DNSsec : la solution et les problèmes

version adaptée par l'Irisa

Plan

- Rappels sur DNS
- Les solutions pour la sécurisation du DNS
  - TSIG
  - DNSSEC
- Les problèmes

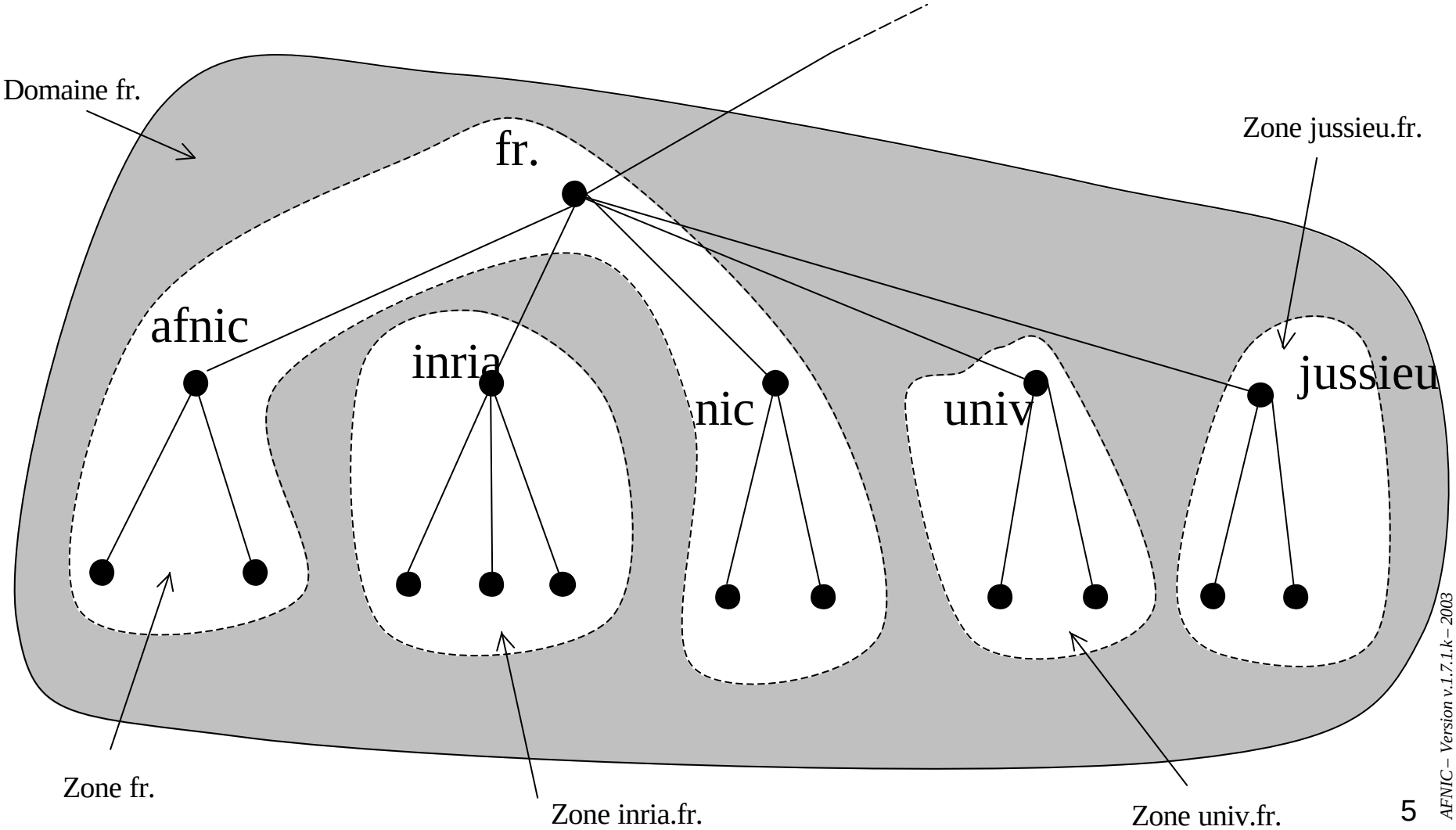
Rappels sur DNS
Plan

- Rappels sur le fonctionnement de DNS
  - Domaines/Zones
  - RR/RRset
  - Résolution
  - Flux
- Vulnérabilités des flux DNS

Fonction de DNS

- Un système d'annuaire réparti : DNS
  - Paul Mockapetris (ISC) - 1984
  - RFC 882/883 puis 1034/1035
- Années 2000
  - DNS est un service critique pour le fonctionnement d'Internet
  - Vers un système de nommage sécurisé :
    - DNSSEC,
    - TSIG,
    - …

Domaines vs. zones



Fichier de zone

```
$TTL 86400
$ORIGIN afnic.fr.

@                IN      SOA     ns1.nic.fr. hostmaster.nic.fr. (
                                         2002080800 ; serial
                                         21600      ; refresh (6 hours)
                                         3600       ; retry (1 hour)
                                         2419200    ; expire (4 weeks)
                                         86400      ; TTL neg. (1 day)
                                 )
                 IN      NS      ns1.nic.fr.
                 IN      NS      ns2.nic.fr.
                 IN      NS      ns3.nic.fr.

                 IN      MX      10 relay1.nic.fr.
                 IN      MX      20 relay2.nic.fr.

asterix          IN      A       192.134.0.5
                 IN      MX      10 relay3.nic.fr.
                 IN      TXT     "Dell GX150 StQuentin"
www              IN      CNAME   asterix
```

Du RR au RRset

- **R**essource **R**ecord
  Enregistrement de Ressource

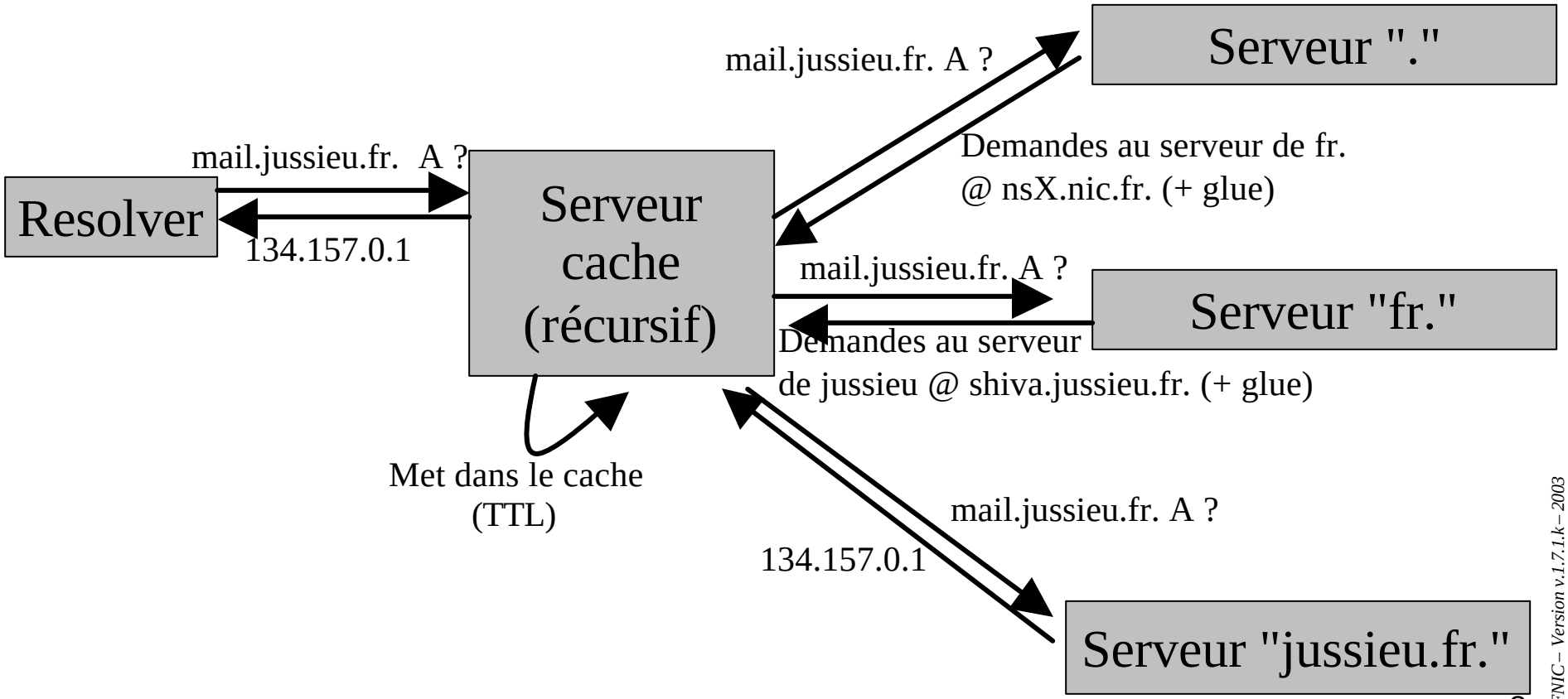
Name	*TTL*	*Class*	*Type*	*Rdata*
host.exemple.fr.fr.	**86400**	**IN**	**A**	**192.134.0.1**

- **R**essource **R**ecord **set**
  Ensemble d'enregistrements de ressources
  de même nom, même classe et même type

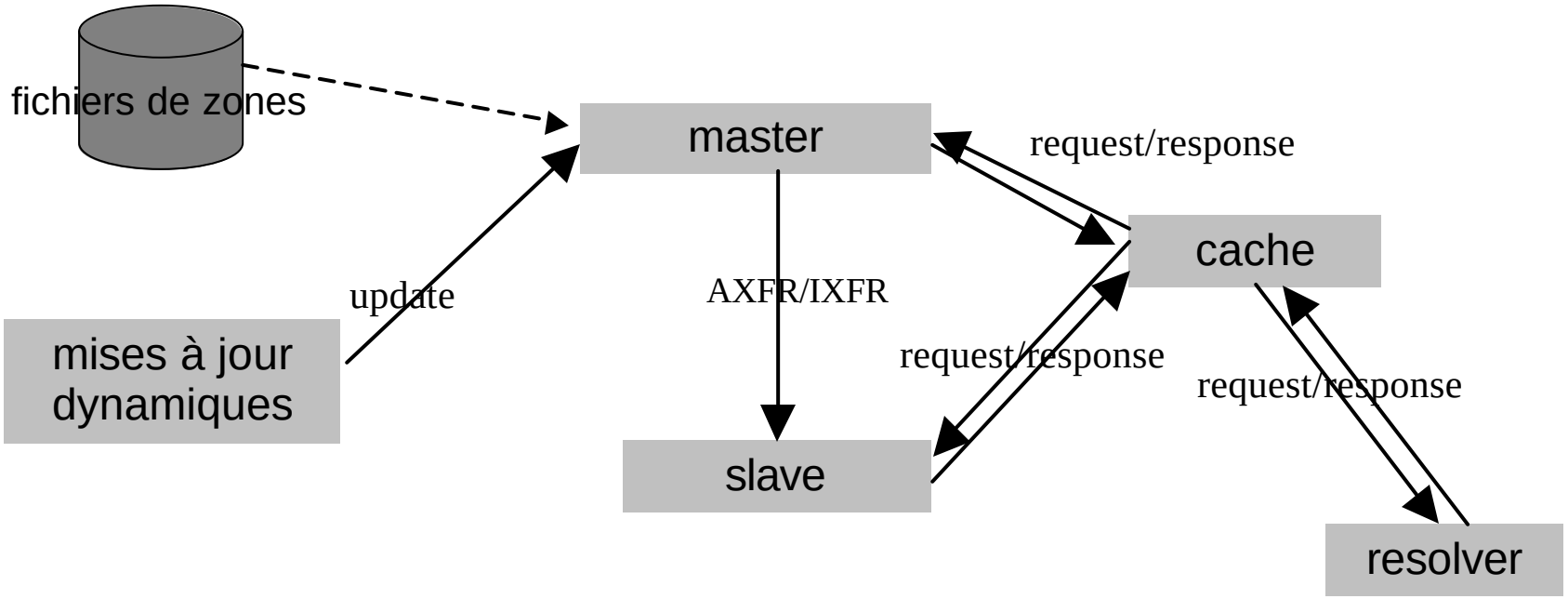
host.exemple.fr.fr.	**86400**	**IN**	**A**	**192.134.0.1**
host.exemple.fr.fr.	86400	**IN**	**A**	**192.134.0.2**

Résolution DNS

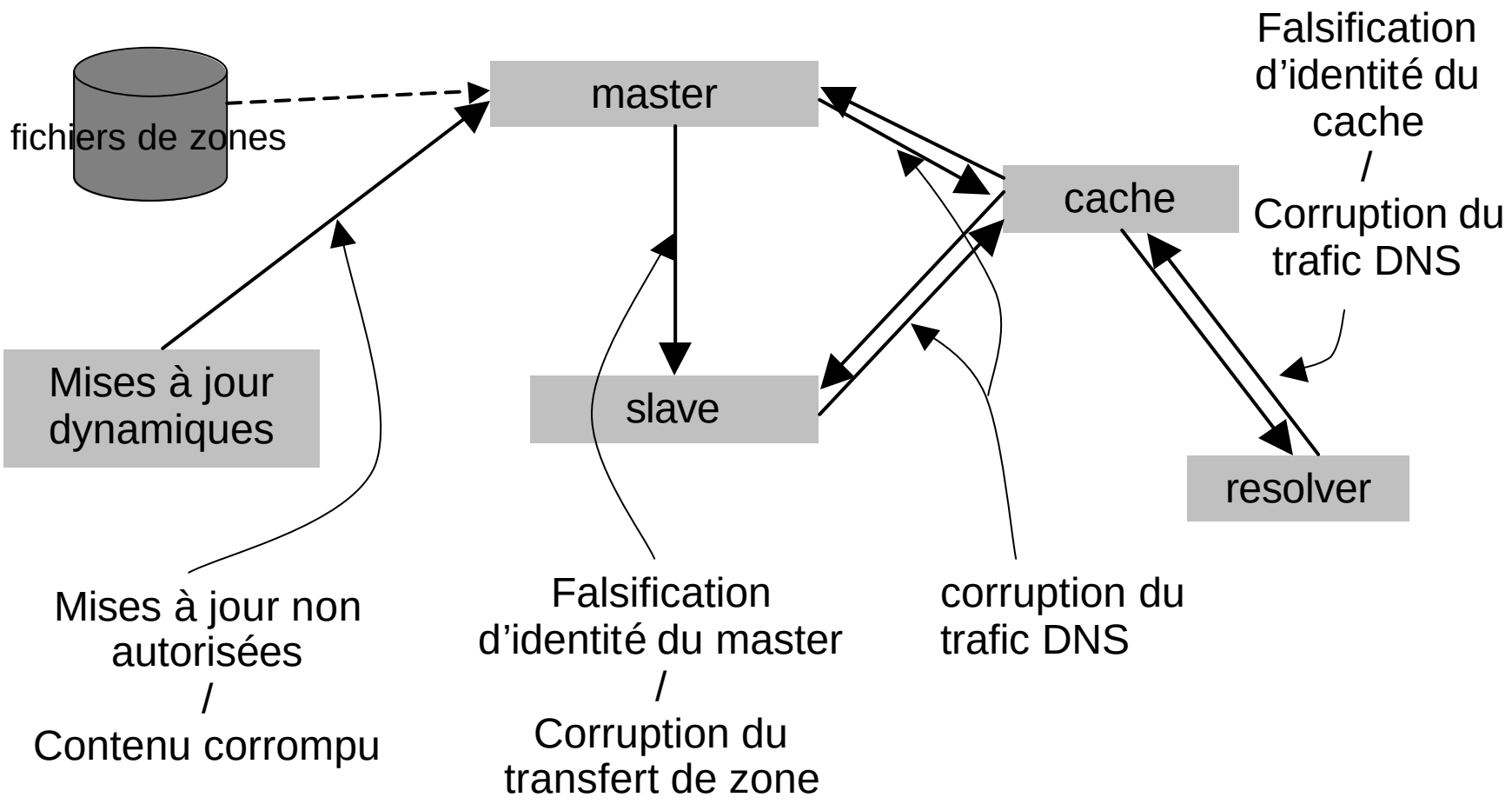
Question : mail.jussieu.fr. A ?



Flux DNS



Vulnérabilités des flux DNS



- Fonctionnement
- Exemple avec AXFR
- Bilan des vulnérabilités résolues et non résolues

TSIG
Fonctionnement

- Transaction Signature [RFC 2845]
  - Secret partagé (cryptographie symétrique)
  - Signature d'un haché
    (actuellement algorithme HMAC-MD5)
    - Authentification
    - Intégrité
  - Tient compte de la date et de l'heure
    Synchronisation de NTP nécessaire

- Utiliser pour
  - Les transferts de zone (master / slaves)
  - Les mises à jour dynamiques
  - Les transactions entre resolvers et caches

Exemple de TSIG avec AXFR

Master →

```
key “transfer-key" {
        algorithm hmac-md5;
        secret "sAfRkDLdld56lfD5LvD46DxlFm6flS=";
};
zone confiance.fr {
        type master;
        file “db.confiance.fr”;
        allow-transfer { key transfer-key; };
}
```

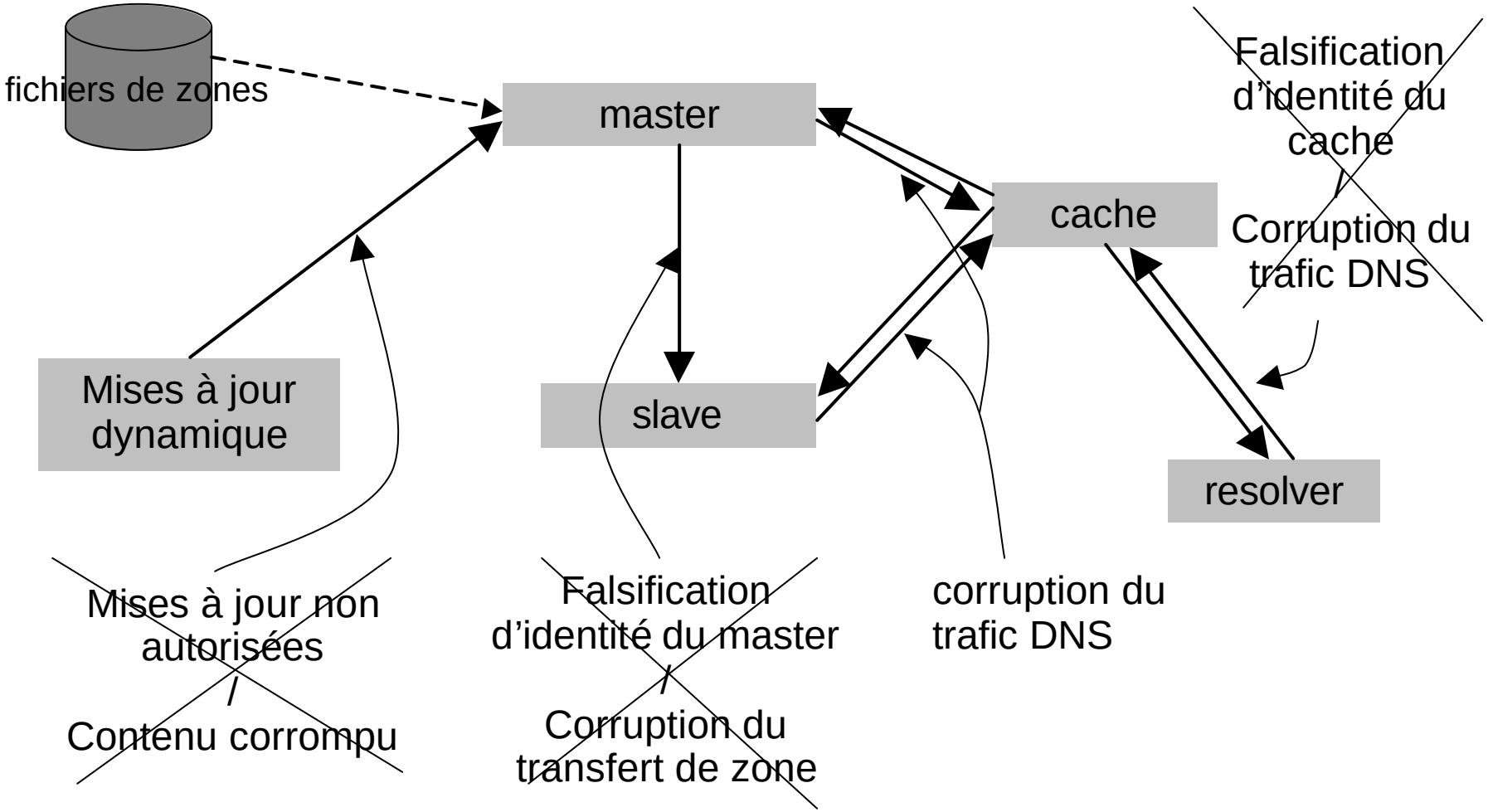
Slave →

```
key “transfer-key" {
        algorithm hmac-md5;
        secret "sAfRkDLdld56lfD5LvD46DxlFm6flS=";
server 192.249.249.1 {
        keys { transfer-key; };
};
zone confiance.fr {
        type slave;
        file “db.confiance.fr”;
        masters { 192.249.249.1; };
};
```

Attention : Secret, algorithme et nom affectés à la clé doivent être identiques sur Master et Slave

TSIG

Bilan des vulnérabilités résolues et non résolues



Inconvénients de TSIG

- L'utilisation de TSIG n'est pas raisonable pour protèger les requêtes DNS :
  - La gestion des clés est trop lourde !

- Nouveaux RR
- Chaîne de confiance
  - RFC 2535
  - Draft DS
    - Principes
    - Exemples
    - Avantages/Inconvénients
- Bilan des vulnérabilités résolues et non résolues

DNSSEC : 4 Nouveaux RR

- KEY
- SIG
- NXT
- DS

RFC 2535

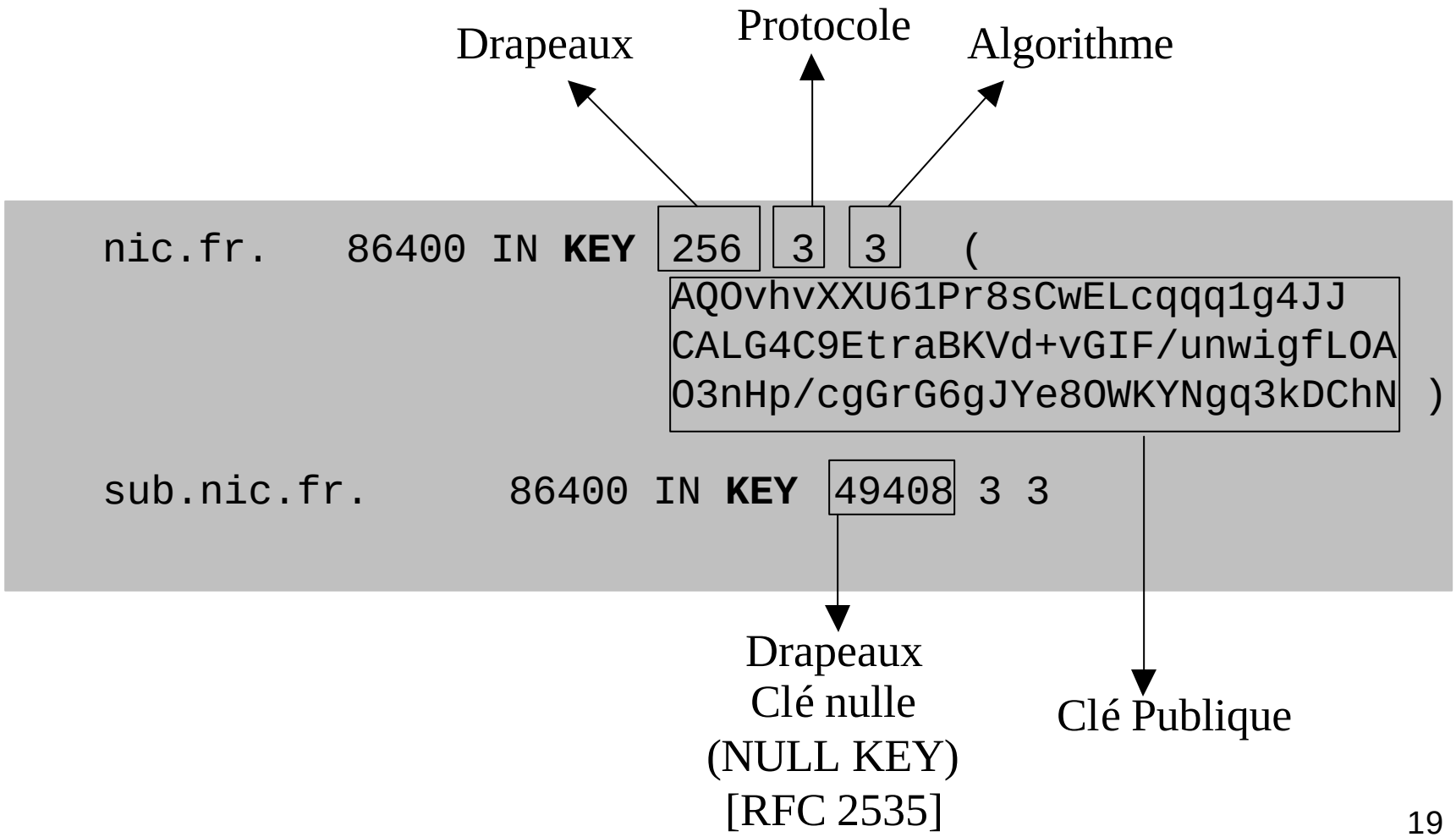
Draft DS

Le RR KEY
Principe

- Support de clé de chiffrement
  - Porte une clé publique (chiffrement asymétrique)
  - Multi usages (protocoles)
    DNSSEC, TLS, IPSec, Messagerie, …
  - Multi algorithmes
    RSA/MD5, Diffie Hellman, DSA, …

- KEY dans DNSSEC
  - Porte la clé publique d'une zone
  - Doit permettre authentification et intégrité
  - Peut représenter la clé nulle (NULL KEY) [RFC 2535]

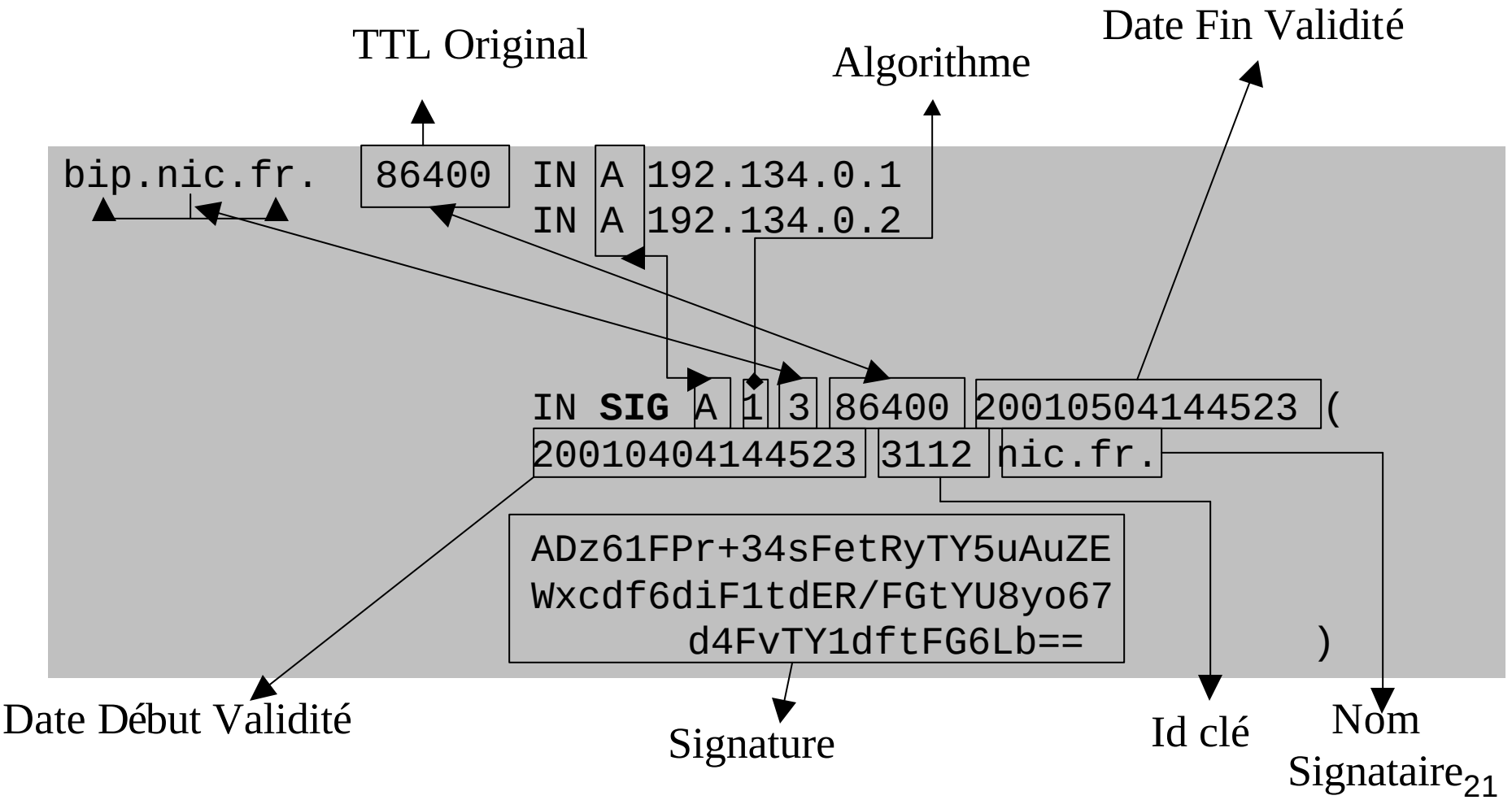
Le RR KEY
Exemple



Le RR SIG
Principe

- Signe
  - un RRset,
    Impossible de rapatrier un membre d'un RRset seul
  - faisant autorité,
    Aux points de délégations,
    les NS et les glues ne sont pas signés dans la zone parente
  - avec la clé privée associée à un KEY
    Signer avec n KEY => n SIG par RRset
- Multi-algorithmes : celui du RR KEY utilisé
- Aspects temporels
  - Dates de début et de fin
    - Représentées dans le RR SIG au format AAAAMMJJHHMMSS
    - En interne nombre de secondes depuis le 1er Janvier 1970 modulo ~ 136 ans
  - Synchronisation NTP nécessaire
- Identifiant de clé utilisé :
  généré à partir de la clé publique (RR KEY)

Le RR SIG
Exemple



Le RR SIG
La Signature

- Données signées :

  data = Rdata | RR(s)
  - Rdata : tous les champs du RR SIG
    (sauf la signature!)
  - RR(s) :
    - Tout le contenu des RR appartenant au RRset signé
    - Ordonnés : voir fonction d'ordonnancement plus loin

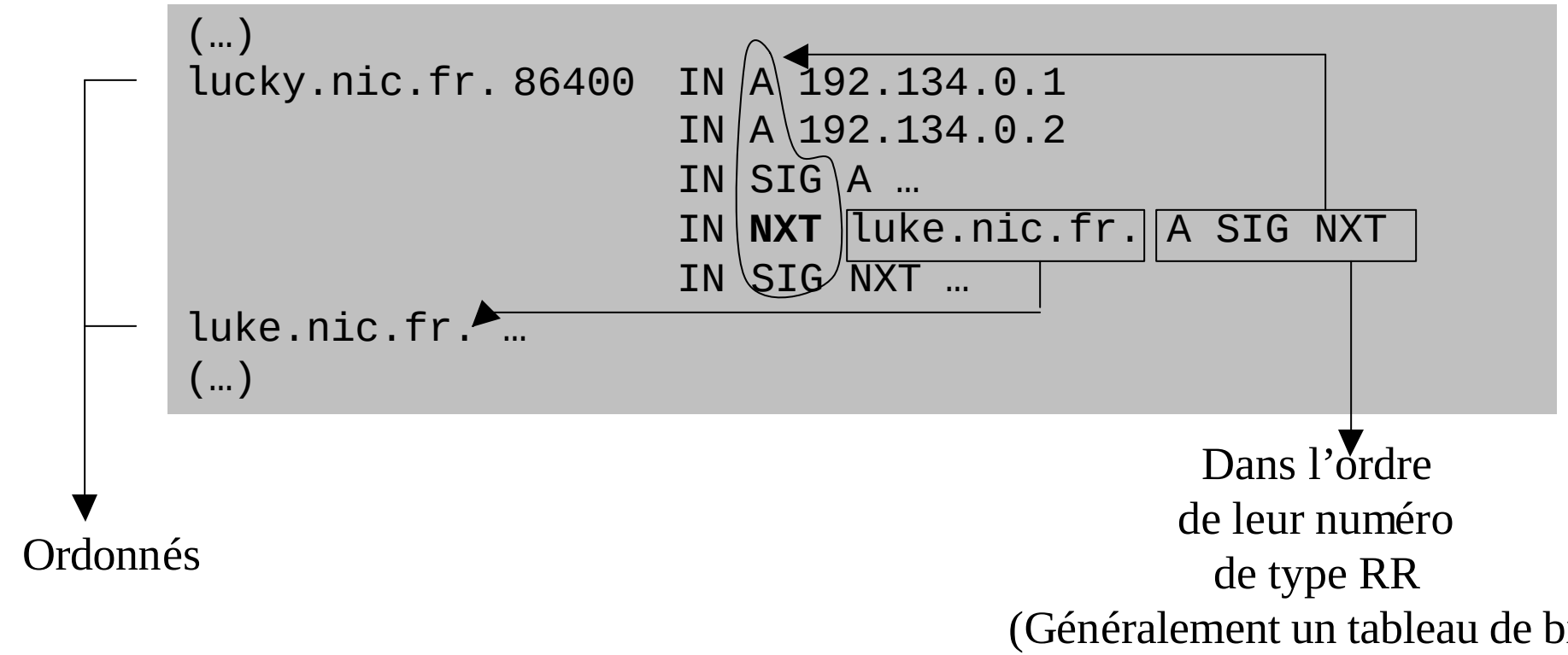
- Le traitement des données, pour fabriquer la signature, dépend de l'algorithme

Le RR NXT
Principe

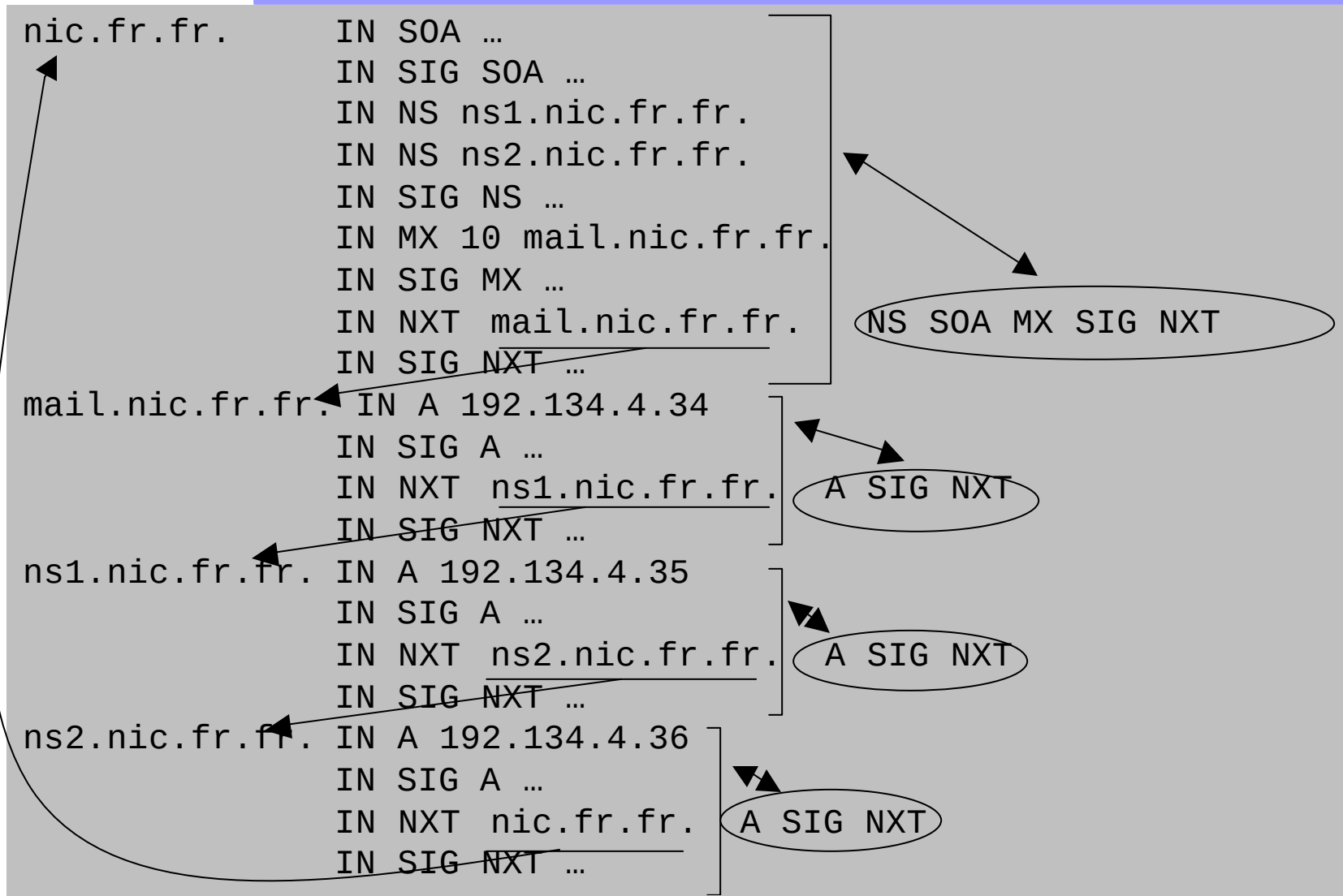
- Le RR NXT répond à la non-existence d'un nom d'une manière sûre en indiquant :
  - L'intervalle d'inexistence
  - Les types de RR existants pour le nom précédent
- La zone est
  - Ordonnée (voir fonction d'ordonnancement plus loin)
  - A ses RRset chaînés en boucle à l'aide des RR NXT
- Exception
  - Sur un point de délégation les glue records (A) et (AAAA) sont exclus de la chaîne NXT dans la zone parente
- Attention
  - Confidentialité : Découverte malveillante de tous les noms contenus dans la zone possible
  - TTL : Le TTL utilisé pour les RR NXT ne doit pas excéder le TTL minimum de la zone
- Exemple

  titi.nic.fr. A ? → tata.nic.fr. IN NXT toto.nic.fr. A SIG NXT

Le RR NXT
Exemple



Chaînage avec le RR NXT
Exemple



Le RR DS
Principe

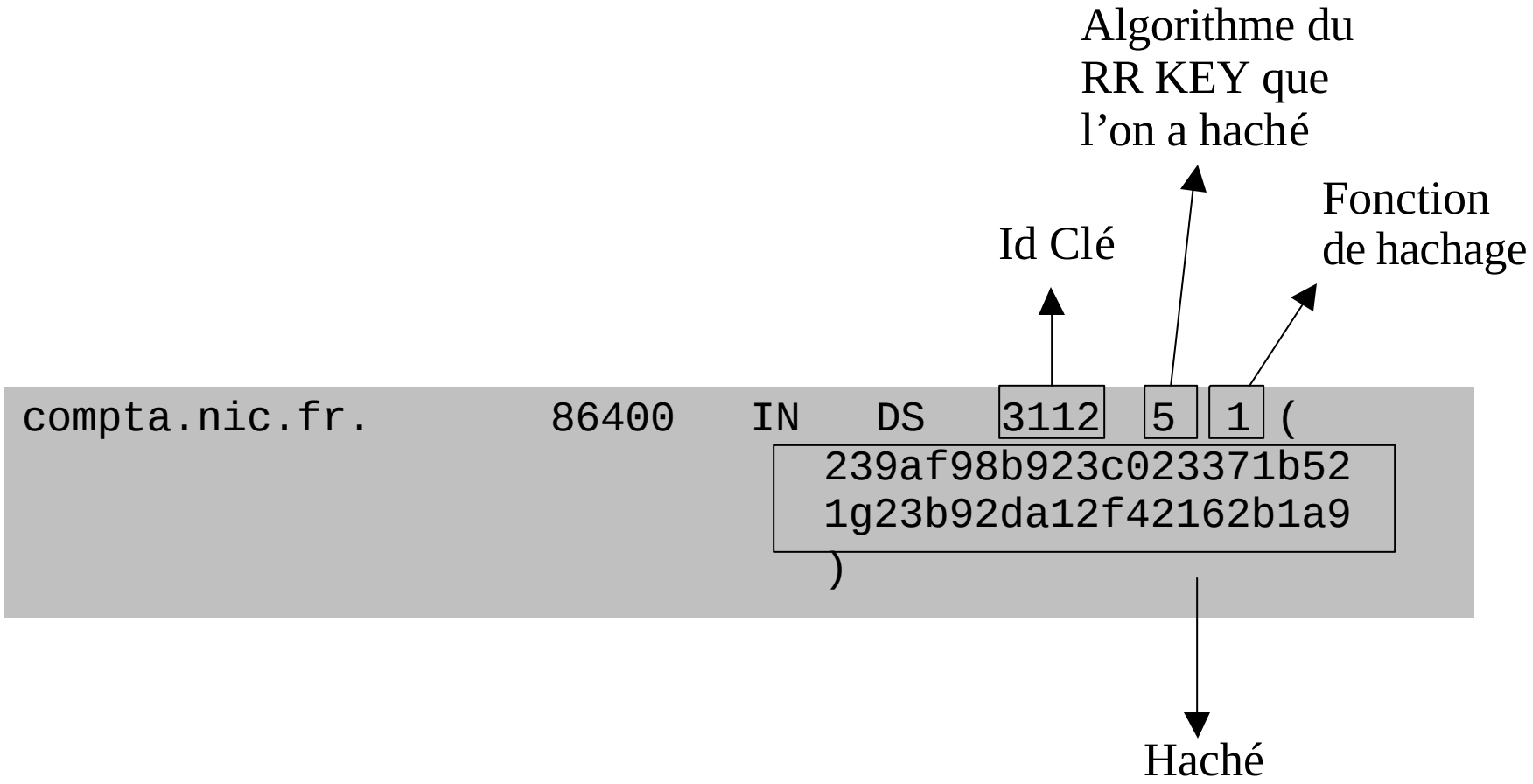
- RR propre au chaînage de confiance [Draft DS]
  Inexistant dans la chaîne de confiance [RFC 2535]

- Porteur d'un haché d'un RR KEY (pas d'un RRset KEY)

  haché= hash (FQDN du RR KEY | Rdata RR KEY)
  Rdata RR KEY=Drapeaux|Protocole|Algorith.|Clé Pub.)

- Fonction de hachage
  - Prévu pour le support de plusieurs fonctions
  - SHA-1 (type 1) est actuellement disponible

Le RR DS
Exemple



Chaîne de confiance
Principes

- Point de départ de la sécurisation
  - Il se trouve
    - A la racine : pour un arbre de nommage entièrement sécurisé
    - En un nœud quelconque de l'arbre : sécurisation du nœud et de son sous-arbre
  - La zone racine d'un ilot de sécurité contient à son sommet un RRset KEY
    - auto-signé
    - accrédité explicitement par les clients

- Un RRset KEY pour une sous-zone sécurisée d'une zone sécurisée
  - Doit être auto-signé
  - Doit être transmise au parent
  - La zone parente construit des RR DS à partir des clefs du RRset KEY

Chaîne de confiance [Draft DS] Principes de fonctionnement

- Sur un point de délégation
  - Une zone parente sécurisée contient pour le nom délégué :
    - Si la zone fille est
      - Sécurisée : Un RRset DS
      - Non Sécurisée : Pas de RRset DS
    - Les NS et glues records (A ou AAAA) non signés (car ils ne font pas autorité dans la zone parente)
  - Une zone fille sécurisée contient:
    - à son sommet (@) un RRset KEY
      - contenant
        - au moins l'une des clés publiques référencées dans le RRset DS pour cette zone dans la zone parente
        - optionnellement d'autres clés
      - signé par au moins une clé dite(s) KSK (pour « Key Signing Key »)
        - présente dans le RRset KEY
        - référencée dans le RRset DS pour cette zone dans la zone parente.
    - tous ses RRset signés par au moins l'une des clés du RRset KEY: clé(s) dite(s) ZSK pour « Zone Signing Key »
  - Une zone fille non sécurisée contient:
    - Aucun changement nécessaire par rapport à une zone standard

Chaîne de confiance [Draft DS]

Exemple de délégation

zone parente : re.re

```
(…)
exemple.re.re.          IN NS ns1.exemple.re.re.
                        IN NS ns2.exemple.re.re.
                        IN DS 3111 …
                        IN SIG … 2134 re.re.
                        IN NXT suite.re.re. NS SIG NXT DS
                        IN SIG NXT … 2134 re.re. …
ns1.exemple.re.re.      IN A 192.134.3.35
ns2.exemple.re.re.      IN A 192.134.4.36
suite.re.re.            (…)
(…)
```

Note : le RRset DS fait autorité dans la zone parente et est donc signé par le parent

Chaîne de confiance [Draft DS]

Exemple de délégation avec une clé unique (KSK & ZSK)

zone fille : exemple.re.re

```
(…)
exemple.re.re.          IN NS ns1.exemple.re.re.
                        IN NS ns2.exemple.re.re.
                        IN SIG NS … exemple.re.re. …
                        IN KEY …
                        IN SIG KEY … 3111 exemple.re.re. …
                        IN NXT ns1.exemple.re.re. NS KEY SIG NXT
                        IN SIG  NXT … exemple.re.re. …
ns1.exemple.re.re.      IN A 192.134.4.35
                        IN SIG A … 3111 exemple.re.re. …
                        IN NXT ns2.exemple.re.re. A SIG NXT
                        IN SIG NXT … 3111 exemple.re.re. …
ns2.exemple.re.re.      IN A 192.134.4.36
                        IN SIG A … 3111 exemple.re.re. …
                        IN NXT autre.exemple.re.re. A SIG NXT
                        IN SIG NXT … 3111 exemple.re.re. …
autre.exemple.re.re. (…)
(…)
```

Id Clé 3111

Chaîne de confiance [Draft DS]

Exemple de délégation sécurisée à clés KSK et ZSK séparées

zone fille : exemple.re.re

```
(…)
exemple.re.re.          IN NS ns1.exemple.re.re.
                        IN NS ns2.exemple.re.re.
                        IN SIG NS … exemple.re.re. …
                        IN KEY …
                        IN KEY …
                        IN SIG KEY … 3111 exemple.re.re. …
                        IN SIG KEY … 7214 exemple.re.re. …
                        IN NXT ns1.exemple.re.re. NS KEY SIG NXT
                        IN SIG  NXT … exemple.re.re. …
ns1.exemple.re.re.      IN A 192.134.4.35
                        IN SIG A … 7214 exemple.re.re. …
                        IN NXT ns2.exemple.re.re. A SIG NXT
                        IN SIG NXT … 7214 exemple.re.re. …
ns2.exemple.re.re.      IN A 192.134.4.36
                        IN SIG A … 7214 exemple.re.re. …
                        IN NXT autre.exemple.re.re. A SIG NXT
                        IN SIG NXT … 7214 exemple.re.re. …
autre.exemple.re.re. (…)
(…)
```

Id Clé 3111 — KSK

ID Clé 7214 — ZSK

Chaîne de confiance [Draft DS]
Avantages

- Peu de contraintes lors de la transmission sécurisée des clés
  - Flux à un seul sens (et non plus à deux sens)
  - Taille limitée (le(s) RR KEY nécessaires à la chaine de confiance, pas tout le RRset KEY)
- Politique flexible de gestion clés pour les zones filles
  - La fille peut se servir de la clé utilisée pour la chaîne de confiance pour signer son RRset KEY *uniquement*
  - La fille ne doit pas informer le parent lors de l'ajout/remplacement de clés ZSK
- Faible volume de données signées avec la clé KSK utilisée pour la chaîne de confiance
  - Plus difficile de compromettre la KSK utilisée pour la chaîne de confiance
  - La ZSK qui signe le reste de la zone peut être changée fréquemment et c'est elle *uniquement* qui est re-signée par la KSK utilisée pour la chaîne de confiance
    - Zones de grandes tailles
    - Zones à contenu changeant fréquemment
- Gestion sous-zones
  - Rien à rajouter si la sous-zone n'est pas sécurisée
  - Si la sous-zone est sécurisée : taille DS (haché) < taille clé publique entière

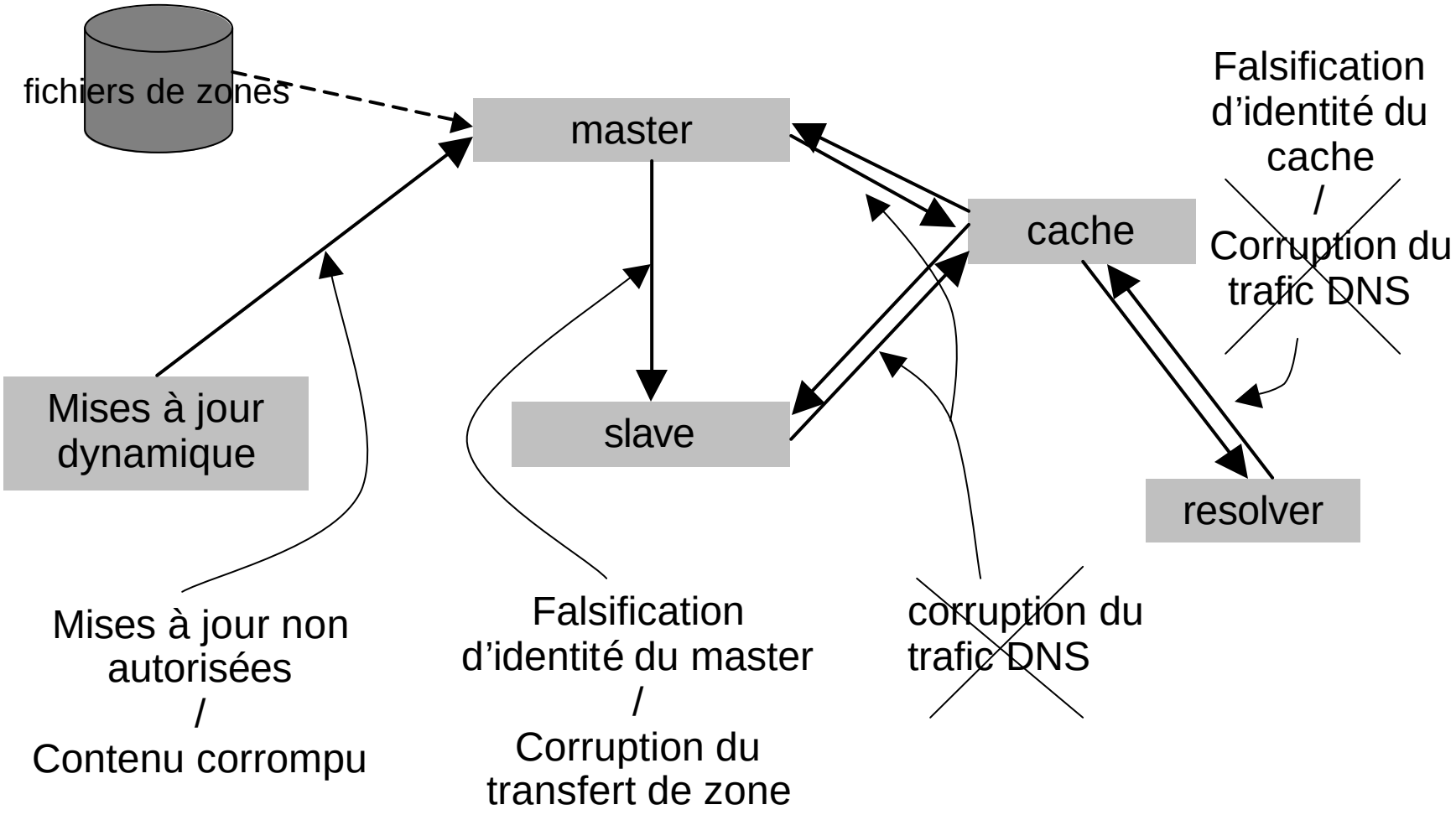
Chaîne de confiance [Draft DS]
Précautions

- Les clés référencées dans le RRset DS chez le parent restent le maillon essentiel de la chaîne de confiance
  - Compromission → Sécurité brisée

- Performances
  - Ce schéma de chaîne de confiance requiert plus de vérifications que rfc 2535
  - Limiter la taille du RRset DS

DNSSEC

Bilan des vulnérabilités résolues et non résolues



Mise en Oeuvre

- Bind
  - Versions
  - Outils
- Expérimentation DNSSEC
- Bilan : Utilisation du DNS Sécurisé

Mise en Œuvre avec Bind (ISC)

- Bind 9.2.x (version de production)
  - Implémentent le [RFC 2535]
  - Keyset (RRset KEY auto-signé) – *Signature du parent* → Signedkey
  - Outils fournis
    - dnssec-keygen: génération de paire de clés
    - dnssec-makekeyset: production d'un keyset par une zone fille
    - dnssec-signkey: signature d'un keyset par le parent → signedkey
    - dnssec-signzone: signature d'une zone (ajout des SIG, NXT)

- Bind 9.3.0s (version de développement)
  - Implémentent le [Draft DS]
  - Nouveau format de keyset: un ou plusieurs RR KEY (non auto-signé)
  - Outils fournis
    - dnssec-keygen: génération de paire de clés
    - dnssec-signzone: signature d'une zone
      - ajout des SIG, NXT
      - ajout des DS en utilisant les keyset des filles
      - génération du keyset adapté pour transmission au parent
      - option spécifiant clé(s) ne signant que le RRset KEY au sommet de la zone (-k)

Expérimentation DNSSEC

- IDsA : **I**nfrastructure **DNSsec et A**pplications
  - Projet RNRT multipartenaires
  - Projet Précompétitif labellisé en 2002
  - *Etude de : IPsec, DNSsec, HIP sur IPv6*
  - URL :
    - *http://www.idsa.prd.fr/*
    - *http://www.telecom.gouv.fr/rnrt/projets/res_02_22.htm*

- Shadow .fr (et .re) :
  - Exemplaire signé non référencé dans les « root-servers »
  - Généré auto-magiquement toutes les nuits par Autosign-TLD
  - Draft-DS
  - Opérationnel 24h/24h
  - Référencé dans OTDR

Expérimentation internationnale : OTDR

- Integration dans l'expérimentation DNSSEC internationnale :
  - OTDR (**O**perational **T**estbed for new **D**NS features in the Internet **R**oot)
    - Bill Manning / ISI
    - IANA
  - Étude de l'impact des nouvelles technologies DNS dans les « root-servers »
    - DNSSEC
    - IPv6
    - IDN
  - Participants : `.fr .nl .kr .se .jp .com`
  - URL : `http://www.rs.net/`

Bilan
Utilisation du DNS Sécurisé

- TSIG
  - Utilisable dès maintenant
    - Les transferts de zone (master / slaves)
    - Les mises à jour dynamiques
    - Les transactions entre resolvers et caches (si l'on dispose du resolver adapté!)
  - Limitation dans le nombre d'acteurs (passage à l'échelle)

- DNSSEC
  - Seule possibilité actuellement :
    - Ilots sécurisés
    - Configuration manuelle des serveurs récursifs clients pour les « trusted-keys » des ilots distants
  - Objectif
    - Une seule racine sécurisée
    - Une clé « trusted-key » correspondante et disponible à tous. (Incluse dans la distribution)

DNSsec pour IPsec

- Certificat X.509 dans le DNS (CERT RR - [RFC 2538])
  - Implémentation : Demon IKE Racoon
- Opportunistic encryption (draft IETF janvier 2003)
  - on déclare la capacité IPsec dans DNS
    - RR TXT

```
IN TXT "X-IPsec-Server(10)=192.1.1.5 AQMM...3s1Q=="
```

  - mise à disposition des clefs
    - RR KEY contenant la clef publique de la machine dans le reverse DNS
  - la (non-)existence et la validité sont garantis par DNSsec
  - Implémentation : FreeS/WAN

Sources et Remerciements

- IETF
  - RFC 2535 : DNS Security Extensions
  - Draft-DS : draft-ietf-dnsext-delegation-signer-13 (Delegation Signer Record)
  - RFC 2845 : Secret Key Transaction Authentication for DNS (TSIG)
- DNS et BIND 4ème Edition (O'Reilly 2002)
- Remerciements
  - Olaf Kolkman (cours DNSSEC RIPE)
  - Miek Gieben (NLnetLabs)
- Rédacteurs
  - Nicolas Notari (stagiaire AFNIC 2002)
  - Jean Philippe Pick / Mohsen Souissi (AFNIC)
  - Bernard Cousin / Olivier Courtay (Irisa)

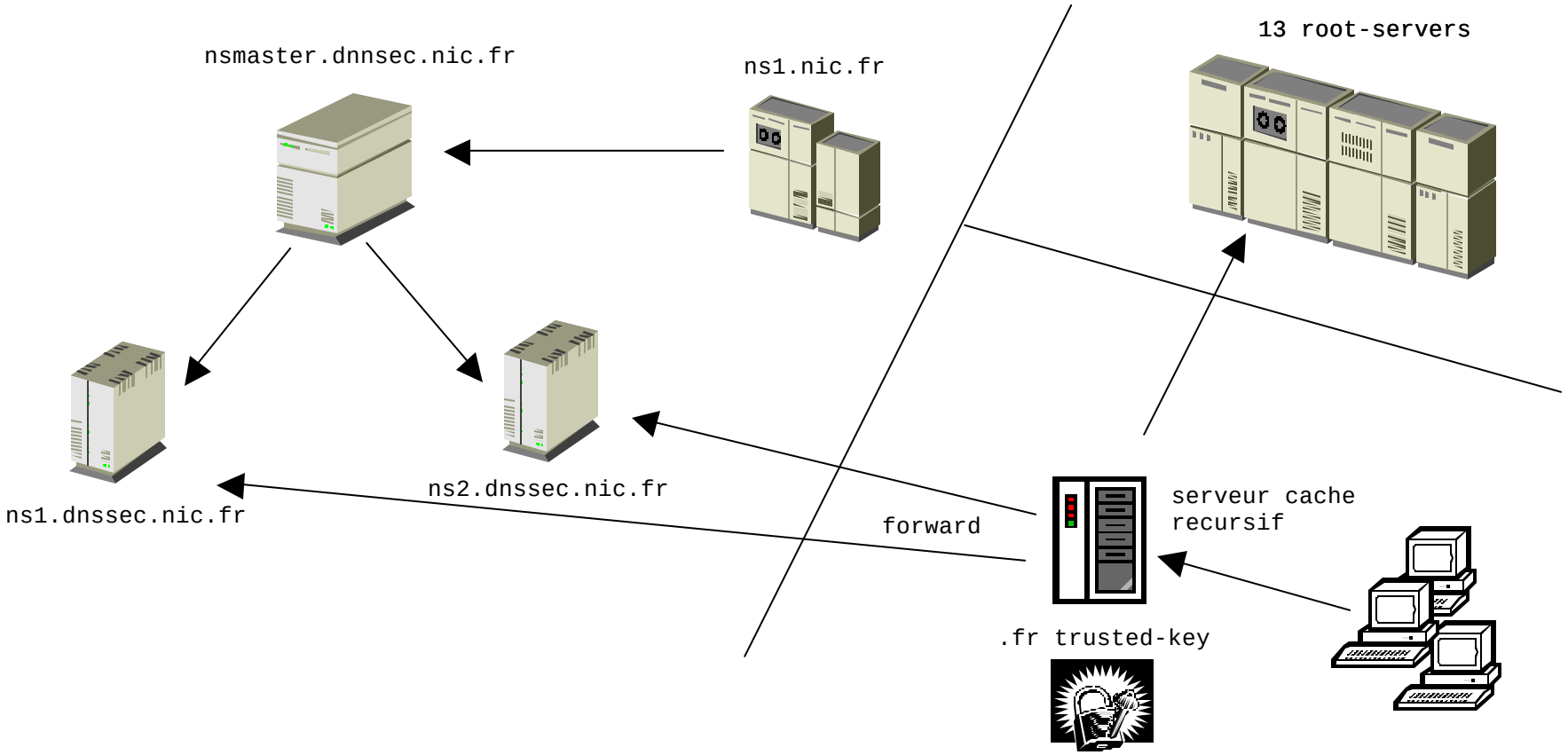
Questions

Clés d'Applications
Le RR APPKEY

- Les clés publiques d'applications peuvent être stockées dans le RR KEY mais :
  - Les Drapeaux représentent des informations qui dans ce cas peuvent être dépendantes de l'application
  - Le champ Protocole est limité à 8 bits
  - Nécessité une nouvelle signature par le parent si modification du RRset KEY au sommet de la zone [RFC2535]
- Un RR APPKEY spécifique aux clés publiques d'applications
  On se limite à stocker :
  - Algorithme
  - Clé publique
- Ne stocke pas les certificats :
  Un autre RR spécifique (CERT) existe [RFC 2538]
- Question : Le DNS doit-il réellement être utilisé comme PKI?

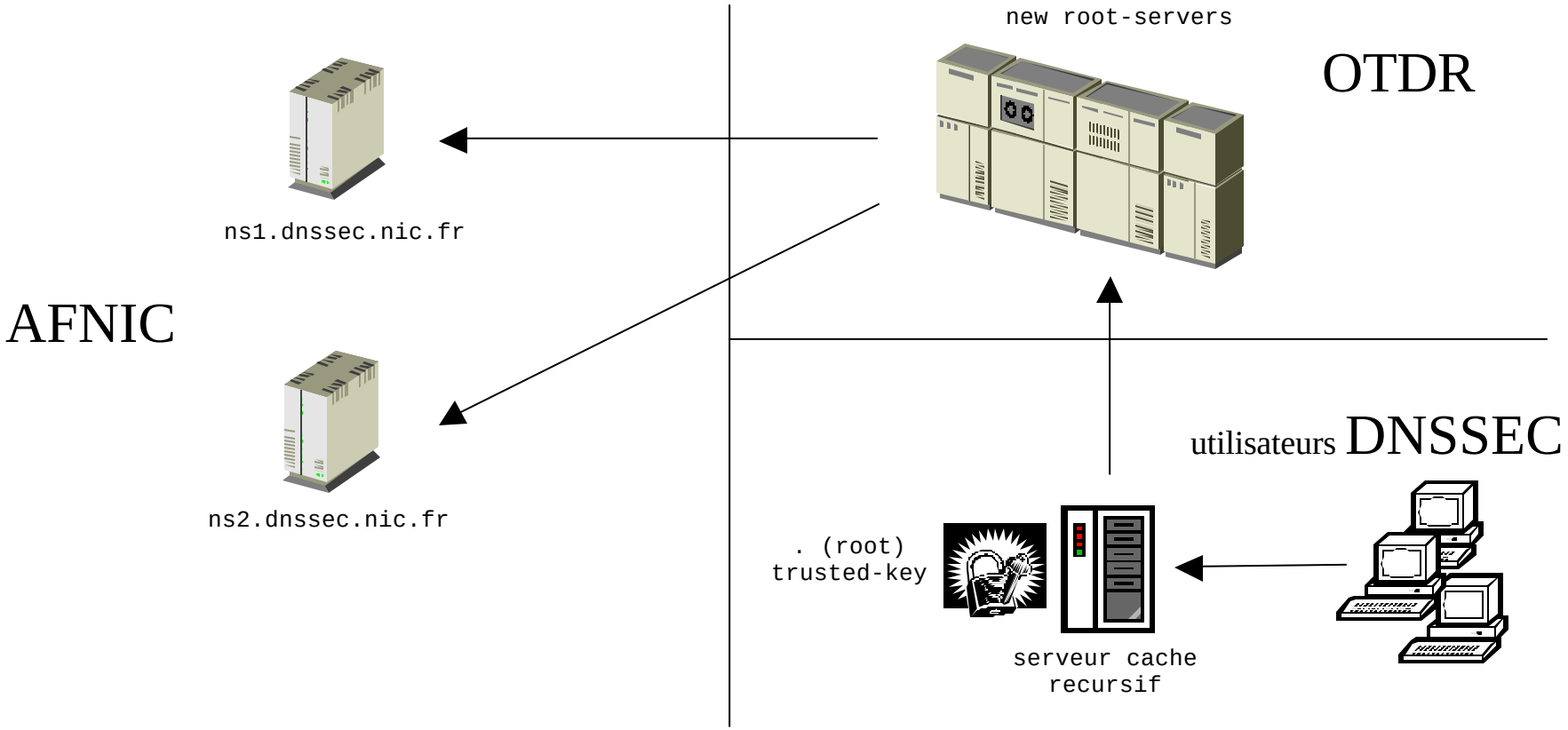
Expérimentation DNSSEC à l'AFNIC : Autosign-TLD

- Architecture :



Expérimentation DNSSEC internationnale : Autosign-TLD et OTDR

- Architecture :



Ordonnancement des RR

- Nécessité d'ordonnancer (RR NXT et SIG)
- Algorithme pour l'ordonnancement de la chaîne des NXT
  - Entre noms différents
    - Noms étendu en « FQDN »
    - Caractères minuscules
    - Ordre alphabétique avec priorité à l'absence de caractère
    - Ordonnancement selon labels de plus haut niveau (droite)
      Si labels identiques à ce niveau: on utilise le label de niveau inférieur, etc …
  - Entre RRset d'un même nom
    - Ordre des numéros de type de RR sauf pour le SIG qui suit le RRset signé
  - Entre RR d'un même RRset
    - Ordre des Rdata (alphabétique avec priorité à l'absence de caractère)
- Algorithme pour l'ordonnancement nécéssaire à SIG
  - Uniquement le point « Entre RR d'un même RRset »
    de l'algorithme précédent