

ຍິນດີຕ້ອນຮັບສູ່

CYBER SECURITY WORKSHOP

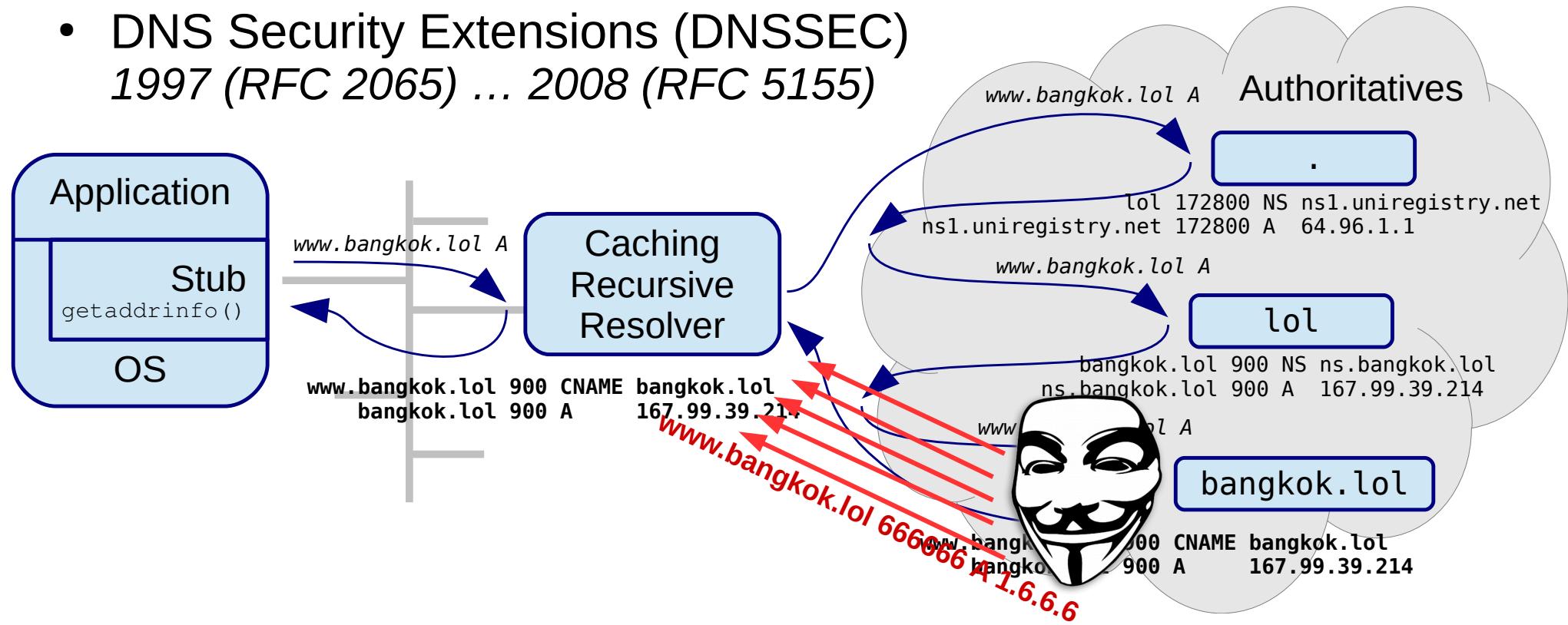


Vulnerabilities

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 * 64 bits)
- **Caching** als sso mechanism
- DNS Security Extensions (DNSSEC) RFC 5155
- DNS Security Extensions (DNSSEC) RFC 5155

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

Application

Stub
getaddrinfo()

Cache
Recursive
Resolver

www.bangkok.lol
A 167.99.39.214

Security
Popstar

lol 172800 NS ns1.uniregistry.net
ns1.uniregistry.net 172800 A 64.96.1.1

www.bangkok.lol A

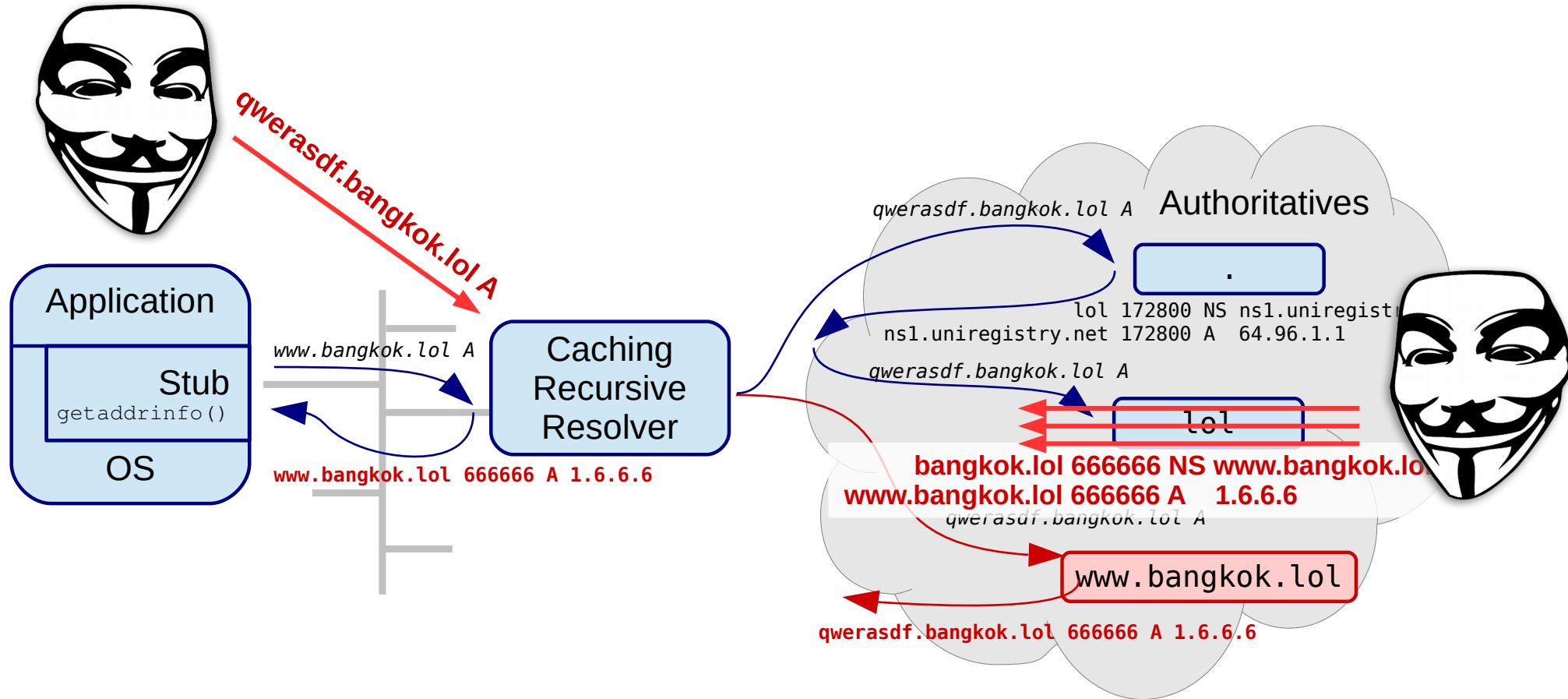
lol

bangkok.lol 900 NS ns.bangkok.lol
ns.bangkok.lol 900 A 167.99.39.214

bangkok.lol

www.bangkok.lol 900 A 167.99.39.214
bangkok.lol 900 CNAME bangkok.lol
www.bangkok.lol 666666 A 1.6.6.6

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 blocking* attacks on standard DNS resolvers, by *off-path*, *adversaries*. Our attacks exploit large DNS messages that cause IP fragmentation; such long requests are increasingly common, mainly due to the use

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 refer to this type of attack as *Domain hijacking*. DNS poi-

Security
Rockstar

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				Frag Offset				IP Header Checksum				IP Header				ICMP Header						
4	32	IPID				x DF MF				Frag Offset				IP Header Checksum				IP Header				IP Header				ICMP Header							
8	64	TTL				Protocol = 1				Source IP = 6.6.6.6				IP Header Checksum				IP Header Checksum				IP Header				ICMP Header							
12	96	Destination IP = 2.2.2.2				Protocol = 17				Source IP = 2.2.2.2				IP Header Checksum				IP Header Checksum				IP Header				ICMP Header							
16	128	Type = 3				Code = 4				ICMP Checksum				ICMP Checksum				ICMP Checksum				IP Header				ICMP Header							
20	160	Unused				MTU = 100				ICMP Checksum				ICMP Checksum				ICMP Checksum				IP Header				ICMP Header							
24	192	v4				IHL = 20				TOS				Total Length = 76				Total Length = 76				IP Header				ICMP Header							
28	224	IPID				x DF MF				Frag Offset				IP Header Checksum				IP Header Checksum				IP Header				ICMP Header							
32	256	TTL				Protocol = 17				Source IP = 2.2.2.2				IP Header Checksum				IP Header Checksum				IP Header				ICMP Header							
36	288	Destination IP = 7.7.7.7				Code = 4				ICMP Checksum				ICMP Checksum				ICMP Checksum				IP Header				ICMP Header							
40	320	Source Port = 53				Destination Port = 12345				Destination Port = 12345				Destination Port = 12345				Destination Port = 12345				IP Header				ICMP Header							
44	352	Length = 56				UDP Checksum = 0				UDP Checksum = 0				UDP Checksum = 0				UDP Checksum = 0				IP Header				ICMP Header							
48	384	Length = 56				Length = 56				Length = 56				Length = 56				Length = 56				IP Header				ICMP Header							
52	416	Length = 56				Length = 56				Length = 56				Length = 56				Length = 56				IP Header				ICMP Header							

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 refer to this type of attack as *Domain hijacking*. DNS poi-

ent practical
uses on stand-
adversaries.
that cause
increasingly common, mainly due to the use
in scenarios, where DNSSEC is partially or

Security
Rockstar

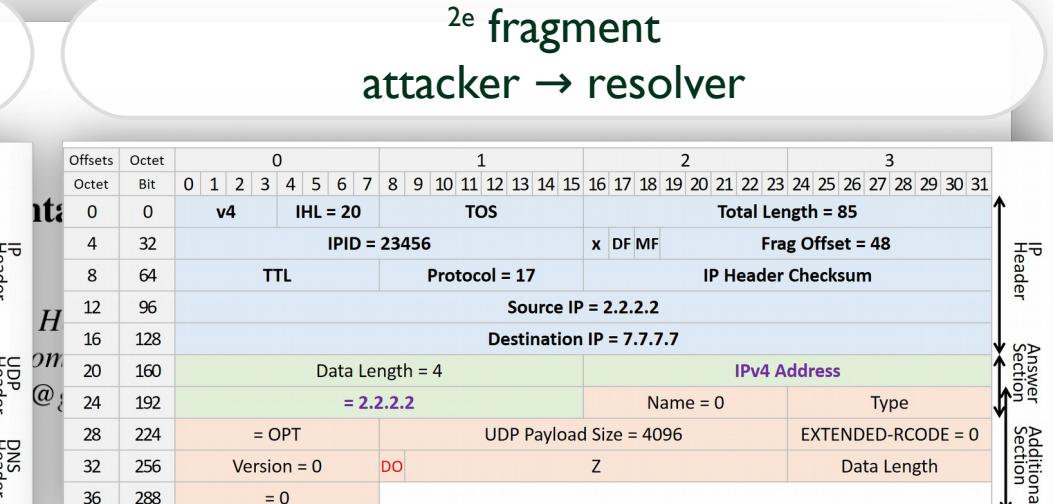
Domain Name System - security

- Help with spoofing DNS responses

1^e fragment

authoritative → resolver

Offsets	Octet	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
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																					
4	32	IPID = 23456								x DF MF	Frag Offset = 0																						
8	64	TTL				Protocol = 17				IP Header Checksum																							
12	96	Source IP = 2.2.2.2																															
16	128	Destination IP = 7.7.7.7																															
20	160	Source Port = 53								Destination Port = 12345																							
24	192	Length = 65								UDP Checksum = 0x14de																							
28	224	TXID = 76543								QR	Opcode = 0	AA	TC	RD	RA	Z	RCODE = 0																
32	256	Question Count = 1								Answer Record Count = 1																							
36	288	Authority Record Count = 0								Additional Record Count = 1																							
40	320	4	m				a				i																						
44	352	l	4				v				i																						
48	384	c	t				2				i																						
52	416	m	0				Type = A																										
56	448	Class = IN								Name (Pointer)																							
60	480	Type = A								Class = IN																							
64	512	TTL																															



server blocks by off-path, it large DNS such long re due to the use

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 refer to this type of attack as Domain hijacking/DNS poi

Domain Name System - security

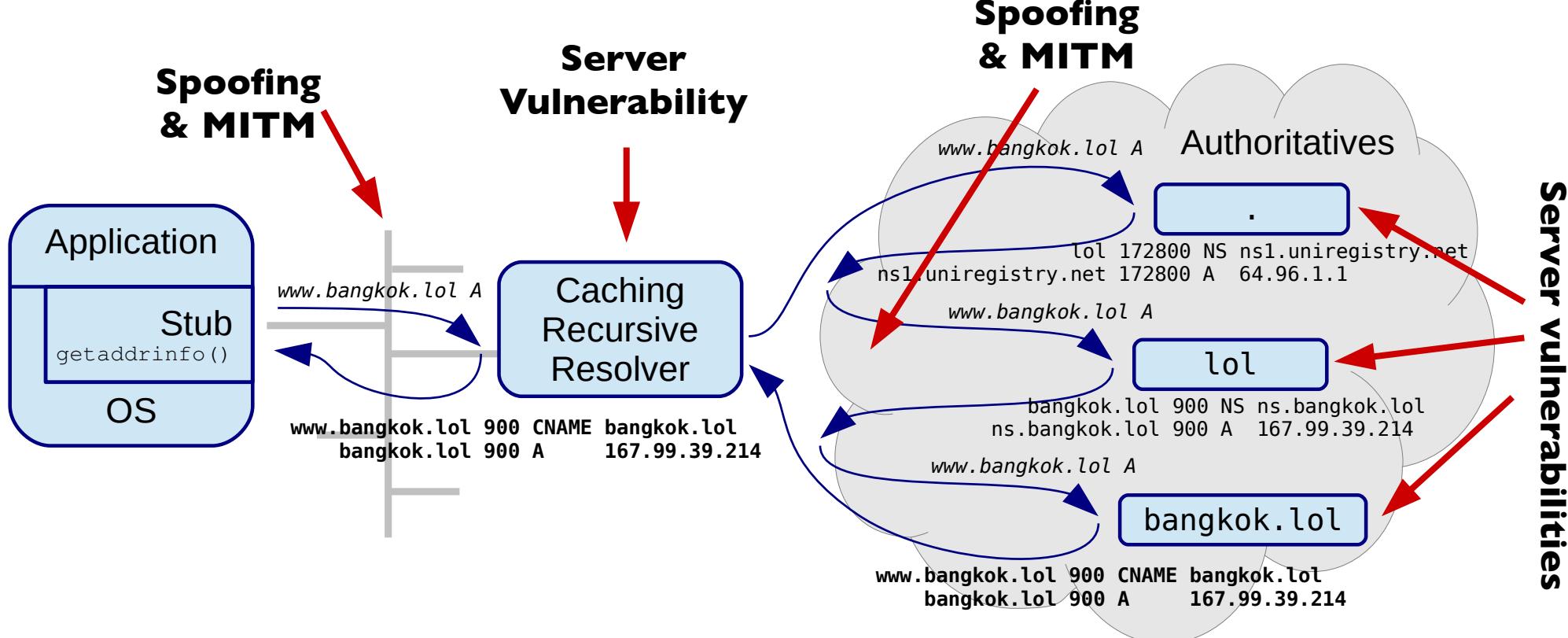
bits	50% chance	5% chance	Method
16	10 seconds	1 seconde	Query ID
26	2,8 uur	17 minutes	1024 source ports
2	0 seconds	0 seconds	All source ports 2 bits server selection
44	288444 days	2844.4 days	0x20 hack
5	0 seconds	0 seconds	IP ID

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

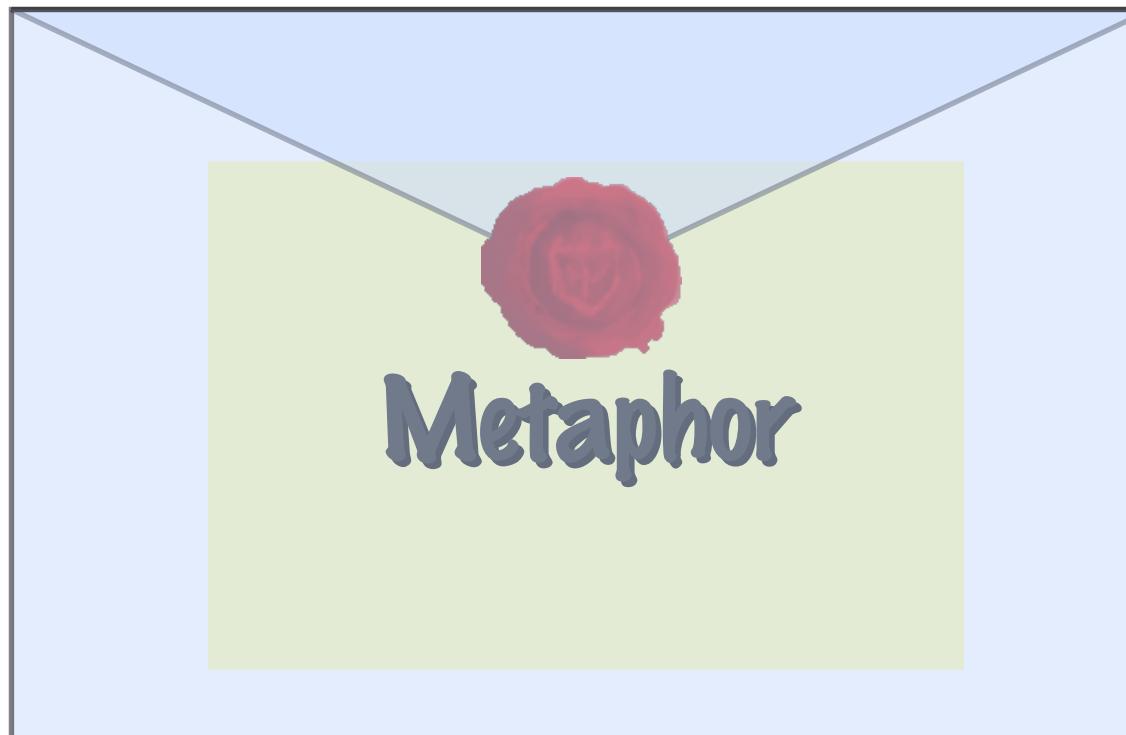
Domain Name System - security

- It's not just spoofing



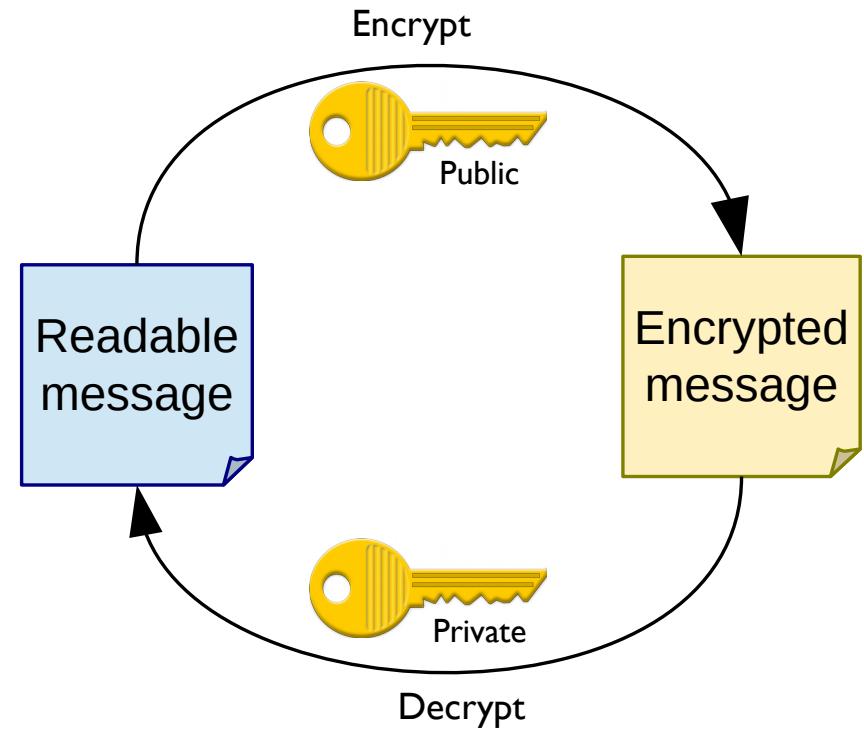
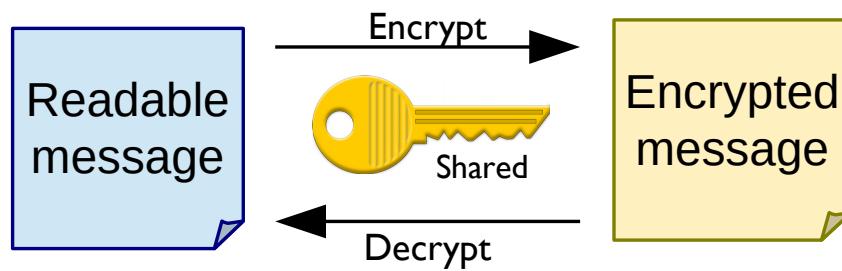
DNS Security Extensions (DNSSEC)

- end-to-end security on top of DNS



DNS Security Extensions (DNSSEC)

Refresher Public Key Crypto

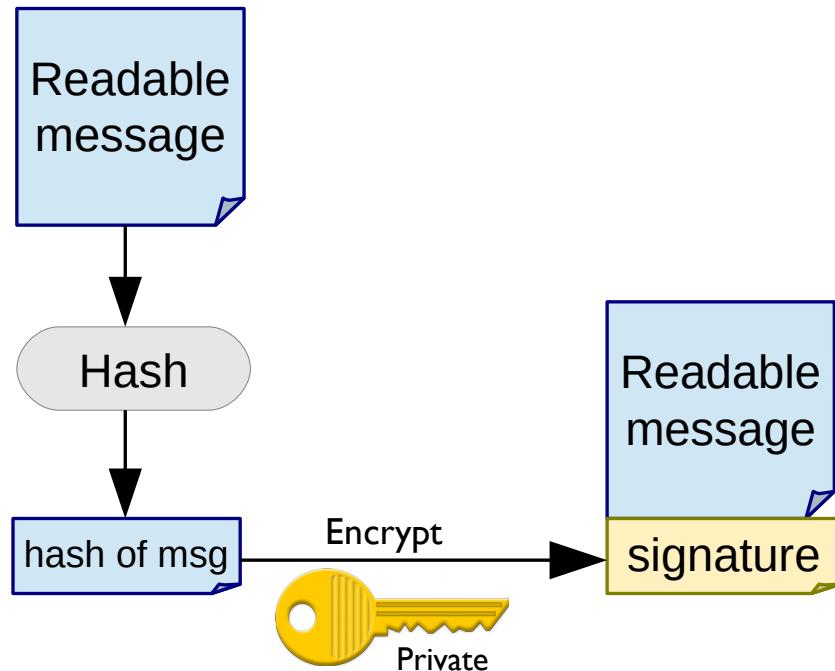


- Symmetric encryption

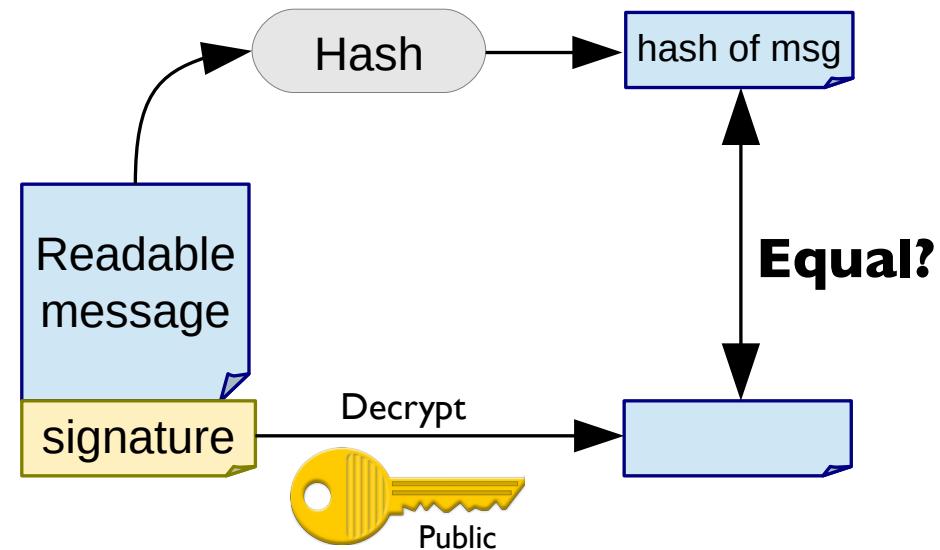
- Asymmetric encryption

DNS Security Extensions (DNSSEC)

Public Key Crypto – Signing



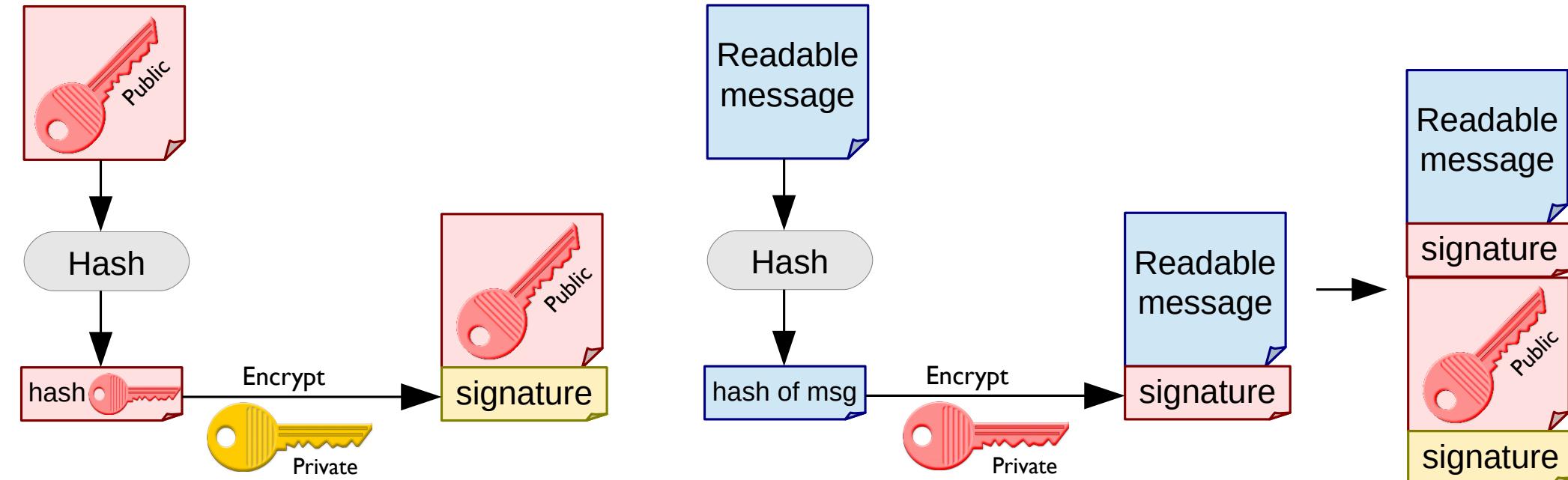
- Create signature



- Verify signature

DNS Security Extensions (DNSSEC)

Public Key Crypto – Delegating Authority



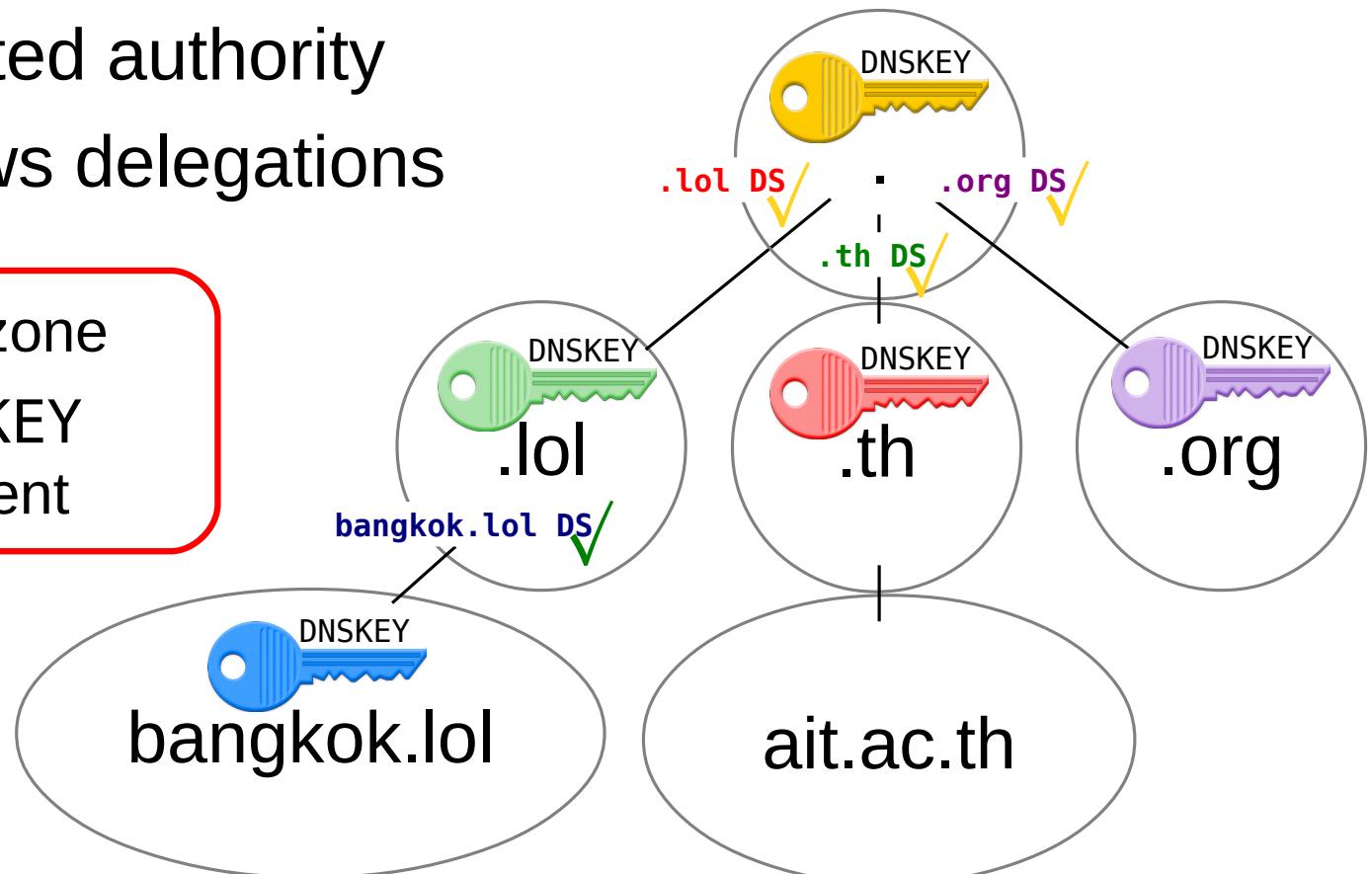
- Building the chain of trust authorizes
- signs the message

DNS Security Extensions (DNSSEC)

Chain of Trust

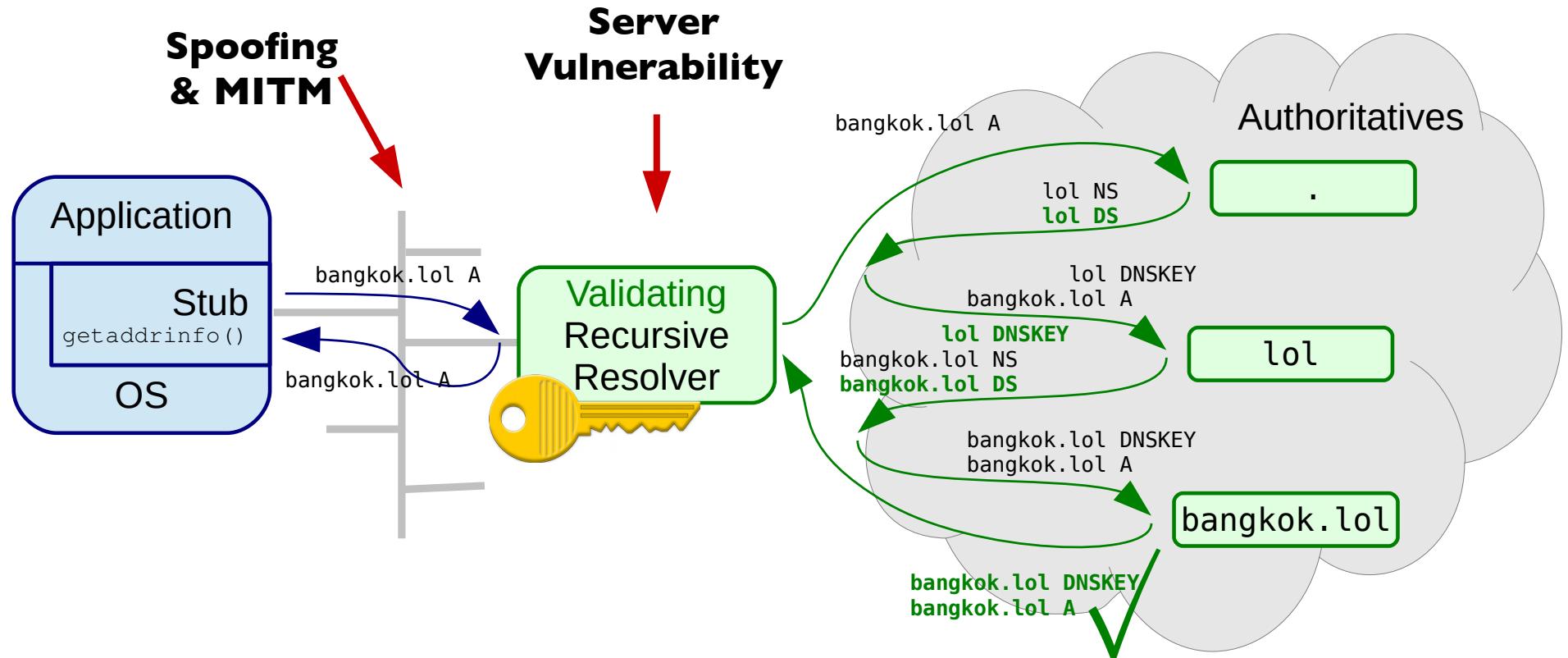
- Zones with distributed authority
- Chain of trust follows delegations

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



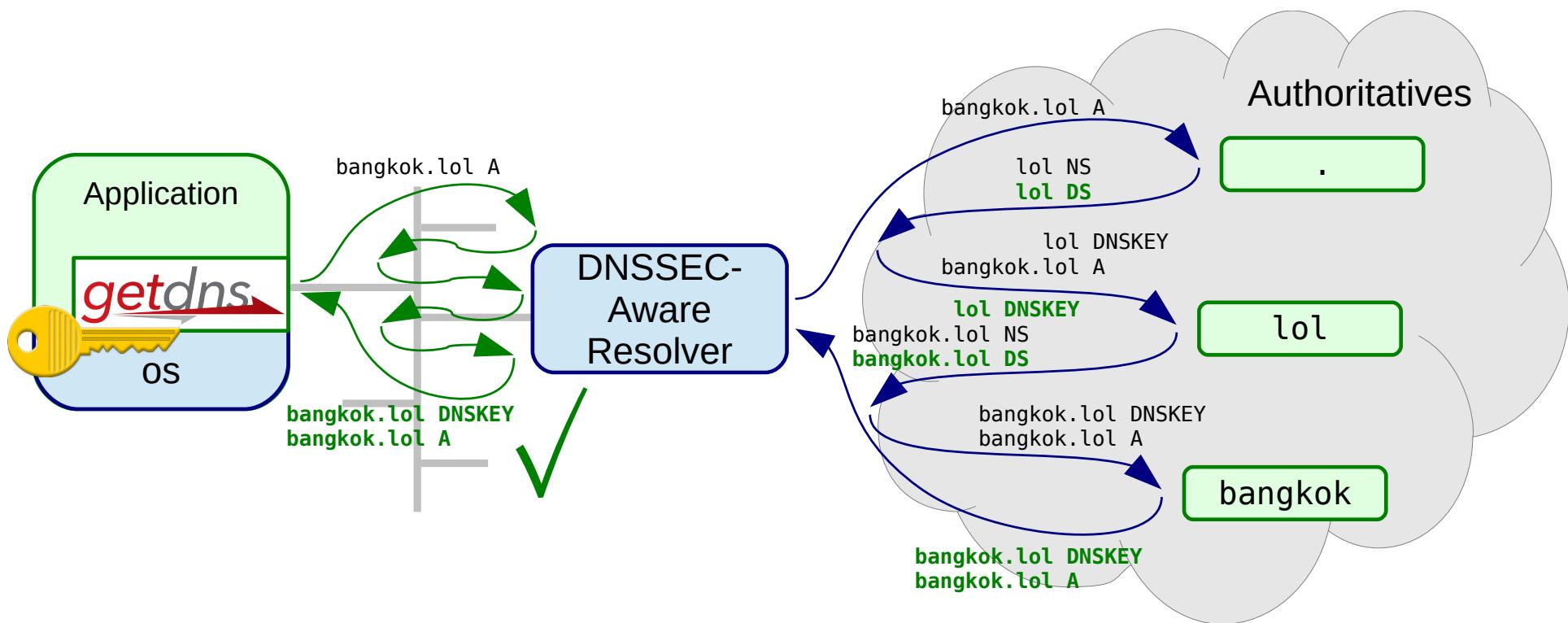
DNS Security Extensions (DNSSEC)

Validation



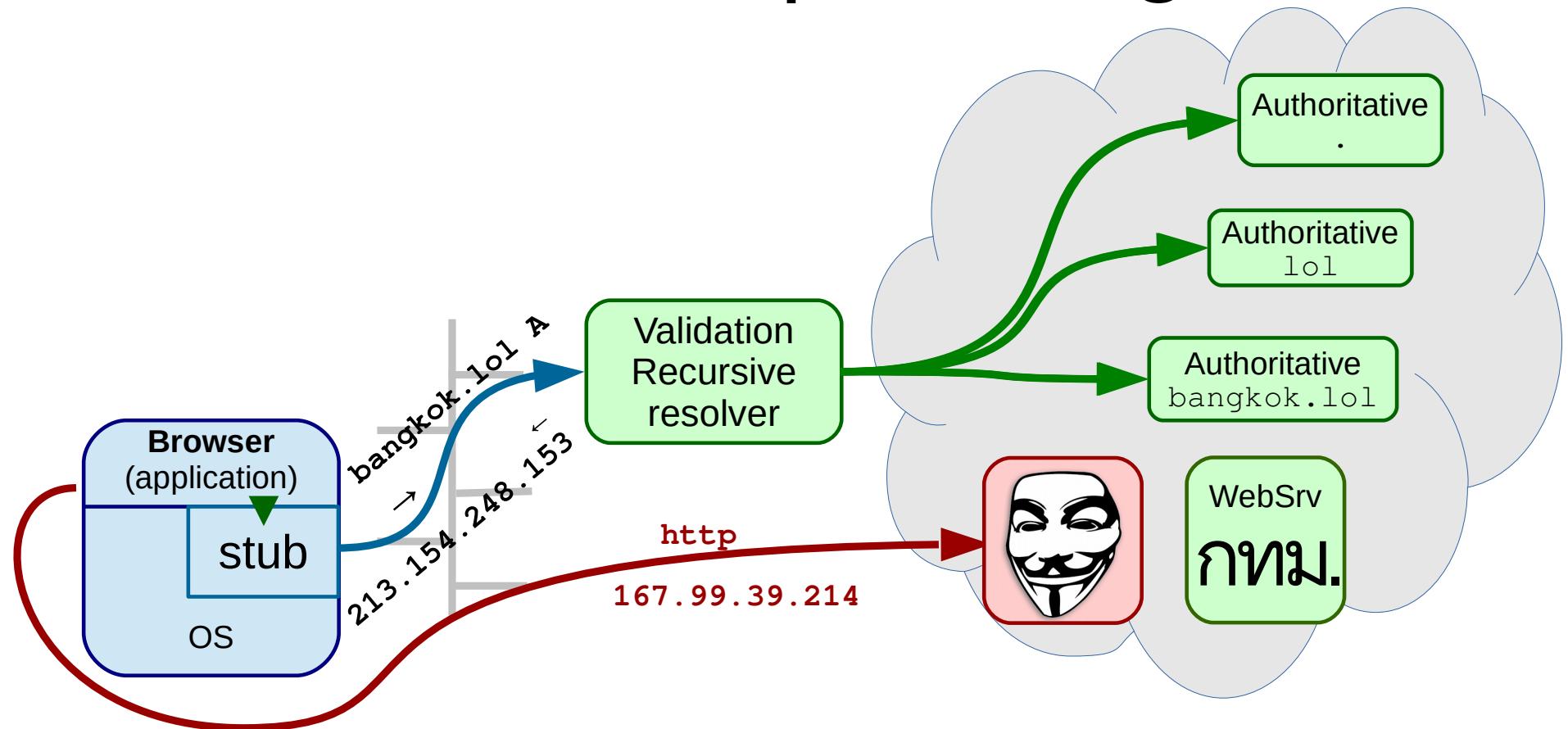
DNS Security Extensions (DNSSEC)

end-to-end validation



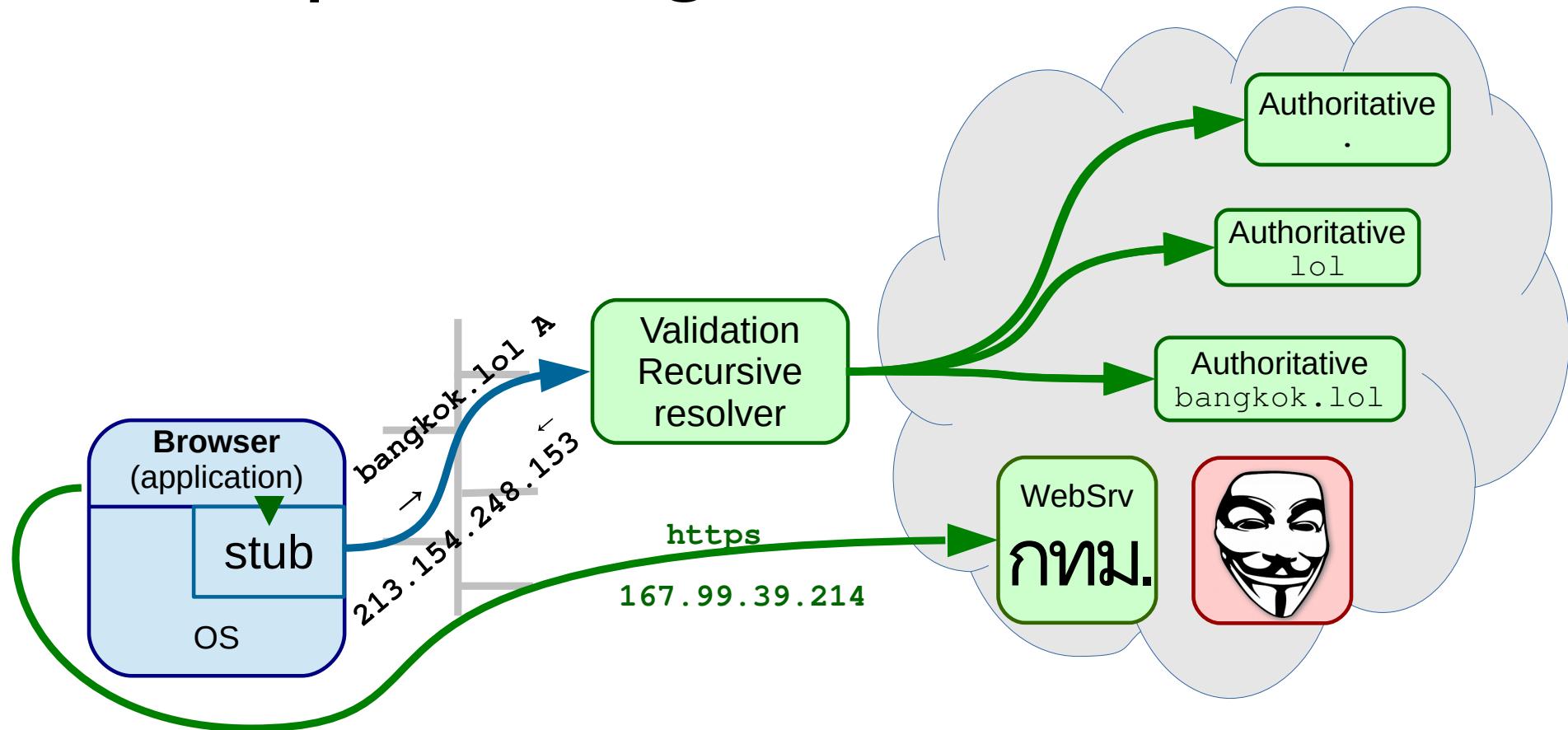
DNS Security Extensions (DNSSEC)

does not protect against MITM



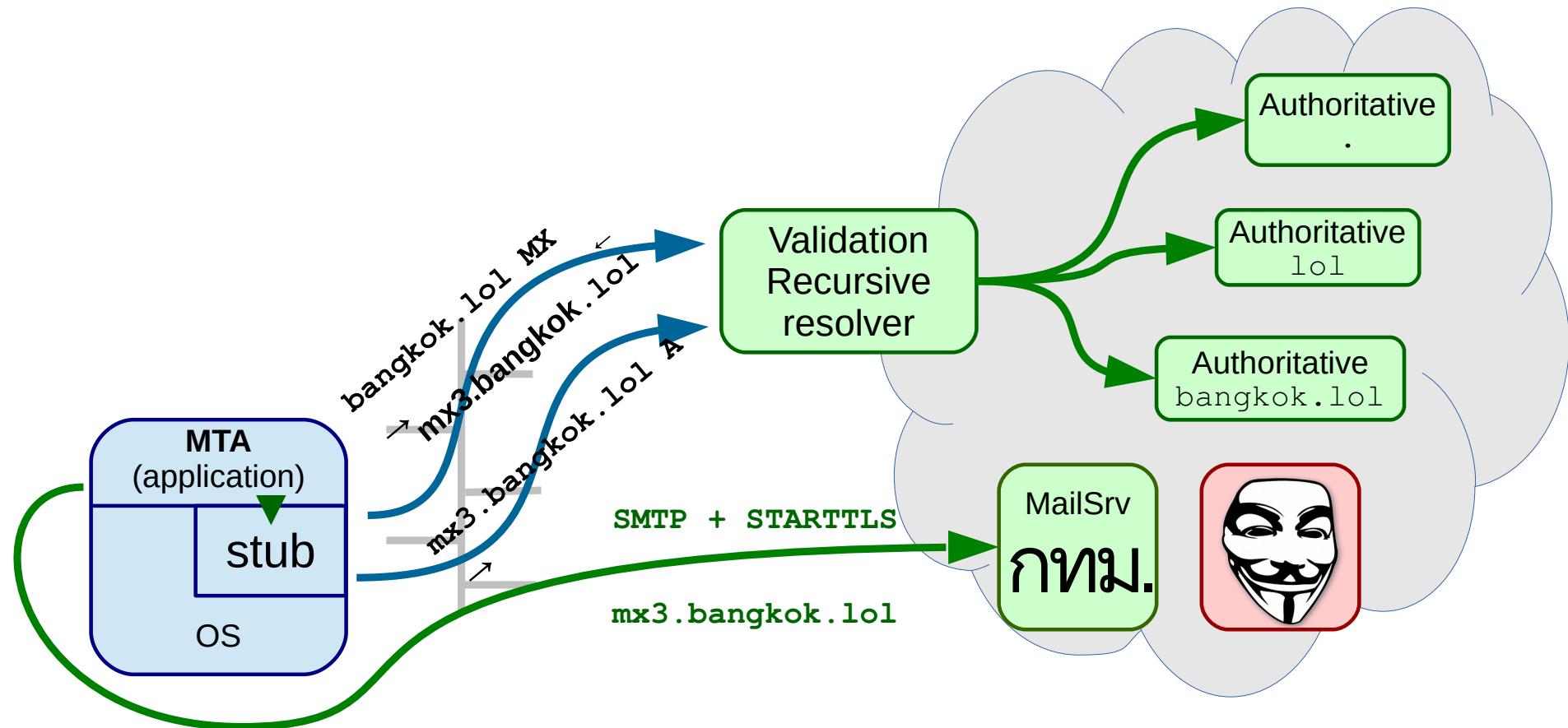
DNS Security Extensions (DNSSEC)

does not protect against MITM – TLS does!



DNS Security Extensions (DNSSEC)

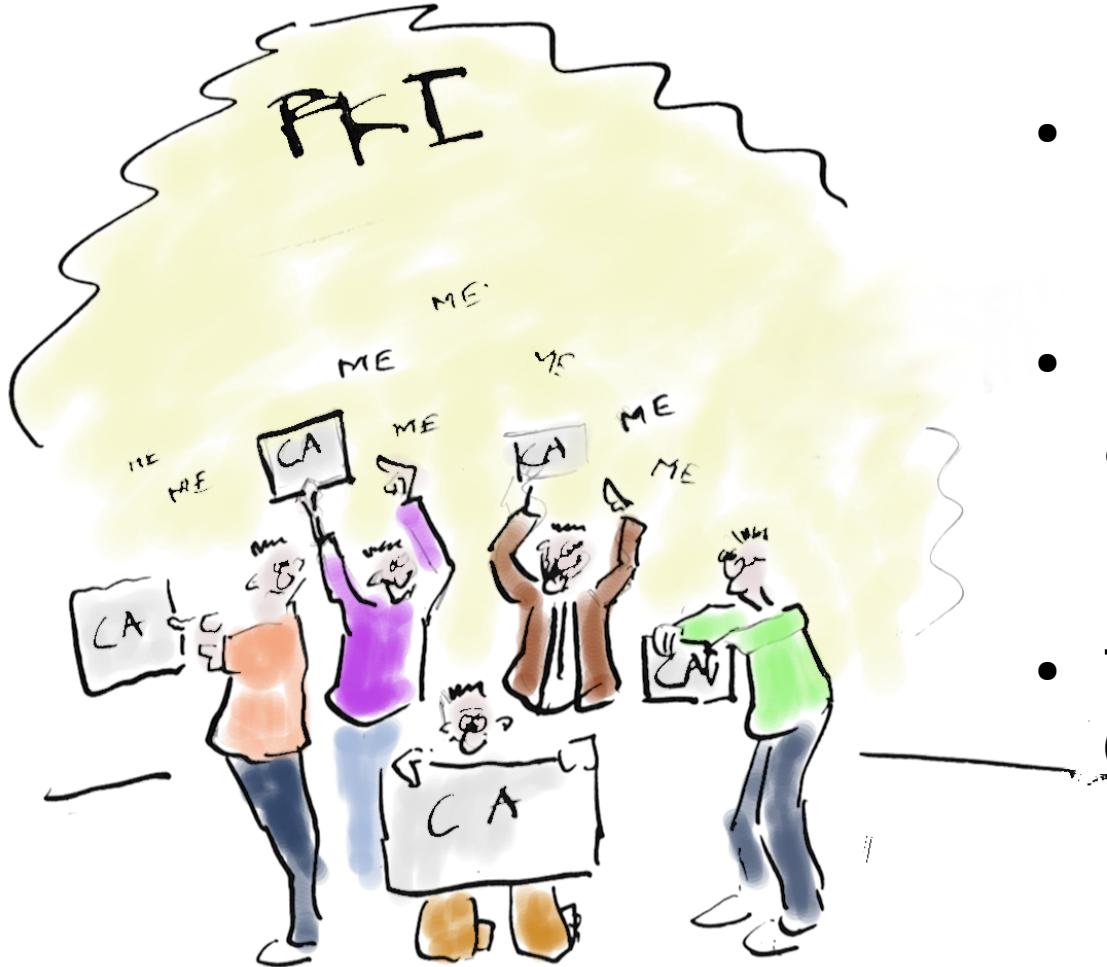
still needed for referrals



DNSSEC for Applications voor TLS

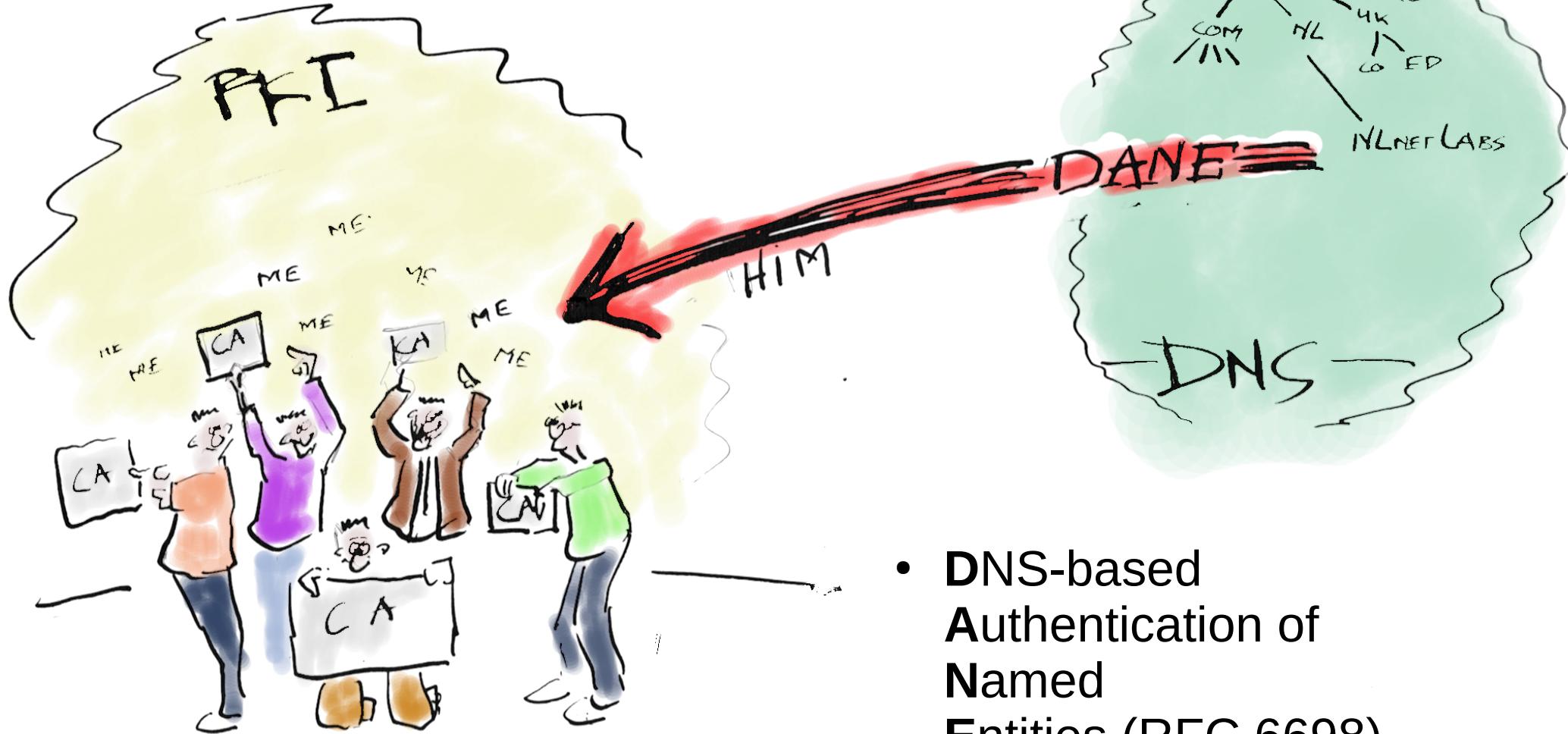
- Transport Layer Security (TLS) uses both asymmetric and symmetric encryption
- A symmetric key is sent encrypted with remote public key
- How is the remote public key authenticated?

TLS without DNSSEC



- 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



- DNS-based Authentication of Named Entities (RFC 6698)

Lab time!



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