Agentúra
Ministerstva školstva, vedy, výskumu a športu SR
pre štrukturálne fondy EÚ

Operačný program
VZDELÁVANIE

Európska únia
Európsky sociálny fond

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

PRÍPRAVA ŠTÚDIA MATEMATIKY A INFORMATIKY NA FMFI UK V ANGLICKOM JAZYKU

ITMS: 26140230008

DOPYTOVO – ORIENTOVANÝ PROJEKT

Moderné vzdelávanie pre vedomostnú spoločnosť/Projekt je spolufinancovaný zo zdrojov EÚ

DNSSEC

Martin Stanek

Department of Computer Science
Comenius University
stanek@dcs.fmph.uniba.sk

Security of IT infrastructure (2014/15)

Content

DNS (Domain Name System)

- hierarchical, distributed, decentralized system
- maps domain names to IP addresses and vice versa
- tree structure (zones, delegation of responsibilities)
  - 13 root "servers" (a, b, …, m) – clusters
  - example: "c" (operated by Cogent Communications) has 8 sites, 1 in Bratislava; usual traffic ≤ 40.000 queries/s (V/2015, c.root-servers.org)
- client – server architecture
- resource record types (RR – resource record):

SOA	start of authority
A	address
AAAA	IPv6 address
NS	name server
MX	mail exchange
CNAME	canonical name (alias) …

DNS (2)

- DNS servers:
  - authoritative – for own zone and for nameservers of delegated subzones
  - recursive – query other DNS servers to answer client requests

    . . . + caching functionality – remembering resolved records (efficiency)
- DNS resolvers (clients)
  - part of an operating system or application (e.g. web browsers)
  - usually include caching functionality
- Protocol (usually):
  - UDP, port 53
  - stateless (query – response)
  - no confidentiality, integrity or authenticity features

DNS – some security problems (1)

- DNS spoofing / cache poisoning (client, mail server etc. redirected to fake IP address)
  - attacker answers to client before his (recursive) DNS server
  - attacker answers to DNS server before the authoritative DNS server
  - attacker changes the response of the DNS server
  - attacker inserts additional information (e.g. NS records) about other zone into his zone's response, …
  - long TTL for fake IP addresses
- some ideas how to make spoofing harder:
  - RFC 5452 Measures for Making DNS More Resilient against Forged Answers
- weaknesses in protocol design, issues in DNS server implementations (e.g. vulnerabilities in bind (NVD): 2012/6, 2013/5, 2014/5, 2015 (I-IV)/1)

DNS – some security problems (2)

- DNS amplification attack
  - DNS server's responses can be much longer than the queries
  - source IP address spoofing
  - publicly open DNS servers . . . DDoS attack
  - counting open resolvers:

    openresolverproject.org ~ 16 mil. (V/2015)

    dnsscan.shadowserver.org ~ 6 mil. (V/2015)
- possible mitigations of amplification attack:
  - Disabling Recursion on Authoritative Name Servers
  - Limiting Recursion to Authorized Clients
  - Response Rate Limiting

DNS – some security problems (3)

- DNS rebinding
  - target the same-origin policy in a web browser
  - the first response of attacker's DNS server with short TTL
  - web browser loads malicious code
  - following question is answered by sending internal IP address
  - …attacking internal web
- mitigation of DNS rebinding:
  - DNS pinning – locking IP address to value from the first DNS response
  - filter out the private IP addresses from DNS responses, etc.
- Threat Analysis of the Domain Name System (DNS), RFC 3833

DNS – few (cryptographic) solutions to security problems

- TSIG
  - Transaction Signatures (RFC 2845)
  - HMAC-MD5 and shared secrets for authentication
  - primary use for dynamic updates of DNS records, zone transfers etc.
  - no means how to manage/distribute shared secrets
- SIG(0)
  - DNS Request and Transaction Signatures (SIG(0)s), RFC 2931
  - digital signatures for dynamic DNS updates
  - public key part of the zone
- DNSSEC
  - *today's theme …*

DNSSEC – introduction

- Domain Name System SECurity extensions

RFC 4033	DNS Security Introduction and Requirements
RFC 4034	Resource Records for the DNS Security Extensions
RFC 4035	Protocol Modifications for the DNS Security Extensions
RFC 5011	Automated Updates of DNSSEC Trust Anchors
RFC 5155	DNSSEC Hashed Authenticated Denial of Existence
…	

- Main idea: digitally signed DNS records (by DNS server)

  …DNSSEC itself is concerned with object security of DNS data, not channel security of DNS transactions.

- Goals:
  - data authenticity and integrity
  - authenticity of non-existent data

DNSSEC does not provide

- data confidentiality (no encryption)
  - no solution to privacy issues
- DoS attacks protection
  - against DNS server
  - against clients (DNS amplification)
  - DNSSEC can worsen the situation – signature verification, longer responses

Current state

- July 2010: DNS root zone signed
- 4th April 2012: (overall 313 TLDs in root zone / 91 signed)
- 17th April 2013: (317 TLDs in root zone / 111 signed)
- 9th May 2014: (571 TLDs in root zone / 386 signed)
- 4th May 2015: (923 TLDs in root zone / 752 signed)
- SK and neighbors:

signed zone with DS in root zone	cz., pl., at., ua., hu.
nothing	sk.

- com. zone signed in March 2011
  - current state – negligible(?) DNSSEC deployment in 2nd level domains
  - approx. 0.4% of domains in com. signed (www.statdns.com)
  - nothing: google.com, facebook.com, microsoft.com, apple.com ...
- Google Public DNS servers (8.8.8.8, 8.8.4.4) support DNSSEC validation by default since 2013

Keys a algorithms

- Key Signing Keys (KSK)
  - signing other keys in DNSKEY records
  - DS record needs to be published in parent zone (public key fingerprint)
- Zone Signing Keys (ZSK)
  - signing other records in the zone
  - simple management of ZSK (completely managed by the zone)
- the most frequent algorithms: RSA (usually 2048 bits) with SHA-256 or SHA-1
  - digital signature scheme: RSASSA-PKCS1-v1.5 (PKCS #1)
  - RSA key length max. 4096 bits (min. 1024 for RSA/SHA-512)

New DNS record types – DNSKEY and DS

examples from the root zone

`http://www.internic.net/zones/root.zone`

- DNSKEY – public key

  ```
  . 172800 IN DNSKEY 256 3 8 AwEAAbd0IPTQdvy. . . Y6YJ
  . 172800 IN DNSKEY 257 3 8 AwEAAagAIKlVZrp. . . hz0=
  ```

  - . 172800 IN – owner, TTL, class
  - 256, 257 – flags (256: ZSK; 257: KSK, the last bit denotes SEP (Secure Entry Point))
  - 3 – protocol (fixed)
  - 8 – algorithm (RSA/SHA-256)
- DS (Delegation signer) – KSK identification (for delegated zone)

  ```
  GR. 86400 IN DS 35136 7 2 F7BF4C362. . . 5C3
  ```

  - 57519 – key tag (for fast selection of DNSKEY record)
  - 7 – algorithm corresponding to referenced DNSKEY record (RSA/SHA1/NSEC3)
  - 2 – hash function (SHA-256)

New DNS record types – DNSKEY and DS

examples from the root zone

`http://www.internic.net/zones/root.zone`

- DNSKEY – public key

  ```
  . 172800 IN DNSKEY 256 3 8 AwEAAbd0IPTQdvy. . . Y6YJ
  . 172800 IN DNSKEY 257 3 8 AwEAAagAIKlVZrp. . . hz0=
  ```

  - . 172800 IN – owner, TTL, class
  - 256, 257 – flags (256: ZSK; 257: KSK, the last bit denotes SEP (Secure Entry Point))
  - 3 – protocol (fixed)
  - 8 – algorithm (RSA/SHA-256)
- DS (Delegation signer) – KSK identification (for delegated zone)

  ```
  GR. 86400 IN DS 35136 7 2 F7BF4C362. . . 5C3
  ```

  - 57519 – key tag (for fast selection of DNSKEY record)
  - 7 – algorithm corresponding to referenced DNSKEY record (RSA/SHA1/NSEC3)
  - 2 – hash function (SHA-256)

Root zone

- Management of KSK and ZSK for the root zone:
  - DNSSEC Root Zone High Level Technical Architecture (Draft)
  - DNSSEC Practice Statement for the Root Zone KSK Operator
  - DNSSEC Practice Statement for the Root Zone ZSK Operator
  - Root Zone DNSSEC KSK Ceremonies Guide (Draft)
- Publishing KSK:
  - DNSSEC Trust Anchor Publication for the Root Zone
  - various formats (certificate, CSR, XML, p7s, pem, pgp, ...)
  - available via HTTP and HTTPS (`data.iana.org/root-anchors/`)

New DNS record types – RRSIG

- RRSIG – signature for a record set

  `. 518400 IN RRSIG NS 8 0 518400 20120411000000 20120403230000 56158 . RJ0ceR. . . 8XpzA=`

  - `. 518400 IN RRSIG` – owner, TTL, class, type
  - NS – type that the signature covers
  - 8 – signing algorithm (RSA/SHA-256)
  - 0 – the number of labels (used to validate $*$)
  - 518400 – original TTL value
  - 20120411000000, 20120403230000 – signature validity (until 11.04.2012 0:00 UTC, starting 03.04.2012 23:00 UTC)
  - 56158 – key tag of the key in DNSKEY record for signature verification
  - . – singer's name (the owner in the DNSKEY record)
  - and finally, the signature

RRSIG

- a record set is signed (RRset)
  - RRset is determined by shared attributes: owner, class, type
- some records are unsigned:
  - NS records of delegated zones
  - A, AAAA records of delegated zones
  - these are data of delegated zones (not their parent zone)

Authenticated denial of existence – NSEC

- How to answer that a record does not exist?
- we don't want to sign on-line (access to private key required, slow)
- sorted records (canonical order)
- NSEC – "next secure" record

```
cz. 86400 IN NSEC dabur. NS DS RRSIG NSEC
cz. 86400 IN RRSIG NSEC . . .
```

  - for particular domain name (`cz.`)
  - `dabur.` – next owner (domain name) in zone file
  - NS DS RRSIG NSEC – types of existing records for current owner/name (`cz.`)
- the last NSEC refers to the beginning (next owner)
- there is one RRSIG record for each NSEC record

Nonexistent record – response types

- NXDOMAIN (nonexistent domain name, e.g. `da.`)

```
$dig da. @8.8.8.8
...
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 4649
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ...
```

- NOERROR & ANSWER: 0 (the name exists but not the name with given type)

```
$dig nic.de. A @8.8.8.8 +short
81.91.170.12
$dig nic.de. AAAA @8.8.8.8 +short
$
```

Nonexistence responses using NSEC

- NXDOMAIN: in response (in authority section) – NSEC record with RRSIG, proving the missing domain name:
  `cz. 10574 IN NSEC dabur. NS DS RRSIG NSEC`
  - no name between `cz.` and `dabur.`
- NOERROR & ANSWER: 0: in response (in authority section) – NSEC record with RRSIG, proving the missing type for the domain name:

```
$dig @149.20.64.20 sk. DS +dnssec
...
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 7087
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ...
...
sk.        7540    IN       NSEC      sky. NS RRSIG NSEC
...
```

NSEC vs. NSEC3

- zone walking (domain names enumeration)
- nonexistent name (NXDOMAIN) leaks neighbors “above” and “below”
- possibility to enumerate the zone ($\sim$ zone transfer via NSEC)
  - number of queries approx. linear with respect to the number of records
- problem for DNSSEC deployment …solution: NSEC3

NSEC3

- replacement of NSEC records; domain names replaced with fingerprints

```
dig @149.20.64.20 p.bund.de A +dnssec
...
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38332
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 8, ...
...
EDE5...79LN.bund.de. 10800 IN NSEC3 1 0 10 DDD087
                     EEPT...CS3K A RRSIG
```

- record-chain ordered according fingerprint values
- resolver computes name's fingerprint and finds the value between fingerprints in NSEC3 record (in practice other NSEC3 and corresponding RRSIG)

NSEC3 (2)

- NSEC3 parameters

```
EDE5...79LN.bund.de. 10800 IN NSEC3 1 0 10 DDD087
                    EEPT...CS3K A RRSIG
```

  - 1 – hash function (SHA-1)
  - 0 – flags
  - 10 – iteration count for fingerprint calculation
  - DDD087 – salt
  - next name's fingerprint, record types for current owner
- off-line attack on fingerprints (the space of domain names is limited)

Differences DNSEC vs. PKI

- no certificates
- no validity interval for keys (but signatures have validity interval)
- keys managed by corresponding zone
- trusting a public key:
  - trusting KSK of the root zone (static import) → ZSK (.) → DS (in . for de.) → KSK (de.) → ZSK (de.) → DS (in de. for .bund.de.) → KSK (bund.de.) → ZSK (bund.de)

    …and then we can verify RRSIG of A record for www.bund.de
  - of course: … + caching