DNSSEC

Samuel Pélissier (samuel.pelissier@inria.fr)

2025

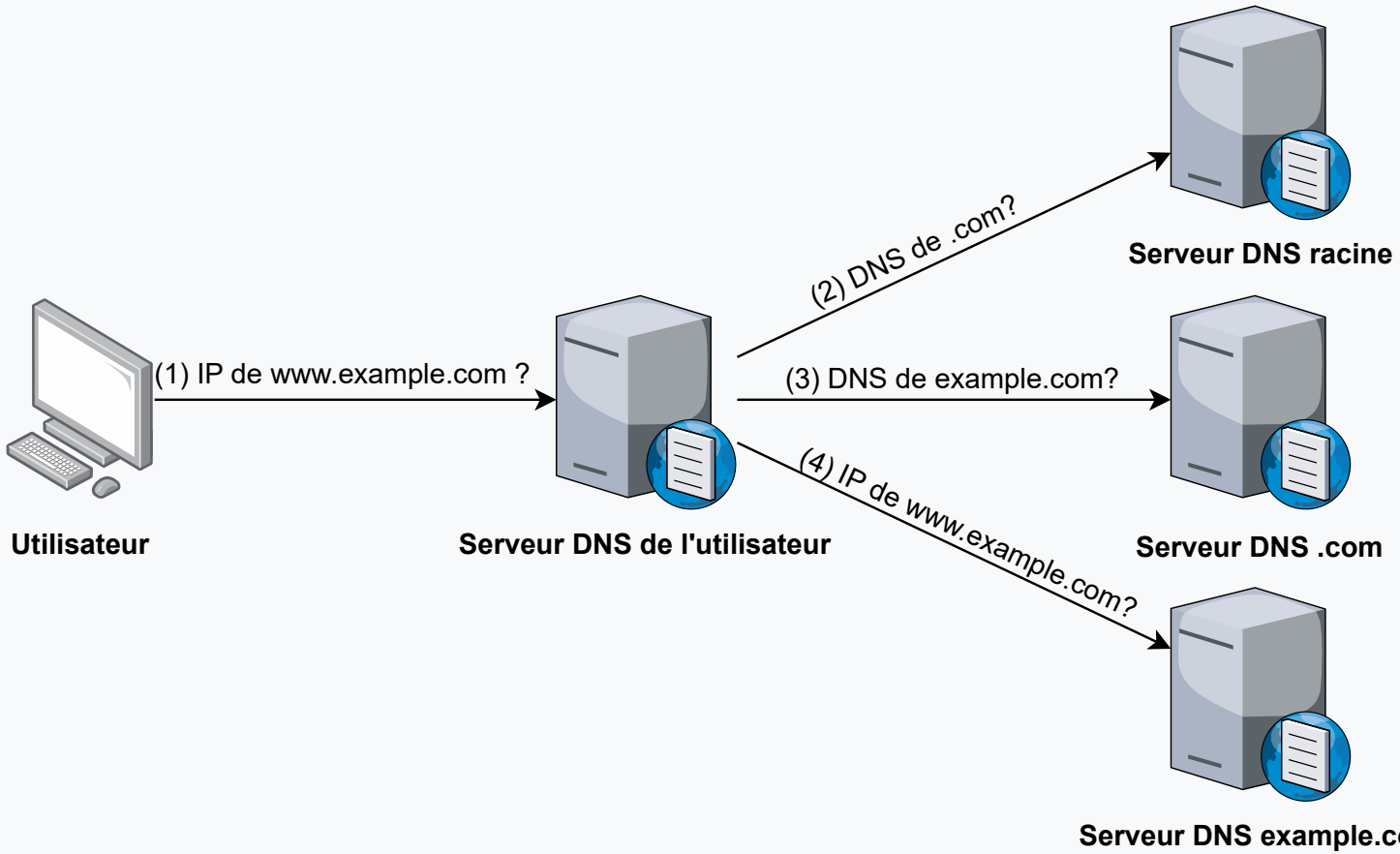
Ressources

- https://www.cloudflare.com/learning/dns/what-is-dns/
- https://www.bortzmeyer.org/search?pattern=dns
- http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

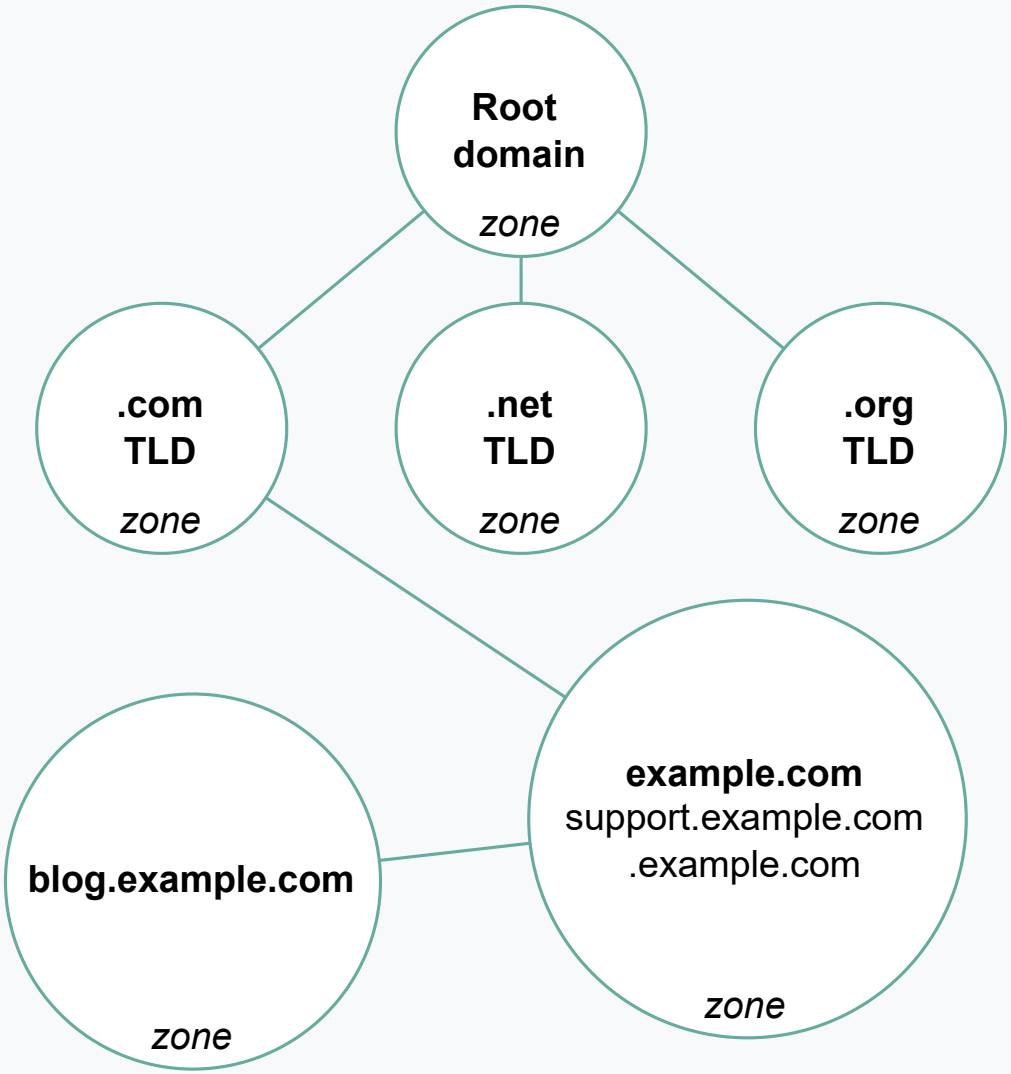
Rappels sur le DNS

Le rôle principal de DNS est :

- D'associer un nom à une adresse IP
- D'associer une adresse IP à un nom



Rappels sur le DNS



Les zones servent à déléguer l'autorité. Une zone peut contenir plusieurs sous-domaines. Plusieurs zones peuvent être administrées sur le même serveur.

Rappels sur le DNS

- Une architecture client-serveur
- 3 entités
  - Serveur ayant autorité sur une zone
  - Serveur cache
  - Résolveur

Rappels sur le DNS

```
$ORIGIN domain.com
$TTL 86400
@     IN     SOA    dns1.domain.com.     hostmaster.domain.com. (
                    2001062501 ; serial
                    21600      ; refresh after 6 hours
                    3600       ; retry after 1 hour
                    604800     ; expires after 1 week
                    86400 )    ; minimum TTL of 1 day

      IN     NS     dns1.domain.com.
      IN     NS     dns2.domain.com.

      IN     MX     10     mail.domain.com.
      IN     MX     20     mail2.domain.com.

             IN     A       10.0.1.5

server1      IN     A       10.0.1.5
server2      IN     A       10.0.1.7
dns1         IN     A       10.0.1.2
dns2         IN     A       10.0.1.3

ftp          IN     CNAME   server1
mail         IN     CNAME   server1
mail2        IN     CNAME   server2
www          IN     CNAME   server2
```

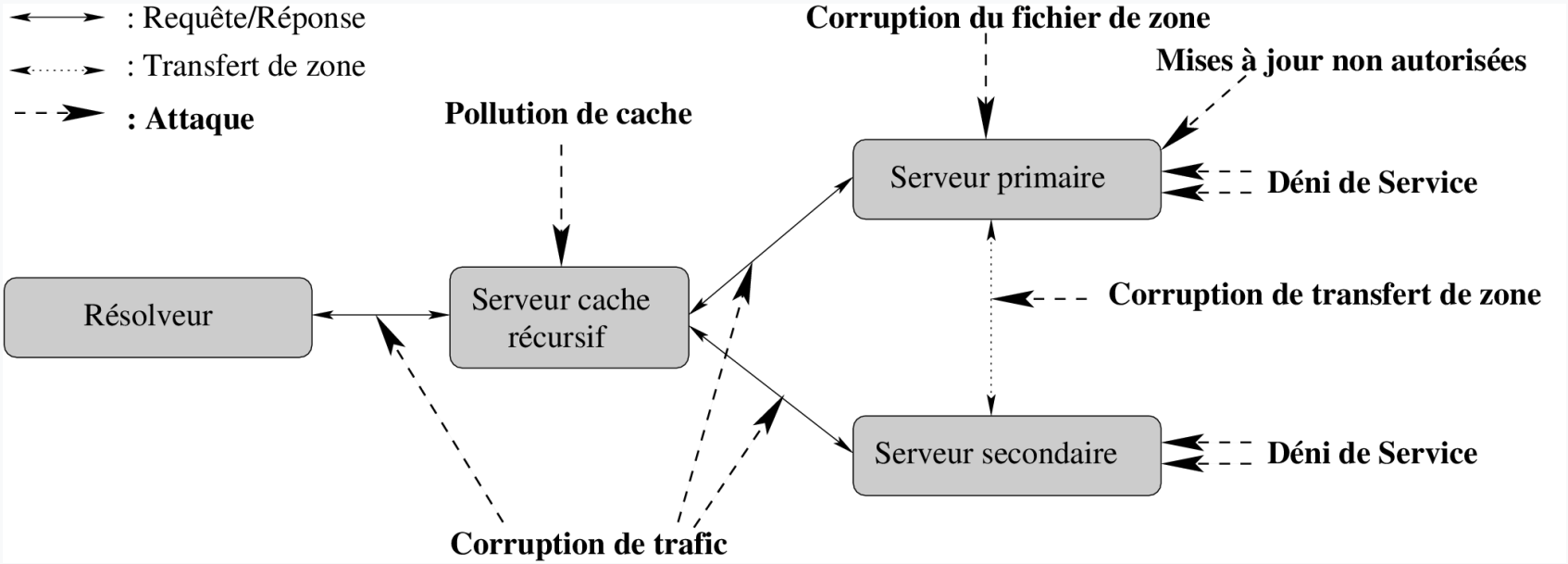
Rappels sur le DNS

```
$ORIGIN 1.0.10.in-addr.arpa
$TTL 86400
@       IN      SOA     dns1.domain.com.      hostmaster.domain.com. (
                        2001062501 ; serial
                        21600      ; refresh after 6 hours
                        3600       ; retry after 1 hour
                        604800     ; expire after 1 week
                        86400 )    ; minimum TTL of 1 day

        IN      NS      dns1.domain.com.
        IN      NS      dns2.domain.com.

20      IN      PTR     alice.domain.com.
21      IN      PTR     betty.domain.com.
22      IN      PTR     charlie.domain.com.
23      IN      PTR     doug.domain.com.
24      IN      PTR     ernest.domain.com.
25      IN      PTR     fanny.domain.com.
```

Services de sécurité dans le DNS : **AUCUN**

Les vulnérabilités du DNS

De multiples points d'attaques :



DNS poisoning

```
; <<>> DiG 9.7.3 <<>> www.univ-rennes1.fr
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45885
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;www.univ-rennes1.fr.           IN      A

;; ANSWER SECTION:
www.univ-rennes1.fr.    20202   IN      CNAME   frontalhttp.univ-rennes1.fr.
frontalhttp.univ-rennes1.fr. 17352 IN   A       129.20.126.100

;; AUTHORITY SECTION:
univ-rennes1.fr.        17377   IN      NS      soleil.uvsq.fr.
univ-rennes1.fr.        17377   IN      NS      resone.univ-rennes1.fr.

;; ADDITIONAL SECTION:
resone.univ-rennes1.fr. 17377   IN      A       129.20.254.1
soleil.uvsq.fr.         9441    IN      A       193.51.24.1

;; Query time: 5 msec
;; SERVER: 131.254.4.3#53(131.254.4.3)
;; WHEN: Thu Sep  1 11:24:35 2011
;; MSG SIZE  rcvd: 158
```

Comment s'assurer que le contenu est légitime ?

DNS poisoning

Un résolveur accepte la réponse si :

- Les couples adresse/port source/destination sont identiques à ceux de la question
- Les ID de la question et de la réponse sont identiques
- La section question de la requête et de la réponse sont identiques

DNS poisoning

Pour donner une fausse info à votre résolveur il suffit :

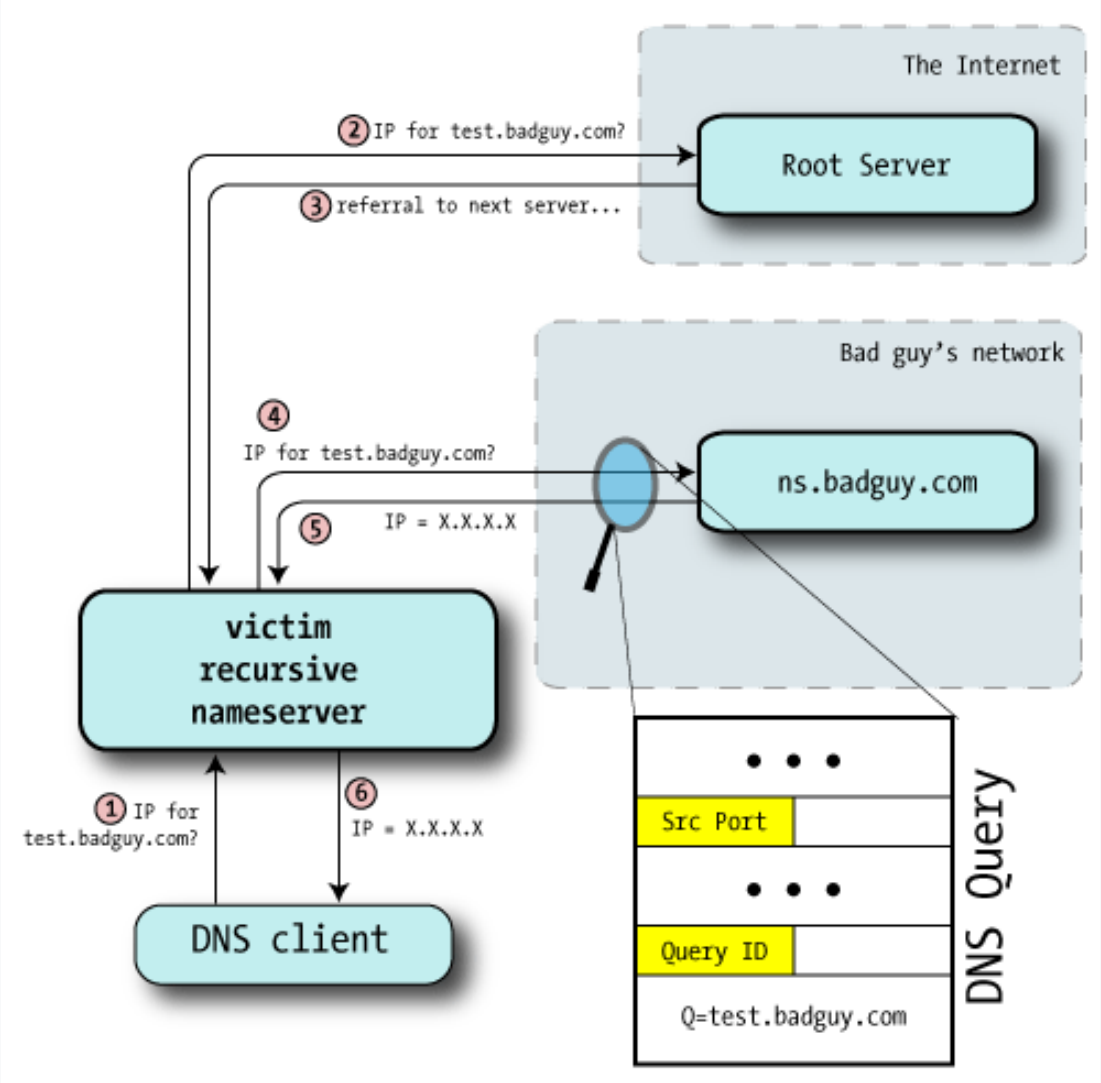
- De répondre plus vite que le « vrai » serveur
- De deviner l'ID et la question
- De deviner les adresses et ports source et destination

DNS poisoning

Est-ce compliqué ?

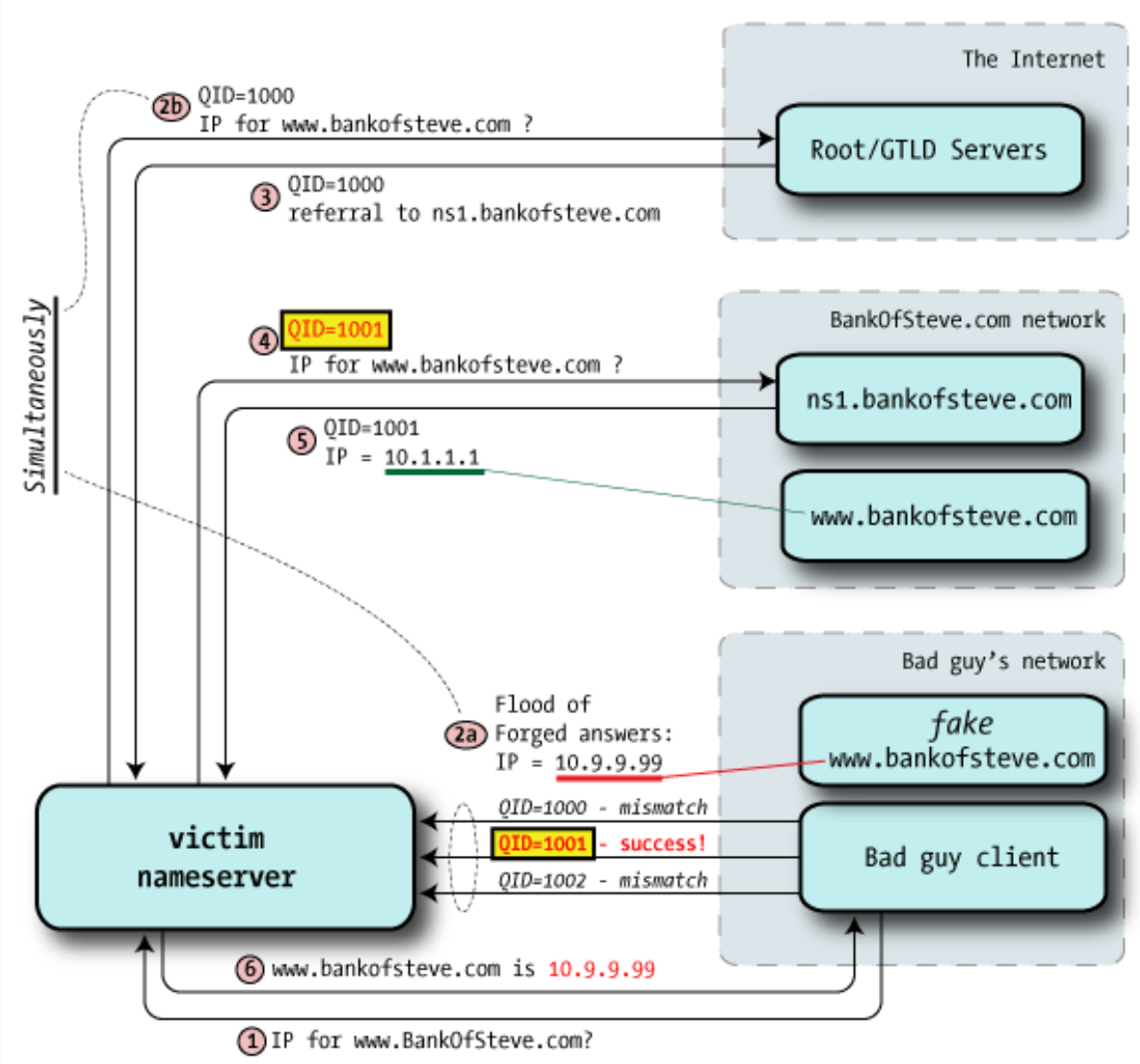
- Pas vraiment
- Une capture (MitM) donne toutes les infos nécessaires
- Si vous ne pouvez pas faire de capture, il y a le paradoxe de l'anniversaire pour deviner l'ID
- Les autres infos sont faciles à deviner (surtout si vous êtes l'auteur de la question au serveur cache)

DNS poisoning : historique



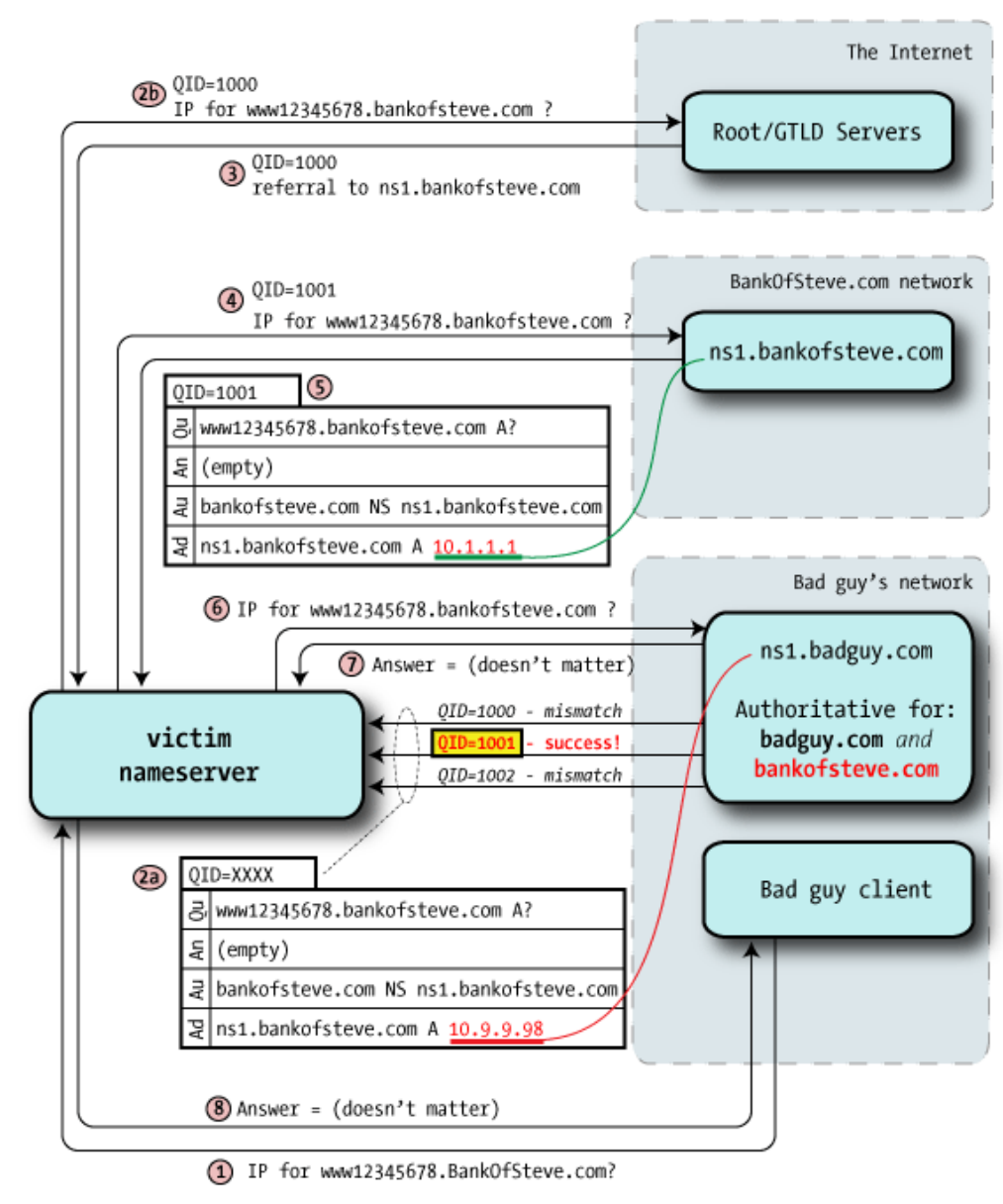
http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

DNS poisoning : v1



http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

DNS poisoning : v2 (Dan Kaminsky)



Résumé

- But
  - Sécurisation du DNS
  - Empêcher les attaques vues précédemment
- Moyen
  - Les enregistrements d'une zone sont signés
  - Il existe une chaîne de confiance pour vérifier les signatures
- Outils
  - Nouveaux Resource Records (RR)
  - DNSKEY pour les clés de chiffrement
  - RRSIG pour les signature
  - DS pour les délégations

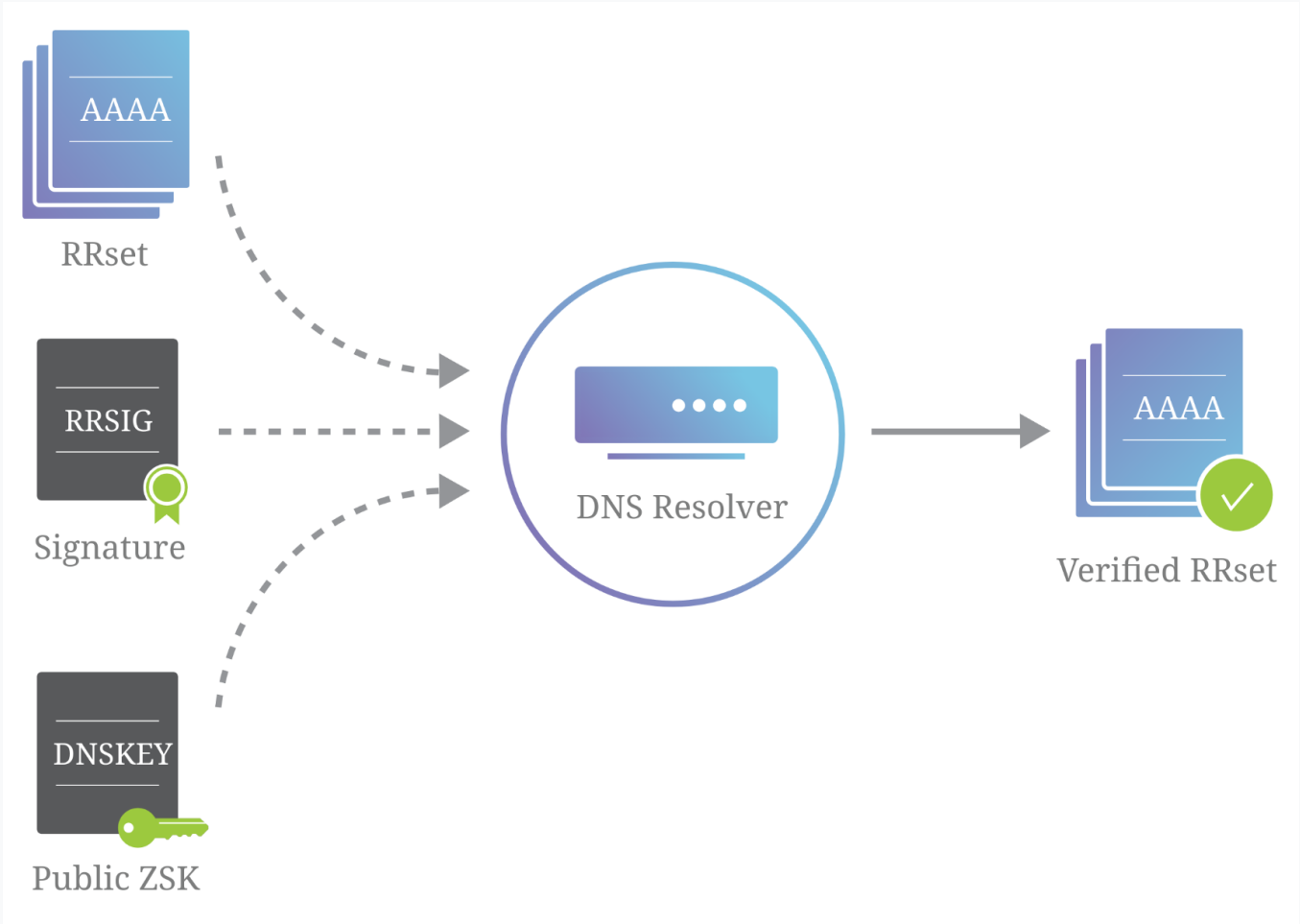
Principe

- On peut suivre un chemin sécurisé dans l'arbre depuis le point de confiance jusqu'à l'enregistrement
- On accepte une réponse DNS que si l'enregistrement est vérifiable

Principe

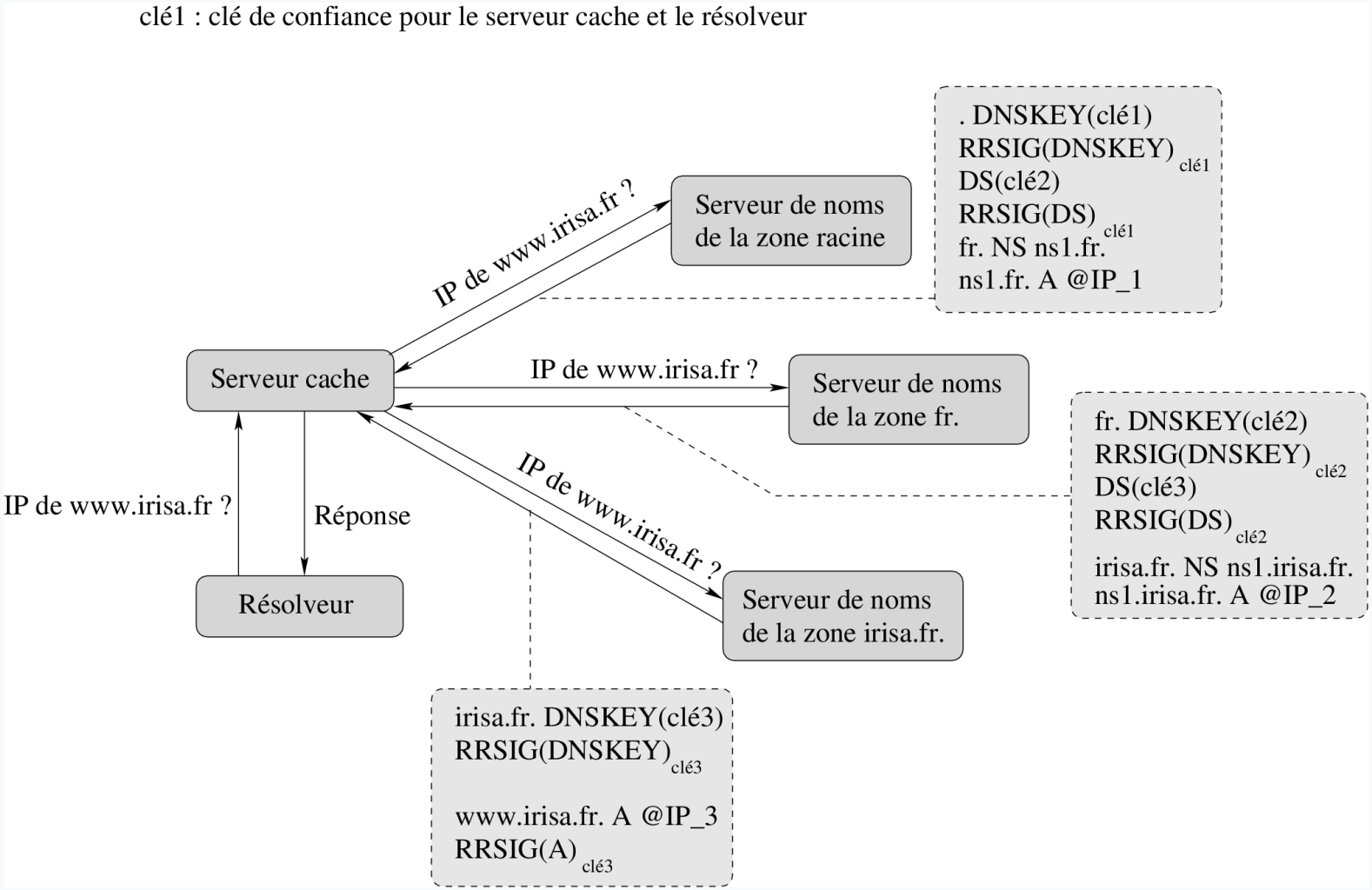
- Créer une chaîne de confiance vérifiable entre un point de confiance et l'enregistrement reçu
- Chaque zone possède au moins une clé publique
- Tous les enregistrements d'une zone sont signés
- Chaque résolveur possède au moins un point de confiance : une clé publique d'une zone haut dans l'arbre DNS

Principe



Les enregistrements de la zone sont signés avec la clé privée relative à l'autorité de cette zone.

Principe

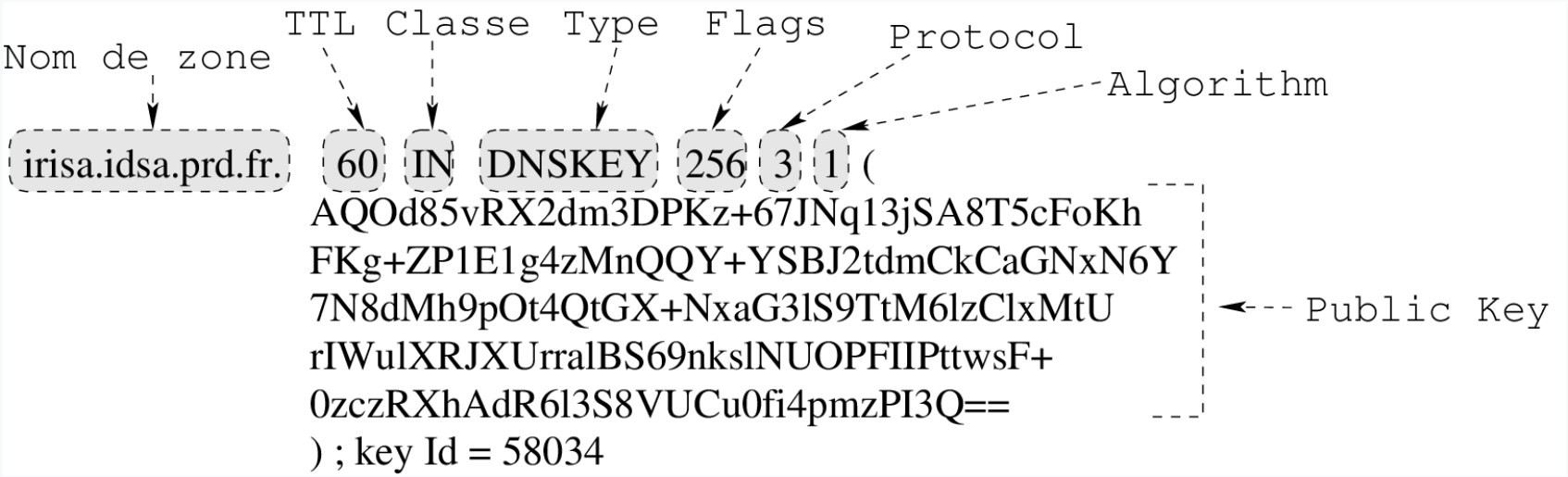


Chaîne de confiance : www.irisa.fr est signé, irisa.fr est signé, .fr est signé.

Les nouveaux RR

DNSKEY RR

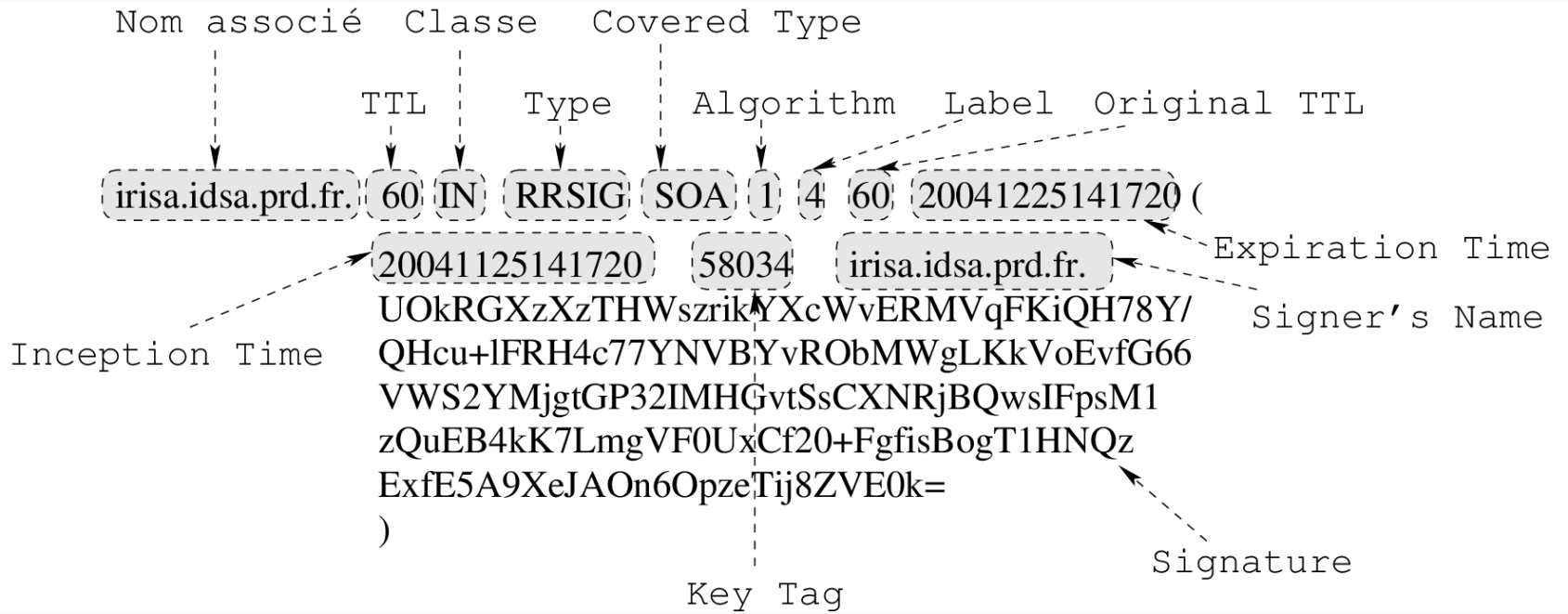
- Il contient une clé publique



Les nouveaux RR

RRSIG RR

- Il contient la signature d'un RR (sauf d'un RRSIG)
- Un RRSIG est généré par chaque clé de zone et pour chaque groupe de RR ayant même nom, type et classe

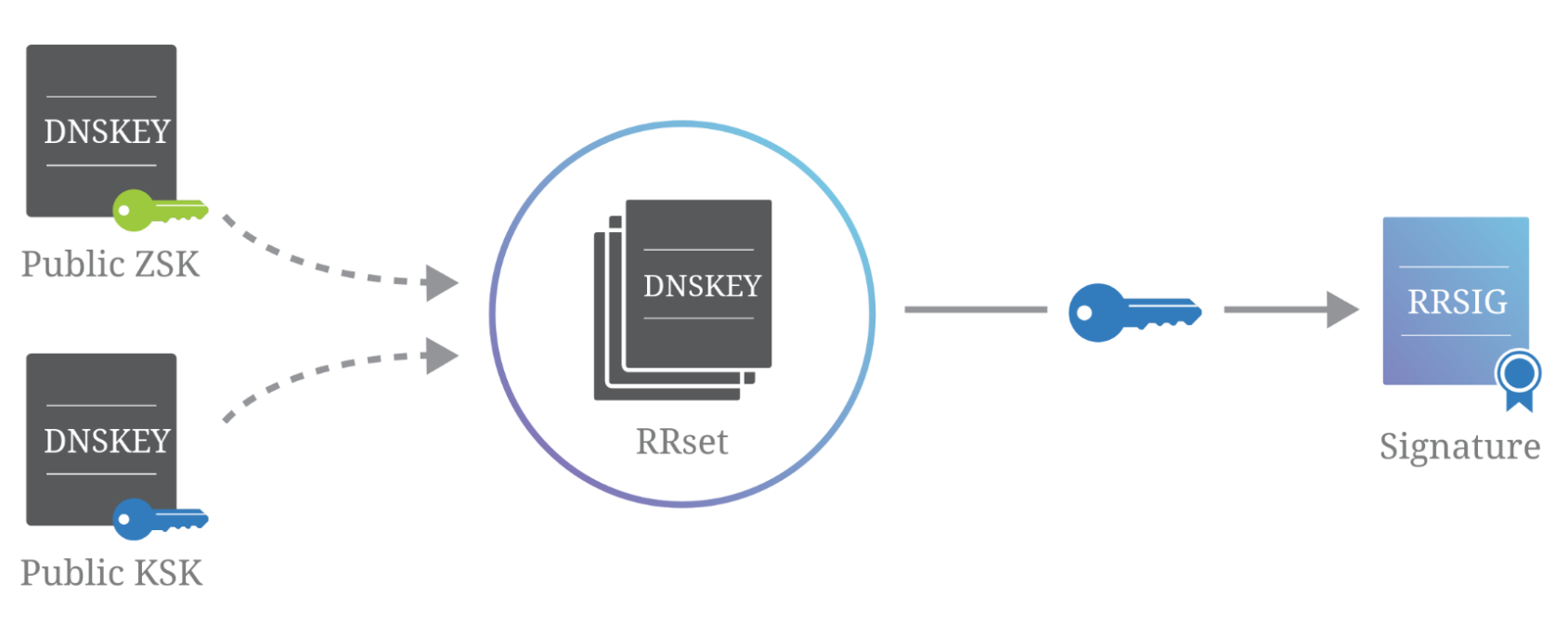


Key Signing Key et Zone Signing Key

RRSIG RR

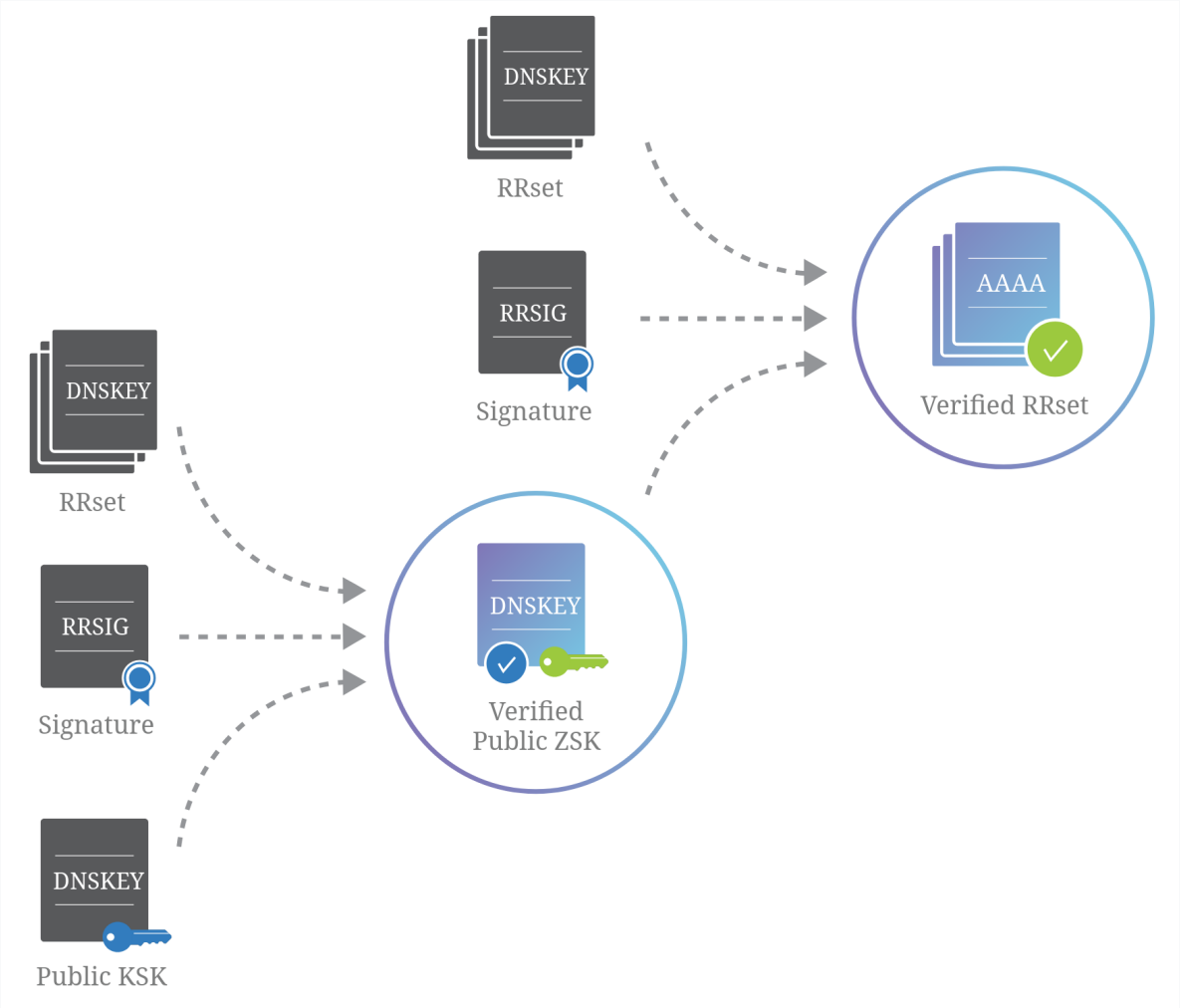
- Plus on utilise une clé, plus elle « s'affaiblit »
- Création de deux types de clé
  - KSK : elle ne signe que les DNSKEY
  - ZSK : elle signe tous les enregistrements (ex : A, CNAME, MX...)
  - Les ZSK sont changées plus fréquemment car plus utilisées
  - Le *Flag* permet de distinguer KSK et ZSK

Key Signing Key et Zone Signing Key



La clef de signature de la zone (ZSK) est signée par la clef de signature des clefs (KSK).

Key Signing Key et Zone Signing Key



On vérifie à la fois l'authenticité des enregistrements DNS et la validité de la ZSK via la signature de la KSK.

Le problème de la délagation

Autres motivations à ZSK/KSK : le problème de la délégation

Les délégations DNS utilisent les RR appelés *glue*

- Ce sont les **NS** RR et **A** RR associés de la zone fille qui sont remontés dans la zone parente

```
~ ❯ dig example.com NS

; <<>> DiG 9.20.8 <<>> example.com NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 891
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 5

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;example.com.                   IN      NS

;; ANSWER SECTION:
example.com.            16901   IN      NS      b.iana-servers.net.
example.com.            16901   IN      NS      a.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net.     17196   IN      A       199.43.135.53
```

Le problème de la délagation

Si on fait de même en DNSSEC :

- La zone parente possède la ZSK1 et signe sa zone
- La zone fille possède la ZSK2 et signe sa zone
- On remonte des RR NS et A signés par ZSK2 dans la zone parente qui ne connaît que ZSK1...

Le problème de la délagation

Pour les délégation on va donc garder l'utilisation des *glues*

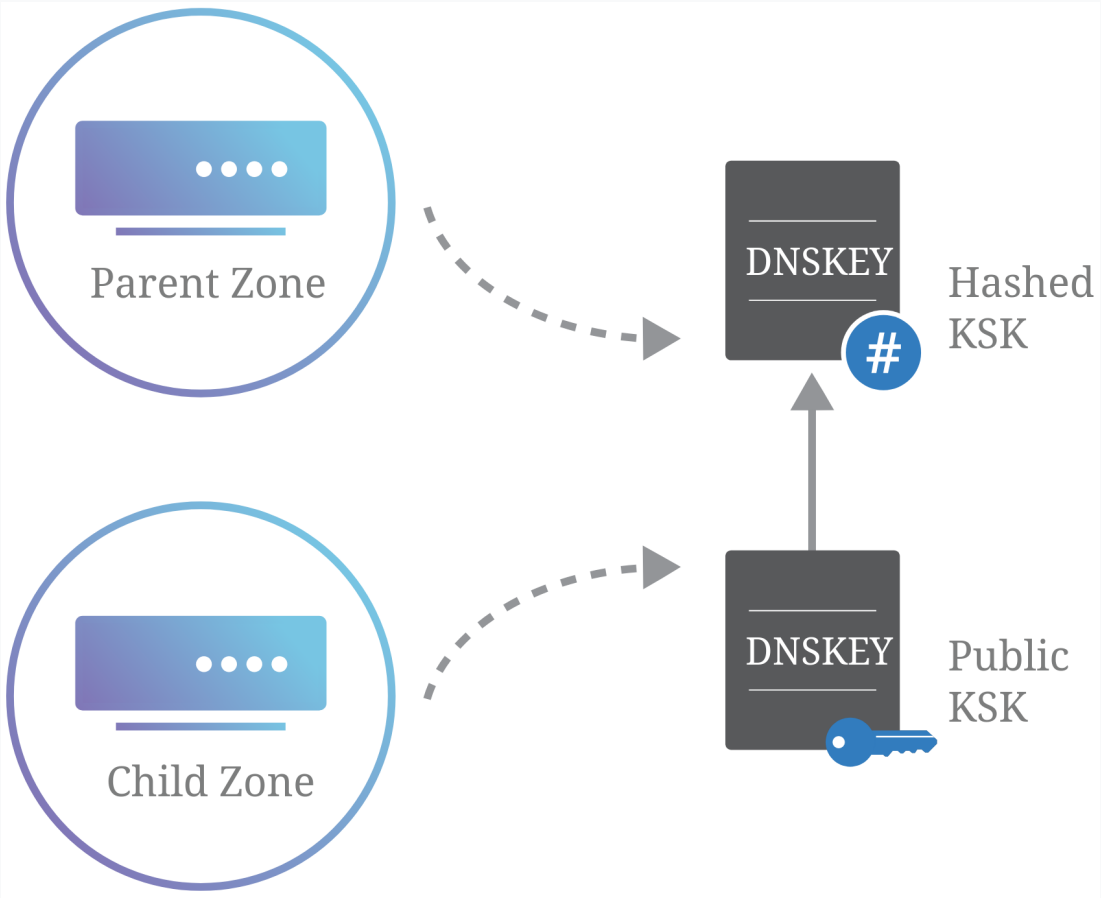
- Mais sans les signer
- Les signatures sont inutiles car on ne peut pas encore les vérifier
- En revanche on va ajouter un nouvel enregistrement pour garder le lien sécurisé entre zone

Les nouveaux RR

DS RR (Delegation Signer)

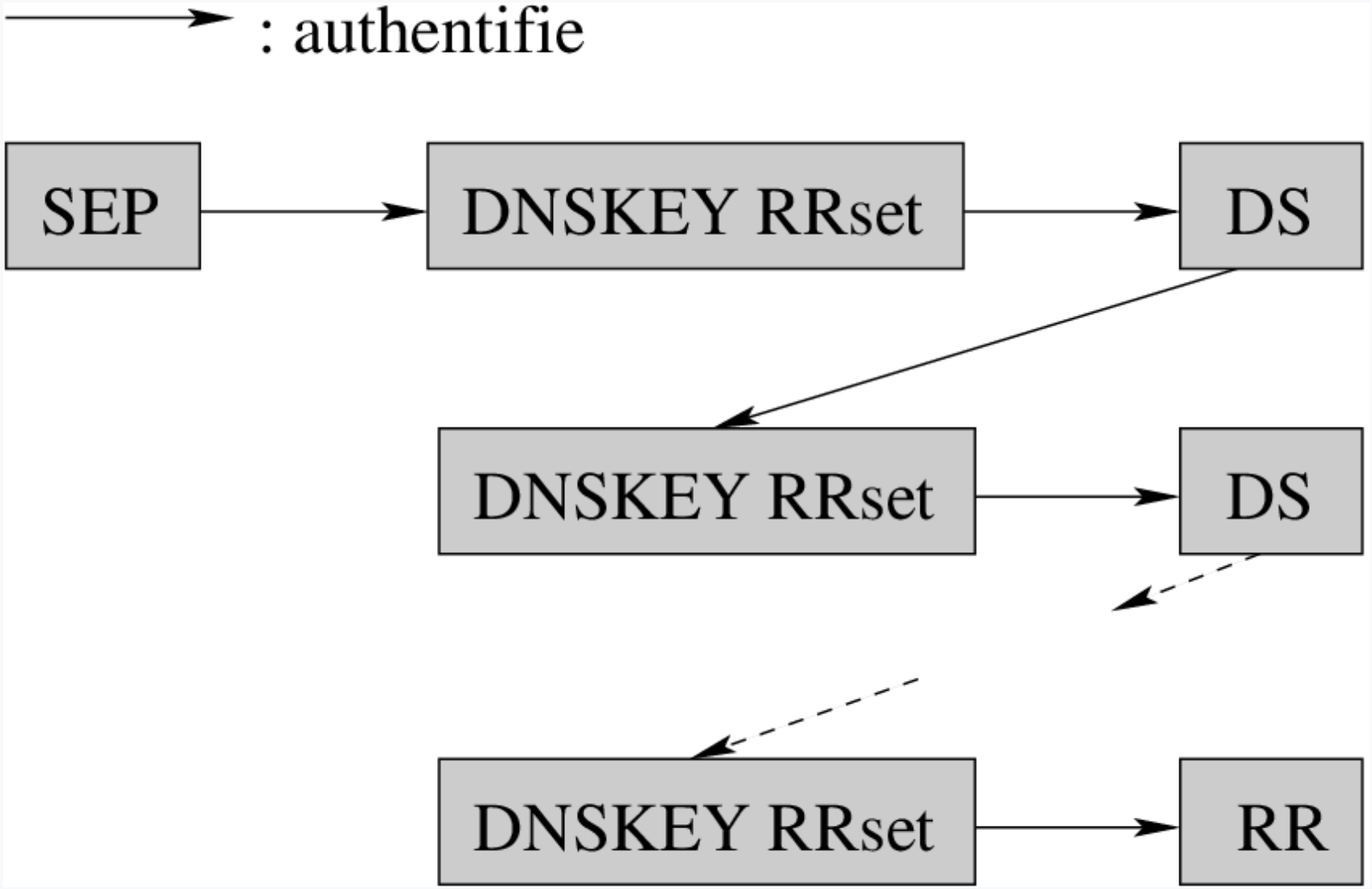
- Il fait le lien authentifié au point de délégation
- Un DS RR de la zone parente authentifie un DNSKEY RR (KSK uniquement) de la zone fille

La chaîne de confiance

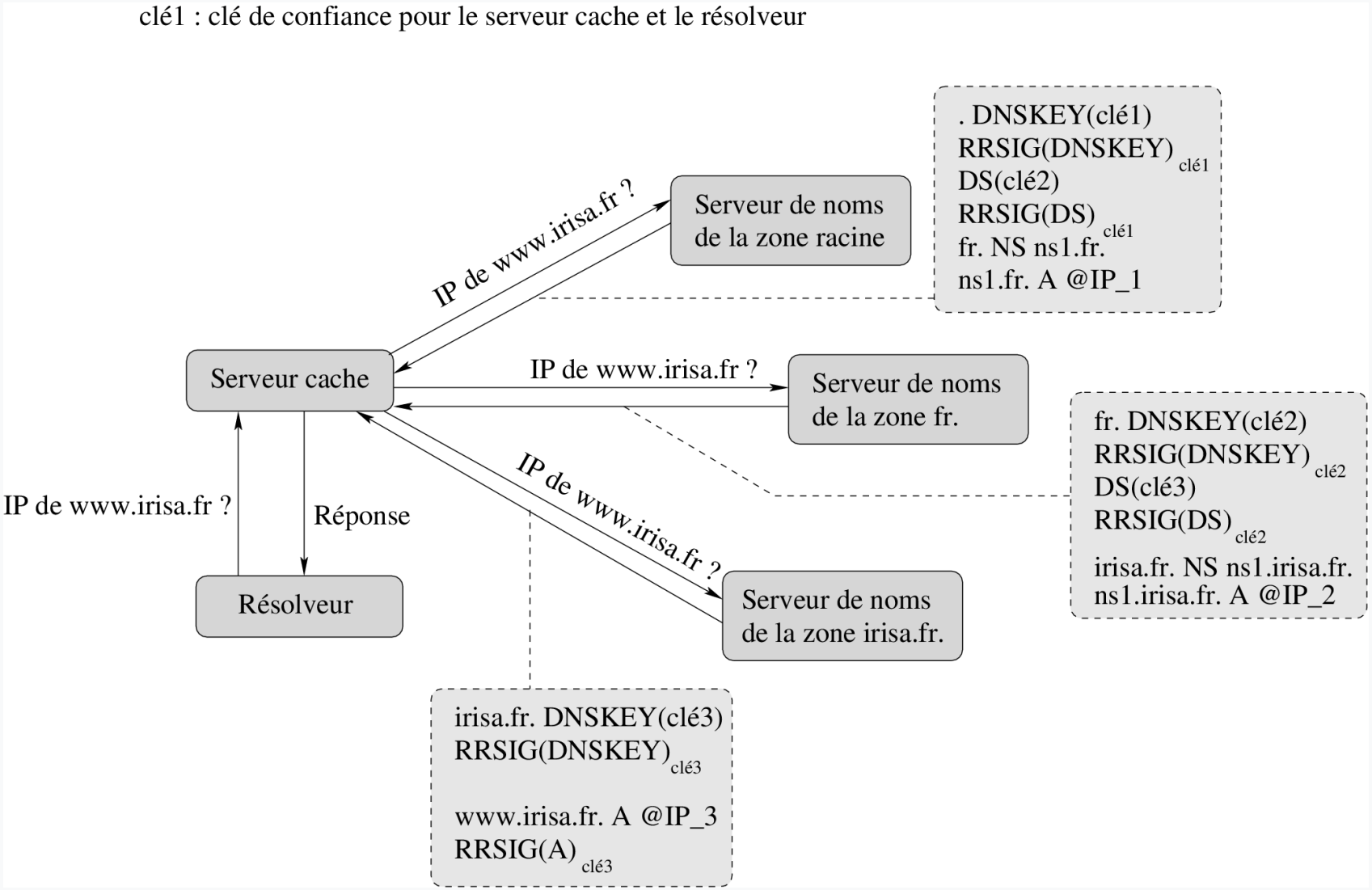


La zone parente publie une empreinte (DS) de la KSK de la zone enfant : la zone parent annonce quelle KSK est attendue dans le DNSKEY de la zone enfant.

La chaîne de confiance



La chaîne de confiance



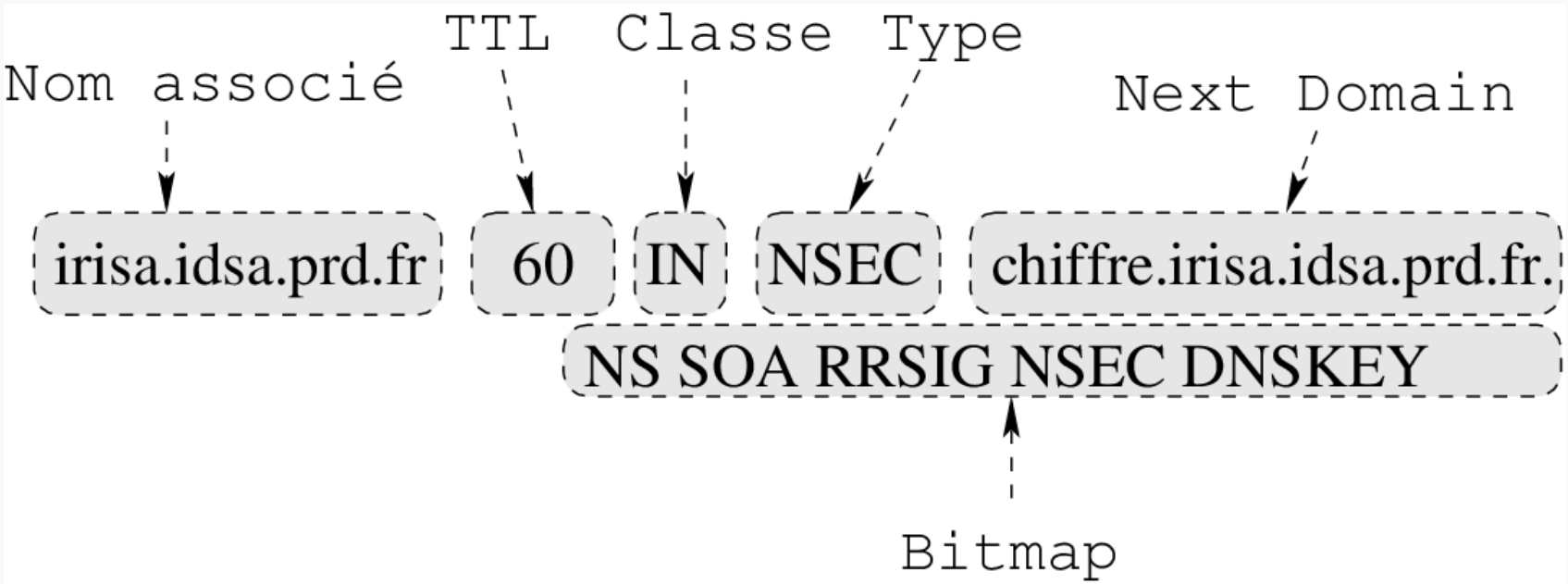
Question

- Prouver la véracité d'un RR existant c'est facile
  - On montre le RR et on montre sa signature
  - Si la signature est vérifiable, le RR est correct
- Et si le RR demandé n'existe pas ?
- Comment prouver de manière sécurisée la non existance ?

Et si ça n'existe pas ?

NSEC RR

- Il sert à prouver la non-existence d'un nom ou d'un RR



NSEC : fonctionnement

- Retourne le nom précédent et le nom suivant dans la zone.
- Permet de prouver qu'un nom n'existe pas entre deux existants.

```
example.com.  NSEC ftp.example.com.  A MX RRSIG
NSEC
```

NSEC : fonctionnement

- Retourne le nom précédent et le nom suivant dans la zone.
- Permet de prouver qu'un nom n'existe pas entre deux existants.

```
example.com.  NSEC ftp.example.com.  A MX RRSIG
NSEC
```

- **Inconvénient** : tous les noms existants deviennent visibles.

Le problème du DNS walking

La première version de l'enregistrement NSEC appelé NXT permet l'énumération de la zone :

- L'attaquant interroge la zone avec un nom qui a peu de chance d'exister : aaaa.zone.fr.
- Le serveur DNSSEC retourne des preuves de non-existence (ex : NSEC/NSEC3).
- Ces preuves révèlent la structure de la zone.
- On interroge de nouveau avec le nom suivant+a par exemple
- Enchaîner ces requêtes permet de reconstruire toute la zone.

Le NXT/NSEC walking

```
~ ❯ dig +dnssec a.ripe.net A                                                    11:04:39
;; Warning: Client COOKIE mismatch

; <<>> DiG 9.20.8 <<>> +dnssec a.ripe.net A
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 44129
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 65494
; COOKIE: 880bf69e73dc994e0100000068187ed86c0e7846232d0919 (bad)
;; QUESTION SECTION:
;a.ripe.net.                    IN      A

;; AUTHORITY SECTION:
_443._tcp.ripe.net.     3516    IN      RRSIG   NSEC 13 4 3600 20250517110002 20250503093002 38
758 ripe.net. r5IYRvxousSsVa1aD25eti/H6uhIUARm+EZc+xmk8LVn0C1j4HfdxHly OBY9EDCJRjI6+k44qmJYd0XO
JGg81w==
_443._tcp.ripe.net.     3516    IN      NSEC    aberdeen.ripe.net. CNAME RRSIG NSEC
ripe.net.               3516    IN      SOA     manus.authdns.ripe.net. dns.ripe.net. 174642985
1 3600 600 864000 3600
```

Le problème du NSEC

- Beaucoup de zone DNS ne sont pas distribuées publiquement
  - La zone est vendue : contrat + CGU
  - Pour la zone .fr 10000 euros par an
  - Aussi : exposition d'informations utiles aux attaquants
- Il faut donc empêcher l'énumération

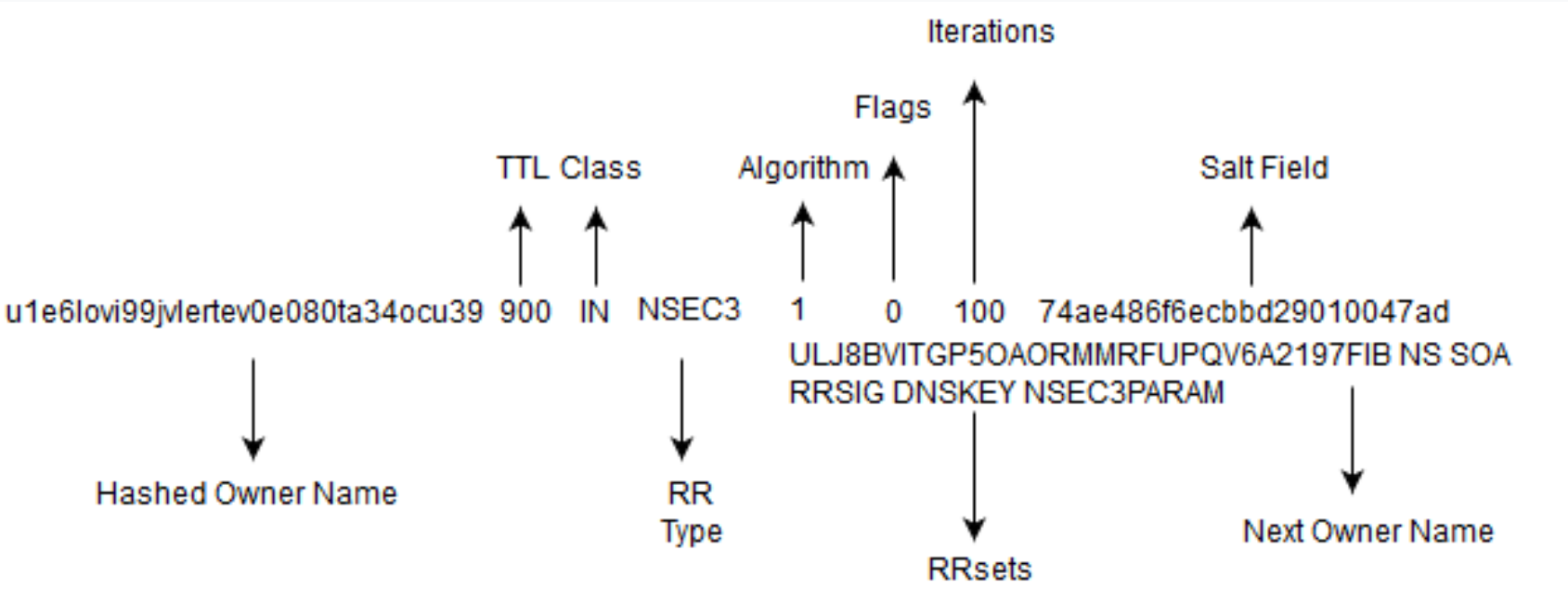
Première tentative : NSEC3

- Utilise les hash (salés) des noms de domaine.
- Rend les noms invisibles directement (non en clair).

```
NSEC3 <hash1> <hash2> ...
```

Problème de NSEC3 : zone walking toujours possible

- Attaque plus difficile, mais pas impossible.
- Les enregistrements restent ordonnés → bruteforce possible.
- Dictionnaires et faibles entropies facilitent le bruteforce.



Seconde tentative : White Lies

- Réponses NSEC/NSEC3 générées dynamiquement.
- Ne permettent pas de déduire l'ordre réel des noms.
- Ralentit considérablement les attaques.
- **Avantage** : conforme à DNSSEC.

Seconde tentative : White Lies

- Réponses NSEC/NSEC3 générées dynamiquement.
- Ne permettent pas de déduire l'ordre réel des noms.
- Ralentit considérablement les attaques.
- **Avantage** : conforme à DNSSEC.
- **Inconvénient** : plus coûteux en calcul + le serveur doit avoir accès à la clef privée pour signer à la volée.

https://www.cloudflare.com/learning/dns/dnssec/ecdsa-and-dnssec/

Black Lies (ex : Cloudflare)

- Technique répondant avec un NSEC contenant `\000.nom_demandé`.
- Empêche toute itération ou tri logique dans la zone.
- **Très efficace** contre le DNS Walking.
- Conforme à DNSSEC.

```
example.com.  NSEC \000.example.com.  A RRSIG NSEC
```

https://blog.cloudflare.com/black-lies/

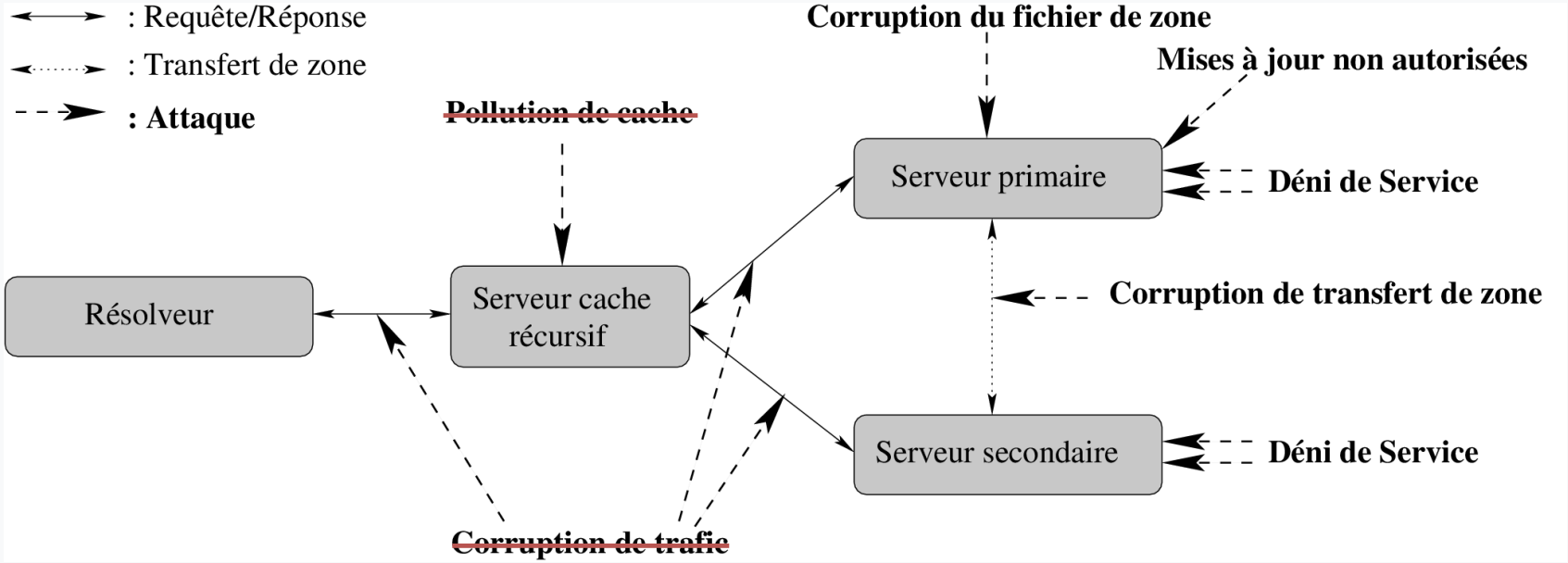
Quelles vulnérabilités a-t-on corrigées ?

- Un attaquant ne peut plus insérer de faux RR dans une réponse DNS
  - Il n'est pas capable d'en générer les signatures
- Un attaquant ne peut plus insérer de faux RR dans un cache (DNSSEC)
  - Il n'est pas capable d'en générer les signatures

Quelles vulnérabilités a-t-on corrigées ?

- Un attaquant peut rejouer de vieilles réponses DNSSEC si la signature est toujours valide
  - Ce n'est pas grave ce sont de vraies info
- Un attaquant ne peut pas supprimer des RR d'une réponse DNSSEC
  - Il peut néanmoins tout supprimer et faire passer la réponse pour une réponse DNS standard !

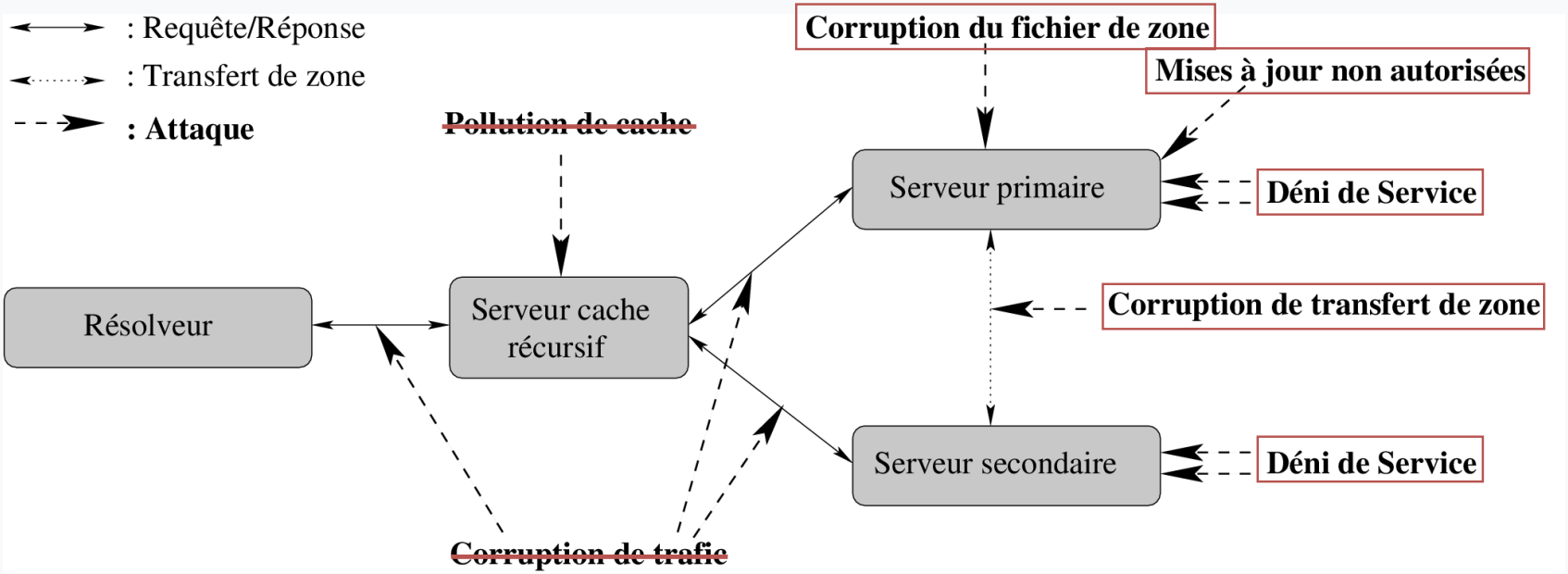
Quelles vulnérabilités reste-t-il ?



Quelles vulnérabilités reste-t-il ?

- Un attaquant peut toujours intervenir en corrompant le fichier de zone
- Un attaquant peut toujours corrompre le transfert de zone

Quelles vulnérabilités reste-t-il ?



La sécurisation des transactions

- Deux mécanismes ont été développés pour sécuriser les transactions
  - TSIG basé sur de la crypto symétrique
  - SIG(0) basé sur de la crypto asymétrique
- L'utilisation est le plus souvent limitée aux transactions entre serveur primaire et secondaire ou pour les mises à jour dynamiques

TSIG

- Il s'agit d'un meta RR
  - Il ne fait pas partie du fichier de zone et n'est **JAMAIS** mis en cache
  - Il est généré si on en a besoin
  - Il contient la signature du haché du message DNS complet
- TSIG nécessite l'implantation de la clé privée sur le serveur primaire et les serveurs secondaires

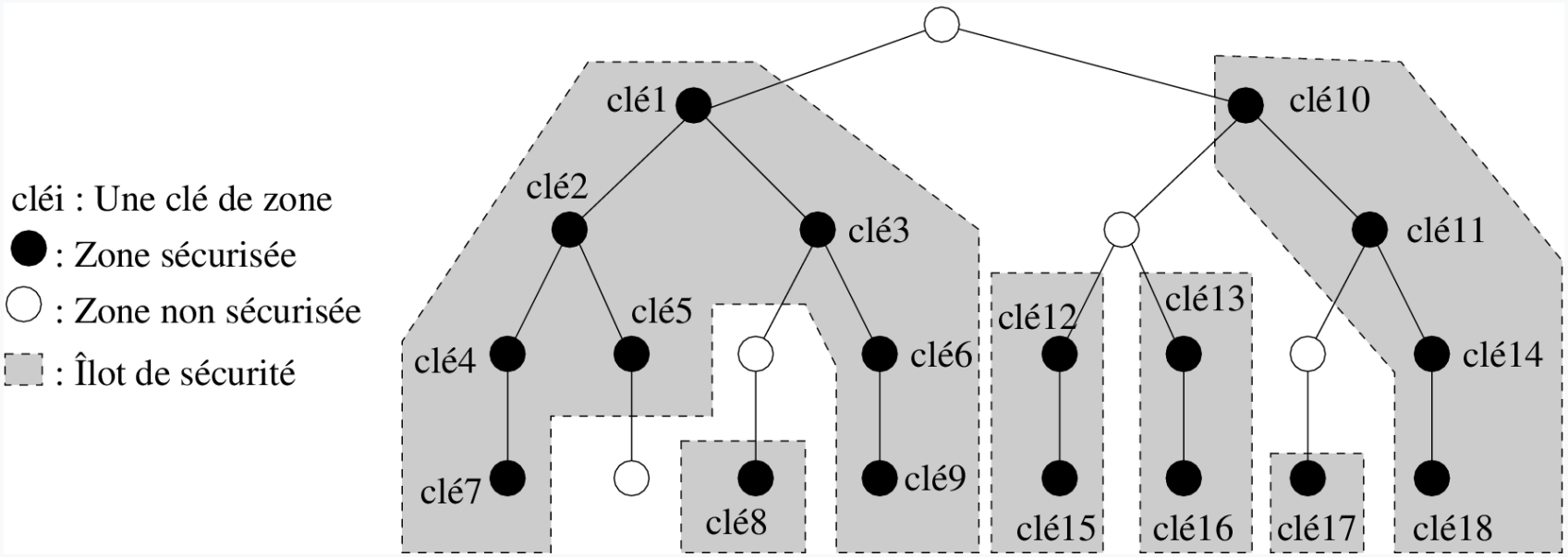
SIG(0)

- Il s'agit d'un meta RR
  - Il ne fait pas partie du fichier de zone et n'est **JAMAIS** mis en cache
  - Il contient une signature de tout le message DNS à la manière d'un RRSIG
  - **ATTENTION** : La signature impose que la partie privée de la clé soit disponible sur le serveur
  - Peu utilisé

Et ensuite : le déploiement

- Le déploiement de DNSSEC, comme beaucoup d'autres (IPv6), ne se fera pas de manière instantanée
- Cohabitation simultanée de zones sécurisées et de zones non-sécurisées
- Présence d'îlots de sécurité, c'est-à-dire de sous-arbres sécurisés par DNSSEC

Et ensuite : le déploiement



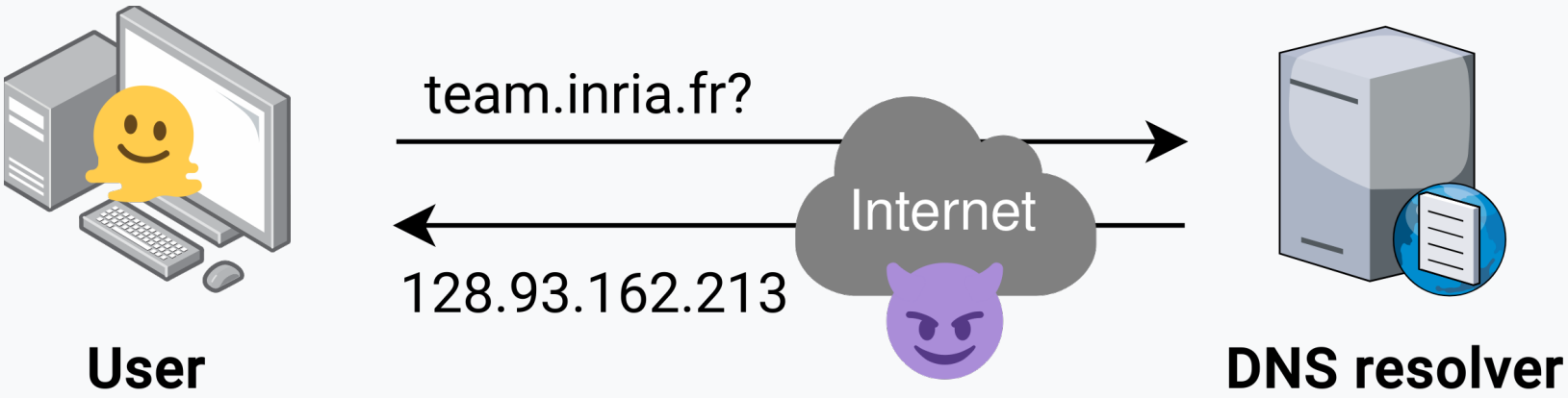
Pour aller plus loin

- Il reste quelques contraintes de déploiement
- Pour aller plus loin vous pouvez regarder du côté :
  - Du meta RR EDNS(0) : pour la taille des messages
- Ou réfléchir aux problèmes de renouvellement de clés de zone KSK et ZSK…

Bonus : quid de la confidentialité ?

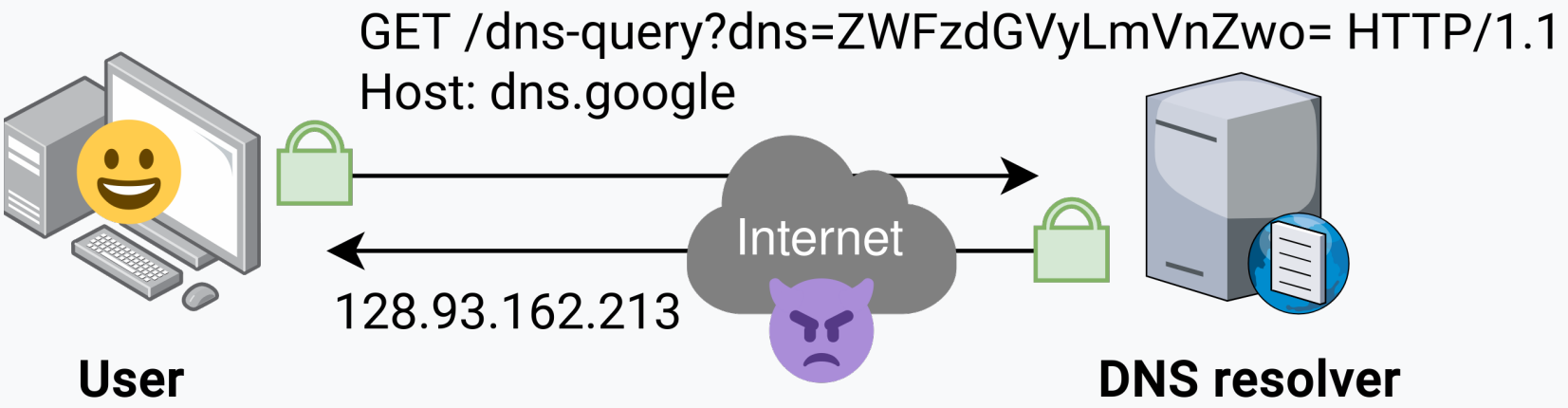
Bonus : quid de la confidentialité ?

- DNSSEC ne s'occupe pas du chiffrement



Bonus : quid de la confidentialité ?

Pour le chiffrement : encapsulation



Exemples :

- DNS-over-HTTPS (port 443)
- DNS-over-TLS (port 853)
- ...