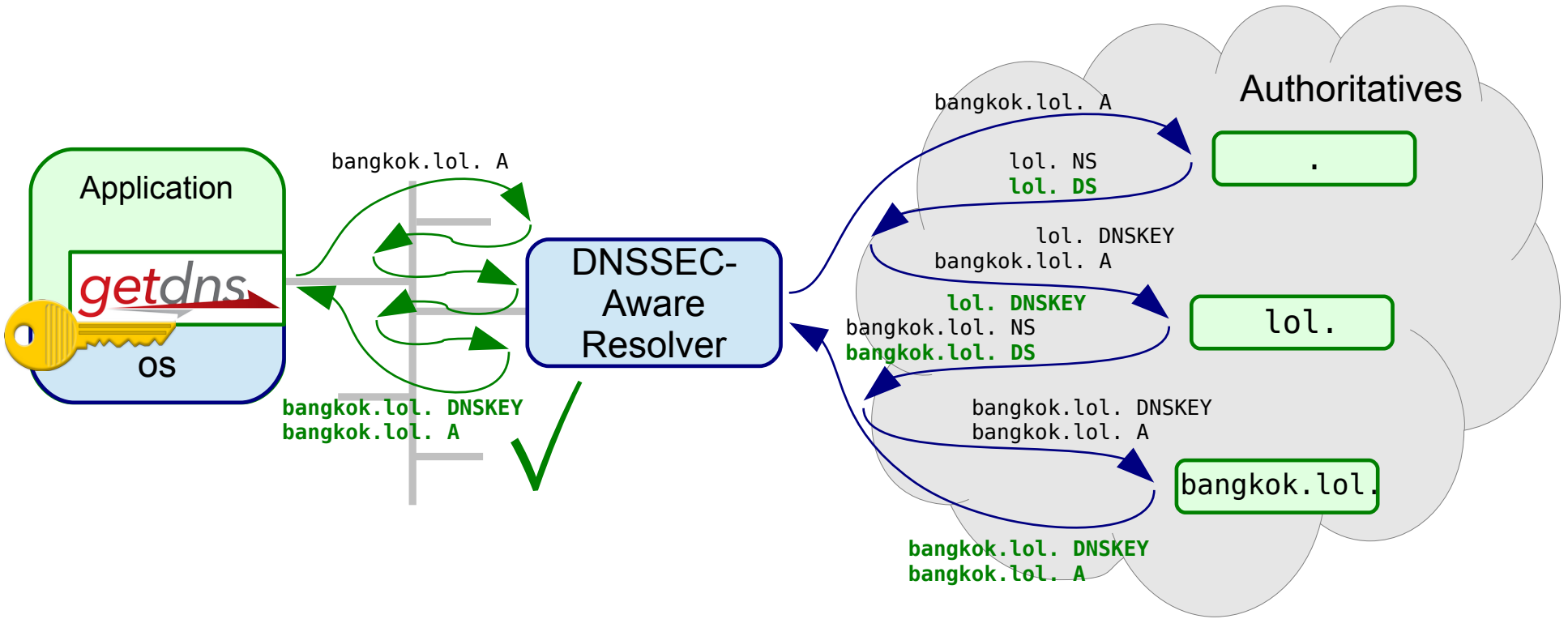
DNS Security Extensions (DNSSEC)

Validation



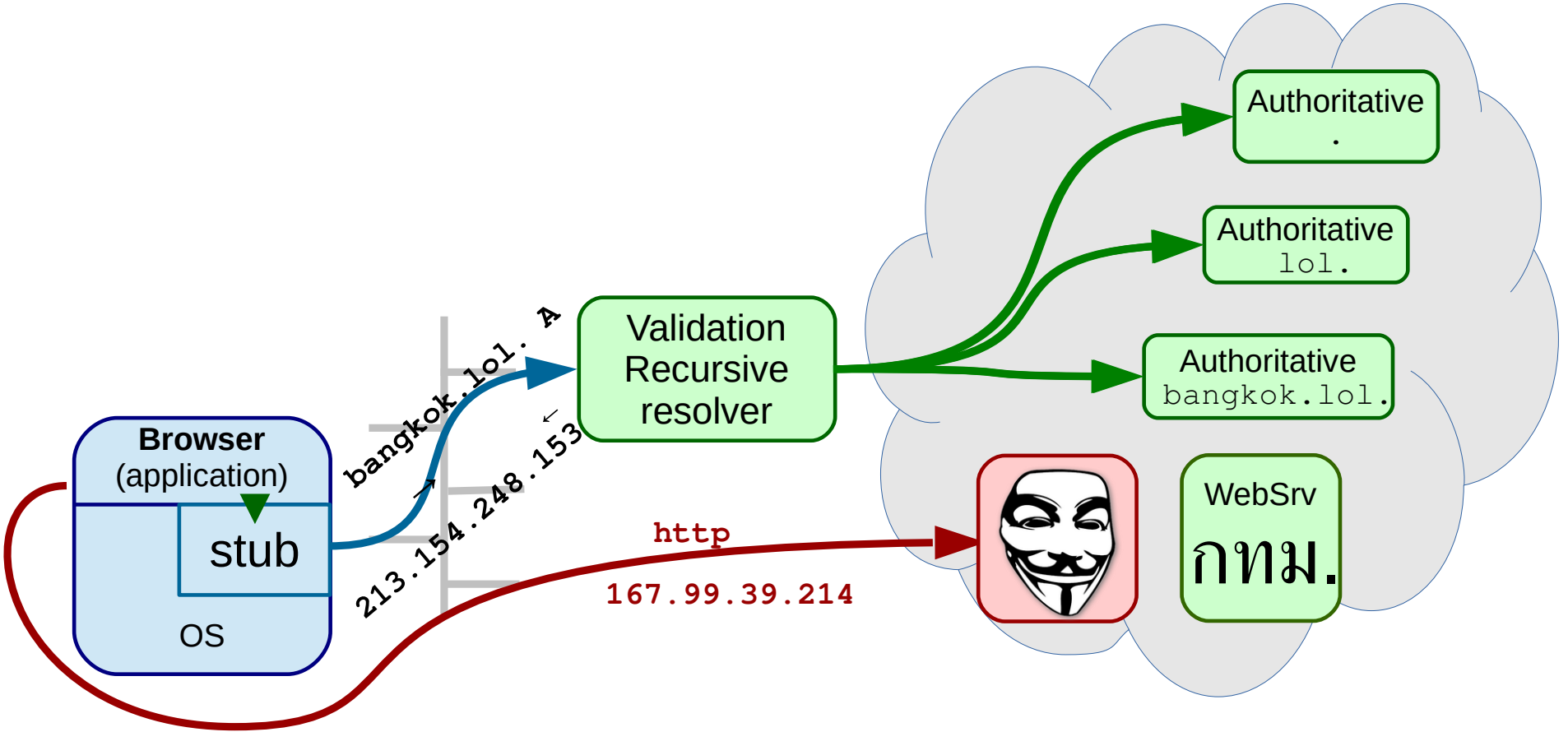
DNS Security Extensions (DNSSEC)

end-to-end validation



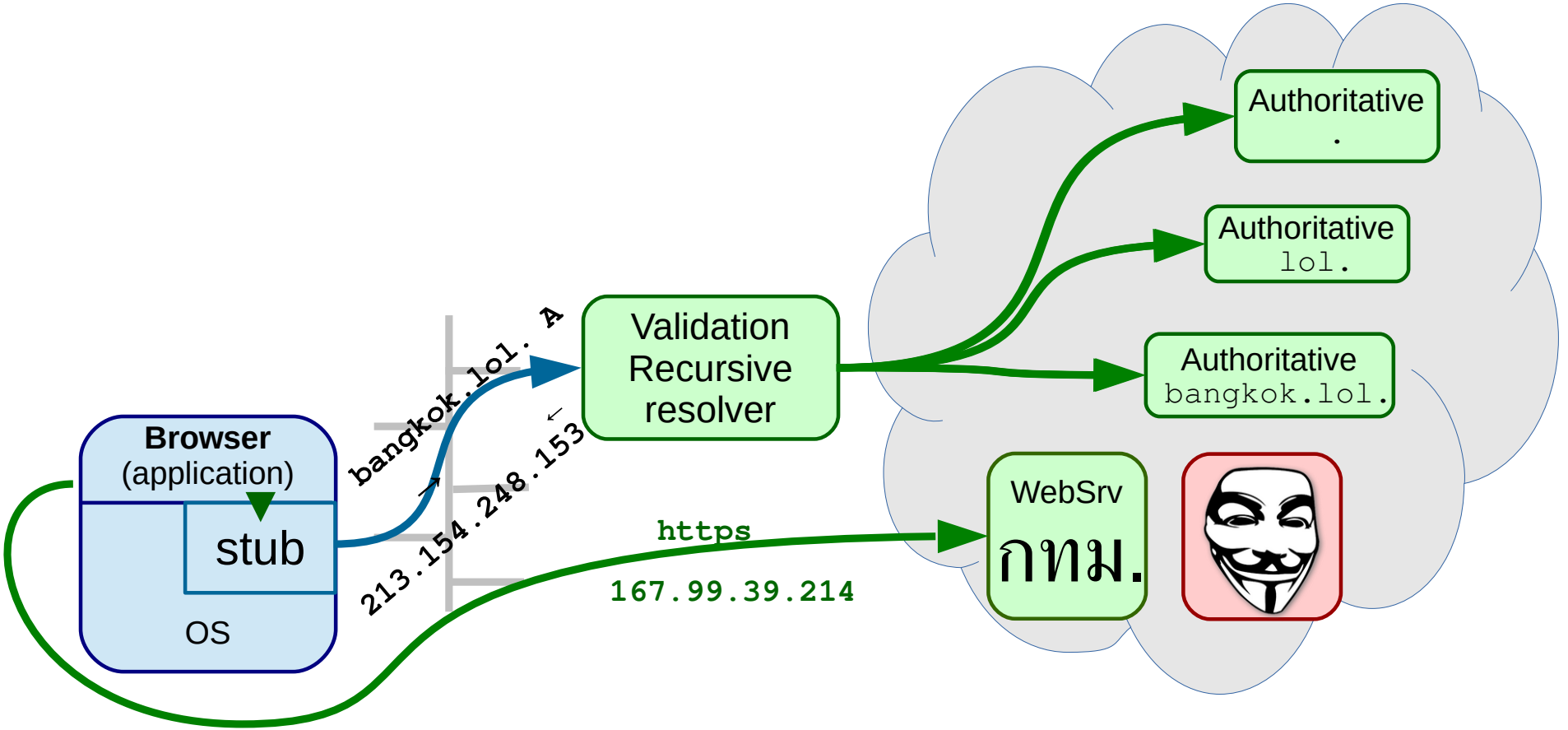
DNS Security Extensions (DNSSEC)

does not protect against MITM



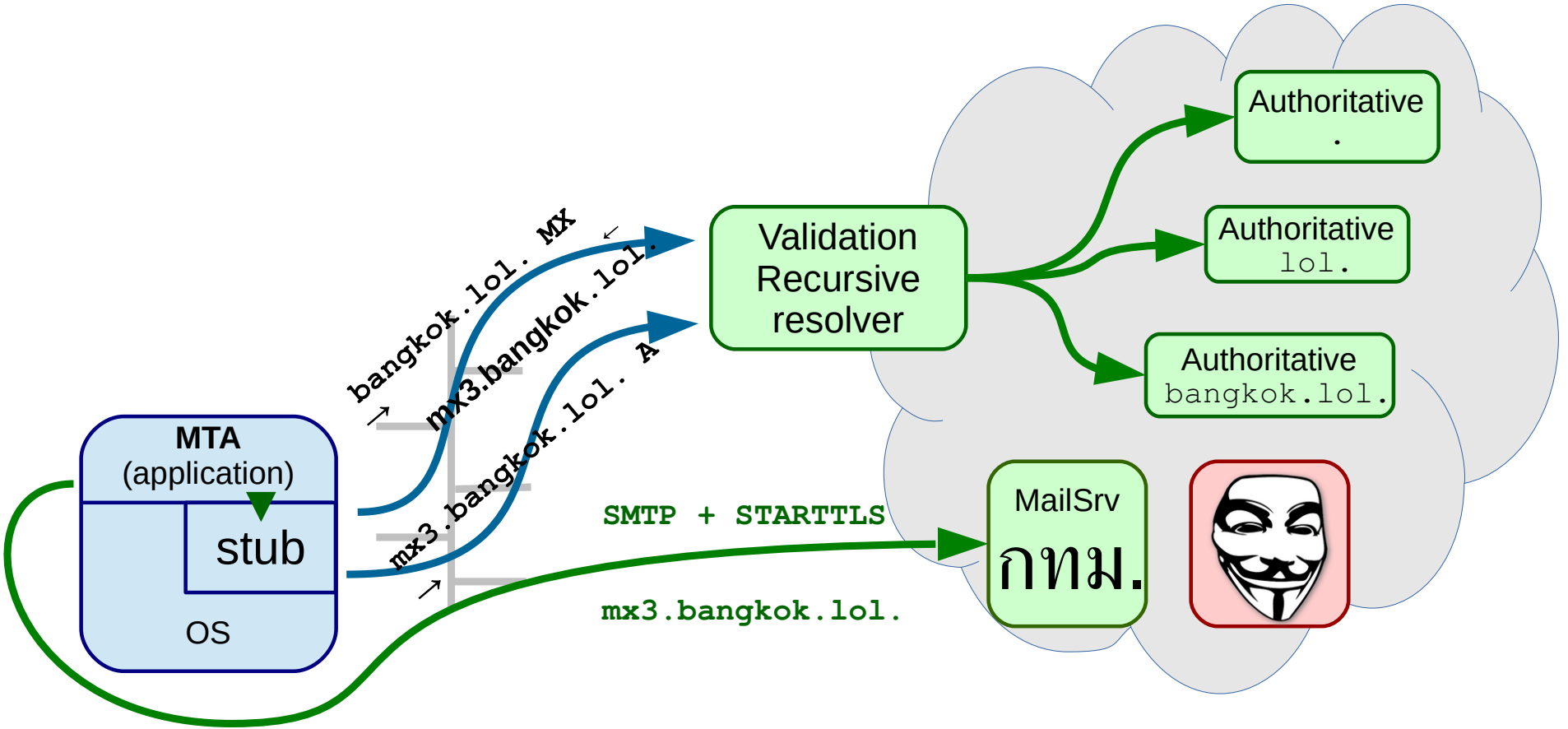
DNS Security Extensions (DNSSEC)

does not protect against MITM – TLS does!



DNS Security Extensions (DNSSEC)

still needed for referrals

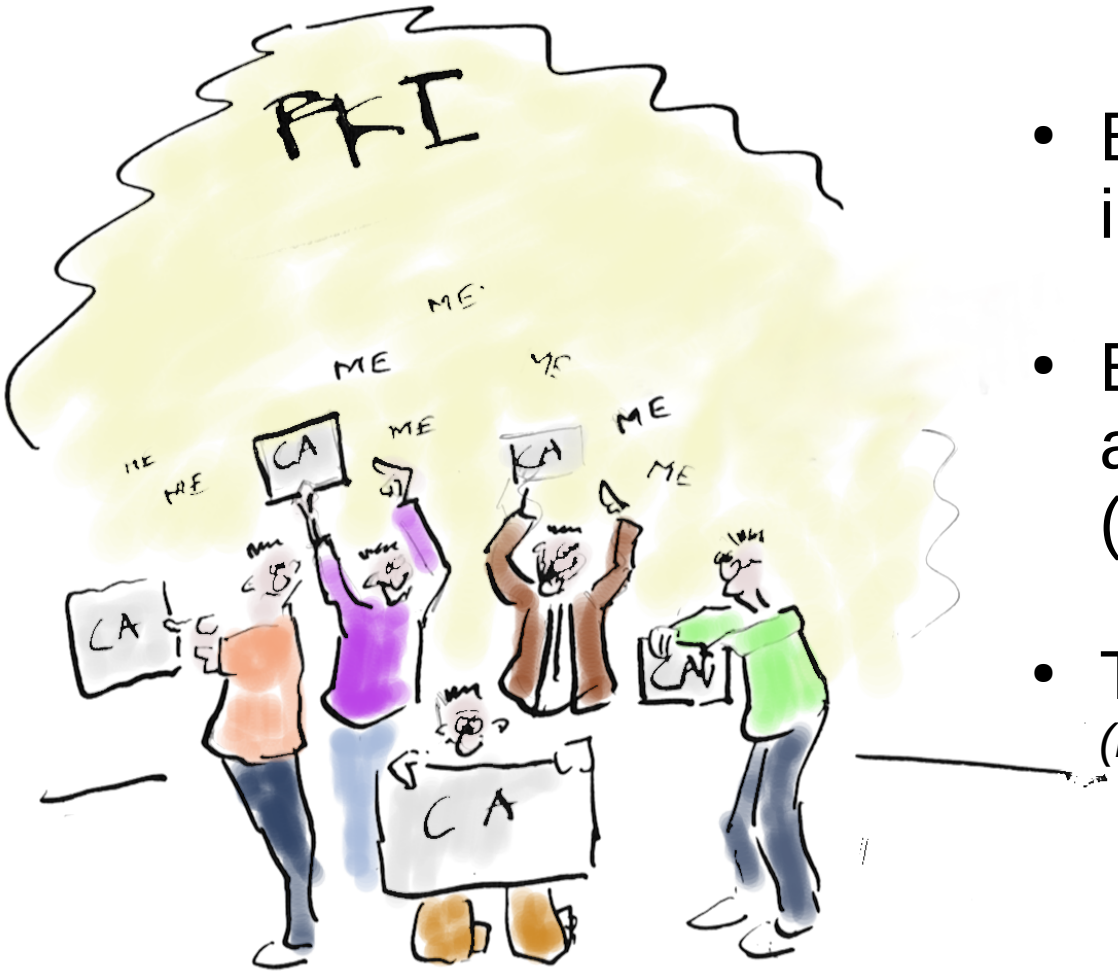


DNSSEC for Applications voor TLS

- Transport Layer Security (TLS) uses both asymmetric and symmetric encryption
- A symmetric key is sent encrypted with remote public key

- How is the remote public key authenticated?

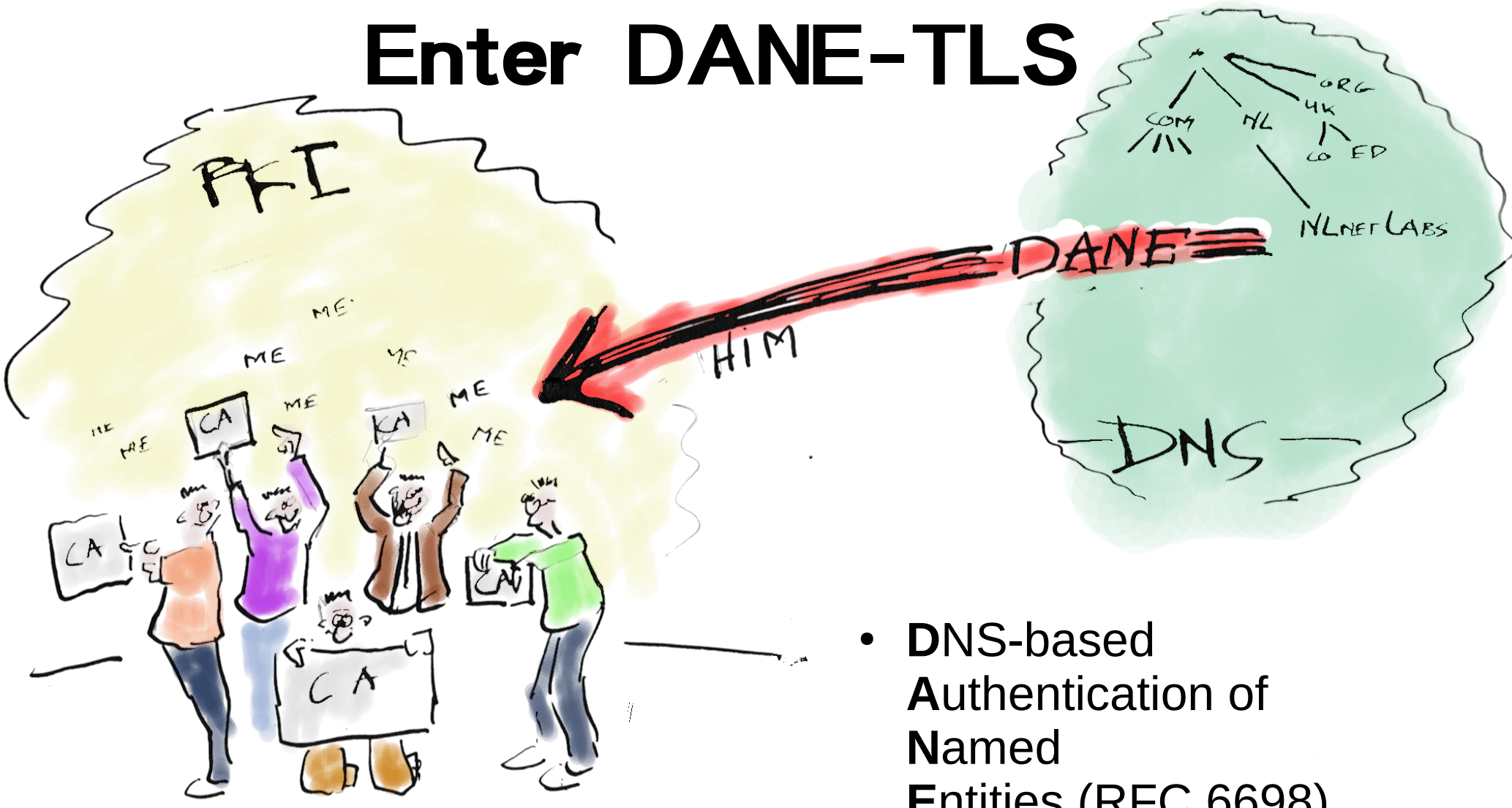
TLS without DNSSEC



Cartoon by Kloot

- By the Certificate Authorities in OS and/or browser
- Each CA is authorized to authenticate for **any** name (weakest link problem)
- There are more than 1500 CAs
  (in 2010, see https://www.eff.org/observatory)

Enter DANE-TLS



Cartoon by Kloot

- **D**NS-based **A**uthentication of **N**amed **E**ntities (RFC 6698)

Lab time!

- Hands on: http://bangkok.lol/
- 5. Turning your recursive resolver into a validating resolver

Photo by Alex Kondratiev on Unsplash