Architectures DNS sécurisées

Guillaume Valadon, Yves-Alexis Perez

Agence Nationale de la Sécurité des Systèmes d'Information

Contexte

- DNSSEC est en cours de déploiement
  1. zone racine signée depuis Juillet 2010
  2. 70 TLDs (com, net, fr, org ...) actuellement signés
  3. de plus en plus de bureaux d'enregistrement supportent DNSSEC

Objectifs

1. comprendre les différentes technologies
2. identifier ce qui fonctionne aujourd'hui
3. étudier une solution alternative utilisant LDAP

ANSSI

Le protocole DNS (Domain Name System)

"Le DNS permet de trouver l'adresse IP associée à un nom."

En pratique, c'est plus compliqué :

- le protocole DNS définit le format de requête et de réponse portant sur un nom et un type de données
- il existe différents types des données : A, AAAA, MX, NS, SOA ...
- les données sont stockées sous forme d'*enregistrements* de la forme

- les données sont réparties sur une hiérarchie de serveurs
  - localement, les données sont généralement stockées dans un fichier de zone

ANSSI

Le protocole DNS (Domain Name System)

"Le DNS permet de trouver l'adresse IP associée à un nom."

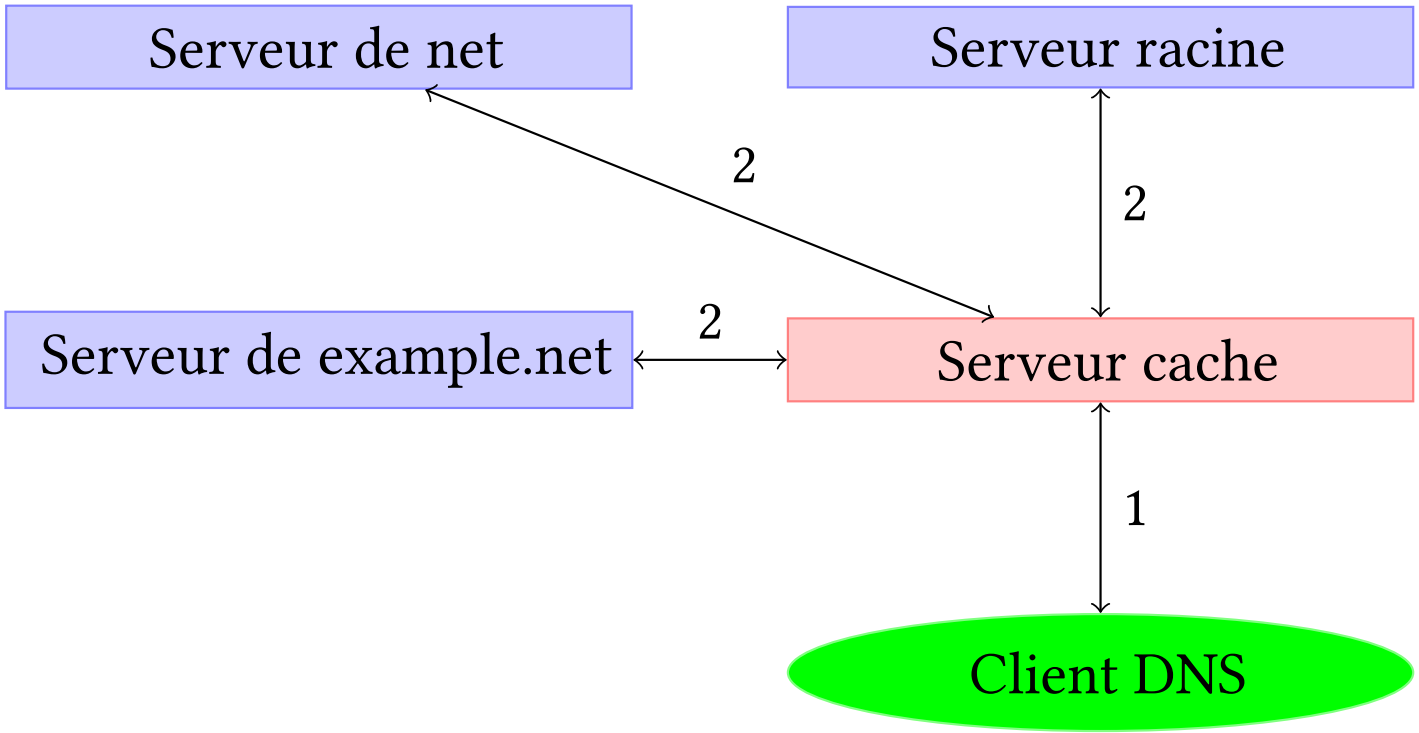
En pratique, c'est plus compliqué :

- le protocole DNS définit le format de requête et de réponse portant sur un nom et un type de données
- il existe différents types des données : A, AAAA, MX, NS, SOA ...
- les données sont stockées sous forme d'*enregistrements* de la forme

```
example.net.      10 NS   ns1.example.net.
ns1.example.net.  42 A    198.51.100.1
pub.example.net.  80 TXT  L'ANSSI recrute !
```

- les données sont réparties sur une hiérarchie de serveurs
  - localement, les données sont généralement stockées dans un fichier de zone

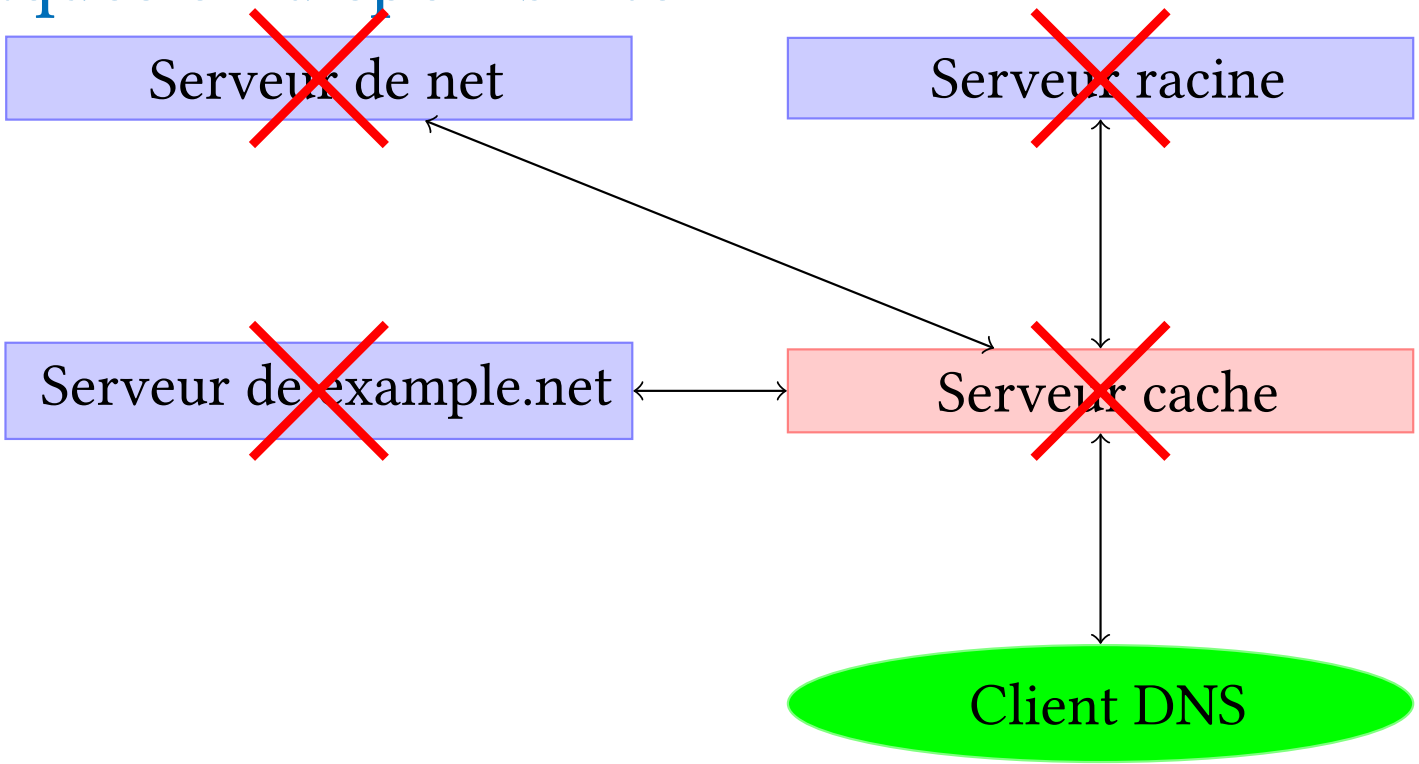
Une résolution DNS



Le client cherche à trouver l'enregistrement A de *www.example.net*.

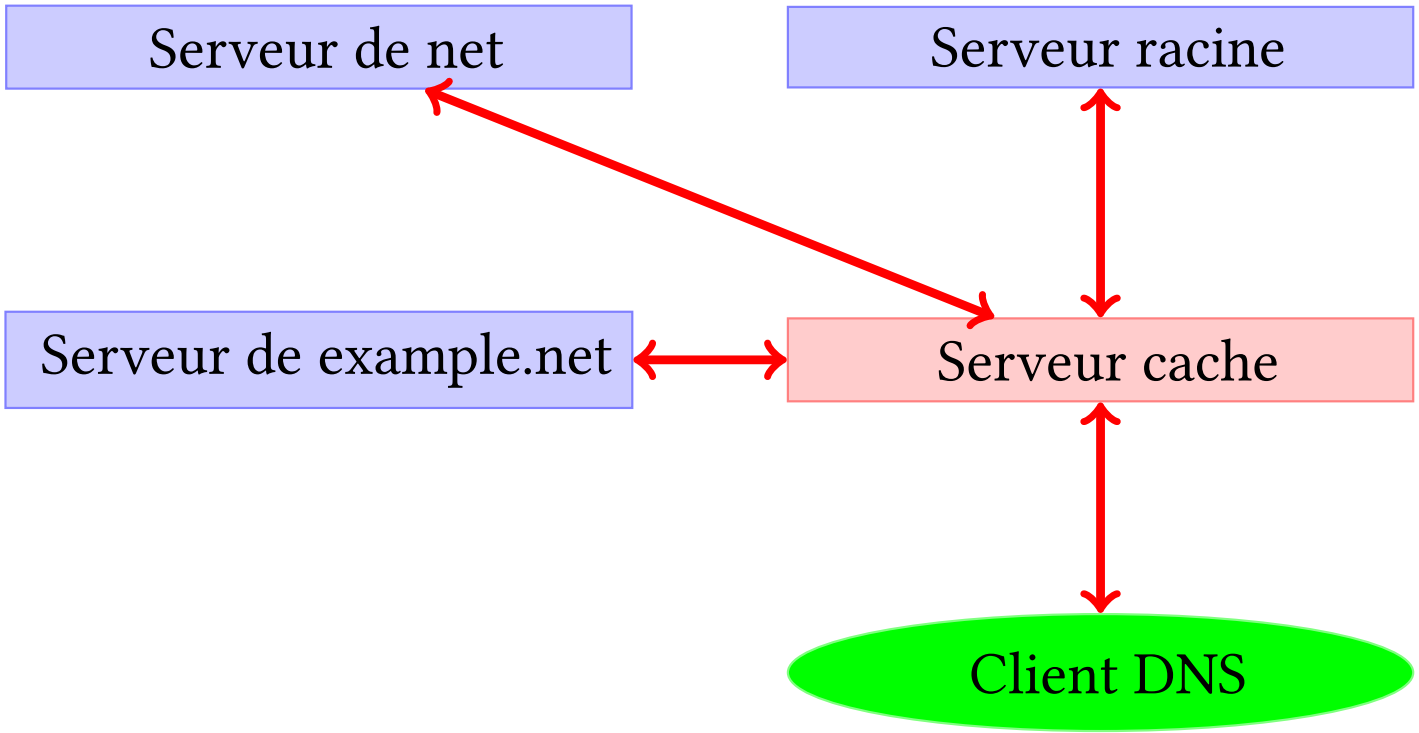
1. il envoie une requête à son serveur cache
2. le serveur cache interroge des serveurs faisant autorité
   - en pratique, le serveur cache pose la même question à tous les serveurs qu'il contacte

Attaques en disponibilité



Cibles	serveurs cache et serveurs qui font autorité
Causes	déni de service, problème d'implantation ...
Impacts	serveurs injoignables
Solutions	serveurs DNS esclaves, anycast ...

Corruption des données



Cibles les communications du serveurs cache & des clients

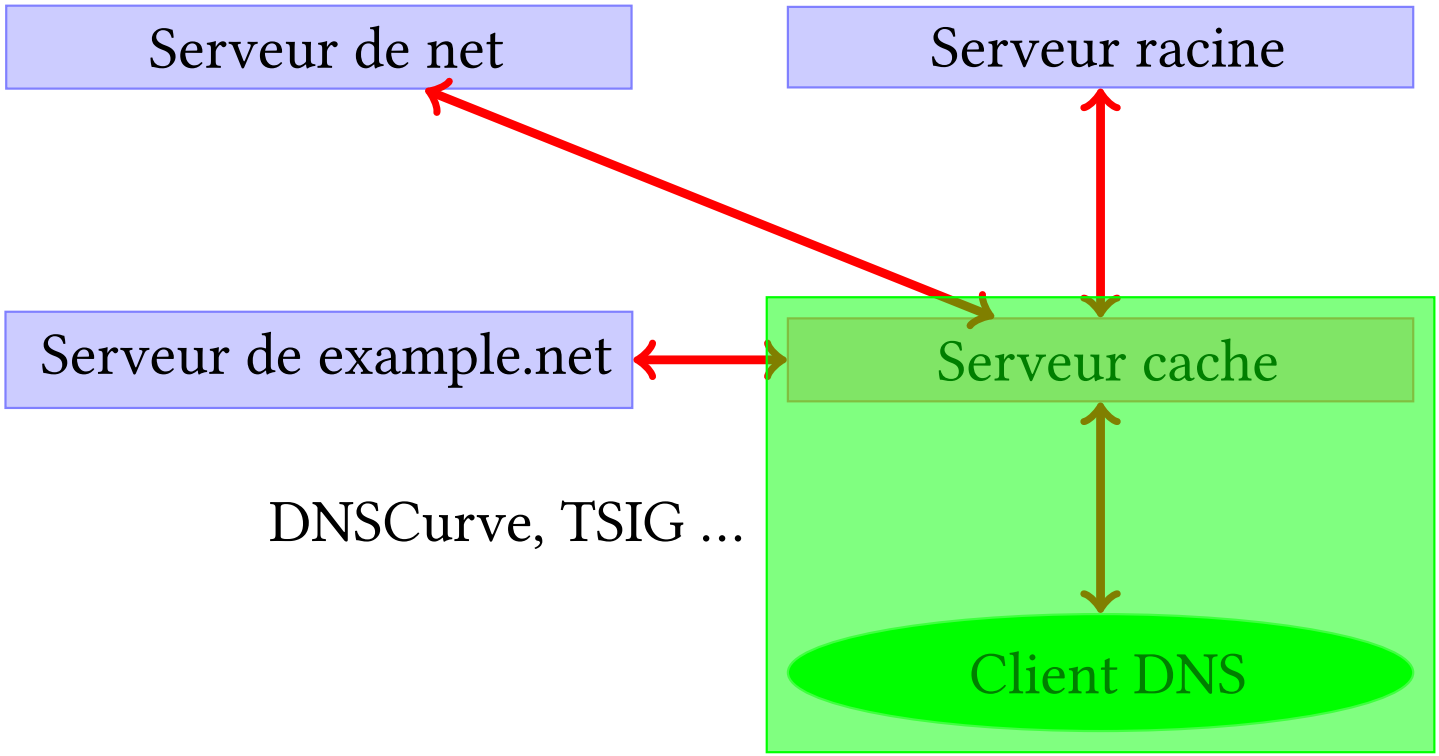
Causes DNS menteur, empoisonnement de cache ...

Impacts les données reçues / stockées sont fausses

Solutions DNSSEC, TSIG, DNSCurve ...

ANSSI

Corruption des données



Cibles les communications du serveurs cache & des clients

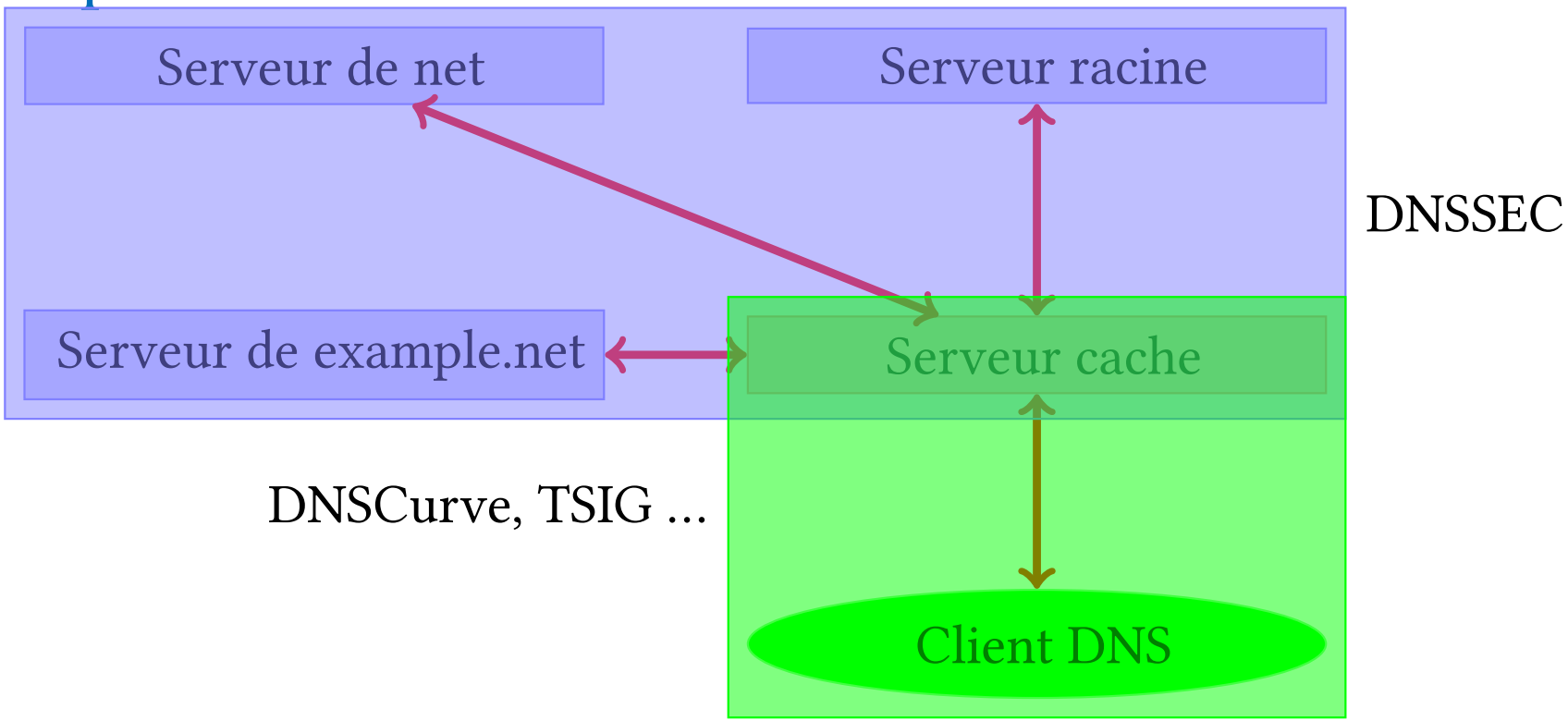
Causes DNS menteur, empoisonnement de cache ...

Impacts les données reçues / stockées sont fausses

Solutions DNSSEC, TSIG, DNSCurve ...

ANSSI

Corruption des données



Cibles les communications du serveurs cache & des clients

Causes DNS menteur, empoisonnement de cache ...

Impacts les données reçues / stockées sont fausses

Solutions DNSSEC, TSIG, DNSCurve ...

DNSSEC : DNS Security Extensions - RFC 4034

"DNS avec des signatures et une architecture de clés."

En quelques mots

- signature des enregistrements avec la clé privée du serveur
  - protection des données en intégrité
- seuls les serveurs DNS cache vérifient les signatures
  - pas de surcout pour les serveurs qui font autorité
- les clés privées peuvent être stockées hors ligne
  - un serveur travaille avec des fichiers de zone pré signés
- hiérarchie de clés calquée sur celle des zones

ANSSI

L'architecture de clés

Infrastructure de clés X.509 (utilisée avec TLS)

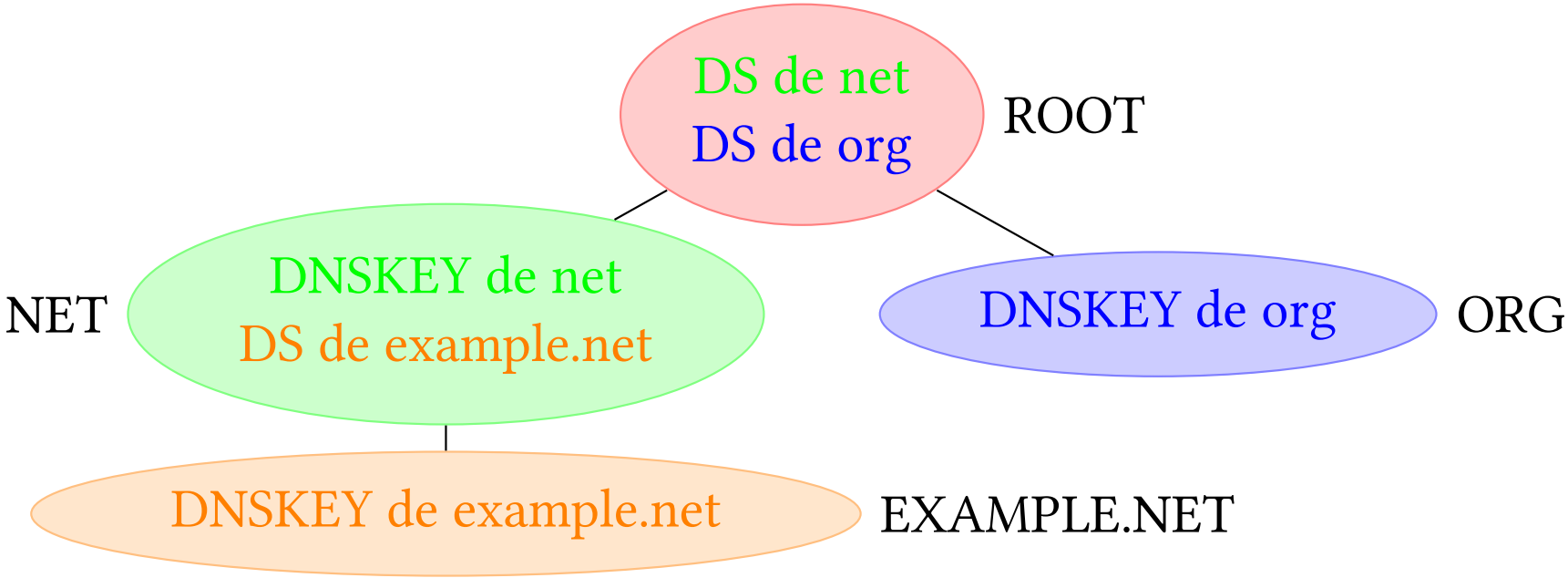
1. racines multiples : toutes les AC des navigateurs
   - une AC peut signer n'importe quel certificat
2. certificats : signature de la clé publique
3. révocation : temps, CRL, OCSP ...

DNSSEC

1. une seule racine : la zone racine
   - une zone peut uniquement signer ses sous-zones
2. deux nouveaux enregistrements jouent le rôle des certificats :
   - DNSKEY : la clé publique associée à une zone
   - DS (*Delegation Signer*) : l'empreinte cryptographique de la clé
3. révocation : temps

ANSSI

La hiérarchie des clés



- Pour la zone `example.net`:
  - le DNSKEY est présent dans la zone
    `example.net. DNSKEY 256 3 7 (AwE .. 56z)`
  - le DS est poussé à la zone supérieure via le bureau d'enregistrement
    `example.net. DS 4889 7 2 23B .. ABB`
- Un serveur cache validant doit connaître le DNSKEY de la zone racine

Les signatures des enregistrements

- elles sont présentes dans des enregistrements RRSIG (Resource Record SIGnature)
- elles permettent de vérifier la validité des enregistrements signés
- elles contiennent notamment :
  - les dates de début et de fin de validité de la signature
  - des indications sur la clé ayant signé le RRSIG

```
$ dig a.example.net. A

a.example.net. A 198.51.100.1
               A 198.51.100.2
               RRSIG A 5 3 86400 20110428115700 20110329115700
               4889 example.net. Qfise [..] 7efg==
```

L'enregistrement NSEC (Next SECure)

But : signer des enregistrements qui n'existent pas

Méthode :

1. classer tous les enregistrements de la zone
2. créer un enregistrement NSEC par intervalle

   ```
   a.example.net. NSEC d.example.net. A RRSIG NSEC
   ```

3. signer ces enregistrements NSEC

Problème l'énumération de la zone devient triviale car NSEC expose les noms des enregistrements et leurs types.

ANSSI

Énumération NSEC à la main

```
$ dig b.example.net A

a.example.net.  IN NSEC d.example.net. A RRSIG NSEC
a.example.net.  IN RRSIG NSEC [..]
```

Énumération NSEC à la main

```
$ dig b.example.net A

a.example.net.  IN NSEC d.example.net. A RRSIG NSEC
a.example.net.  IN RRSIG NSEC [..]


$ dig d.example.net NSEC

d.example.net.  IN NSEC m.example.net. CNAME RRSIG NSEC
d.example.net.  IN RRSIG NSEC [..]
```

ANSSI

Énumération NSEC à la main

```
$ dig b.example.net A

a.example.net.  IN NSEC d.example.net. A RRSIG NSEC
a.example.net.  IN RRSIG NSEC [..]


$ dig d.example.net NSEC

d.example.net.  IN NSEC m.example.net. CNAME RRSIG NSEC
d.example.net.  IN RRSIG NSEC [..]



$ dig m.example.net NSEC

m.example.net.  IN NSEC mx0.example.net. CNAME RRSIG NSEC
m.example.net.  IN RRSIG NSEC [..]
```

ANSSI

L'enregistrement NSEC3 - RFC 5155

Même méthode que NSEC mais

1. une requête ne peut pas porter sur un enregistrement NSEC3
2. empreintes cryptographiques utilisées à la place des noms

Etant donné une fonction de hachage, un sel et des itérations

1. le client envoie une requete portant sur nexistepas.example.net
2. le serveur répond :

```
lvvgq44a75ecklia53o50qfcstpto6ob.example.net. NSEC3 1 0 1
                414243 yp667bc9imetnnakj9mj9s8vm2dtb90e CNAME RRSIG
```

3. le client peut vérifier l'empreinte de nexistepas.example.net (`t2383b9ih85v0gj3u3epkot1nv6vqjkv`) est encadrée par le NSEC3

Problème : il est toujours possible d'énumérer la zone mais il faut pré-calculer des empreintes.

Énumération NSEC3 assistée

```
$ dig unknown.example.net.

DJK7HR5J4BNPJU6O6JFIUMAGI8VBNGBR.example.net. NSEC3 \
N68K5M8HOGBM8HFPR6M4IIKVEL0N6SKR [..]
```

Énumération NSEC3 assistée

```
$ dig unknown.example.net.

DJK7HR5J4BNPJU6O6JFIUMAGI8VBNGBR.example.net. NSEC3 \
N68K5M8HOGBM8HFPR6M4IIKVEL0N6SKR [..]


$ dig vs.example.net # n6ic6st3l2cqktd0oafg4g6tsgq3kt9r.example.net.

N68K5M8HOGBM8HFPR6M4IIKVEL0N6SKR.example.net. NSEC3 \
PQBHJPPONI6S4H7L8S3PG8RU5IKIFEEC [..]
```

Énumération NSEC3 assistée

```
$ dig unknown.example.net.

DJK7HR5J4BNPJU6O6JFIUMAGI8VBNGBR.example.net. NSEC3 \
N68K5M8HOGBM8HFPR6M4IIKVEL0N6SKR [..]


$ dig vs.example.net # n6ic6st3l2cqktd0oafg4g6tsgq3kt9r.example.net.

N68K5M8HOGBM8HFPR6M4IIKVEL0N6SKR.example.net. NSEC3 \
PQBHJPPONI6S4H7L8S3PG8RU5IKIFEEC [..]


$ dig x6.example.net. # pr06tg4crrt70nhkoef8vgr02h7af95k.example.net.

PQBHJPPONI6S4H7L8S3PG8RU5IKIFEEC.example.net. NSEC3 \
5353544943 RBAVNBTBKJFJ3HQJNFMVOHF3GREGJ93V [..]
```

ANSSI

DNSSEC : les implantations

Fichiers de zone pré signés : le modèle DNSSEC classique

- prévu pour les serveurs DNS très sollicités
- la clé privée peut être stockée dans un coffre
- l'énumération de la zone est possible

Serveurs bind & powerdns

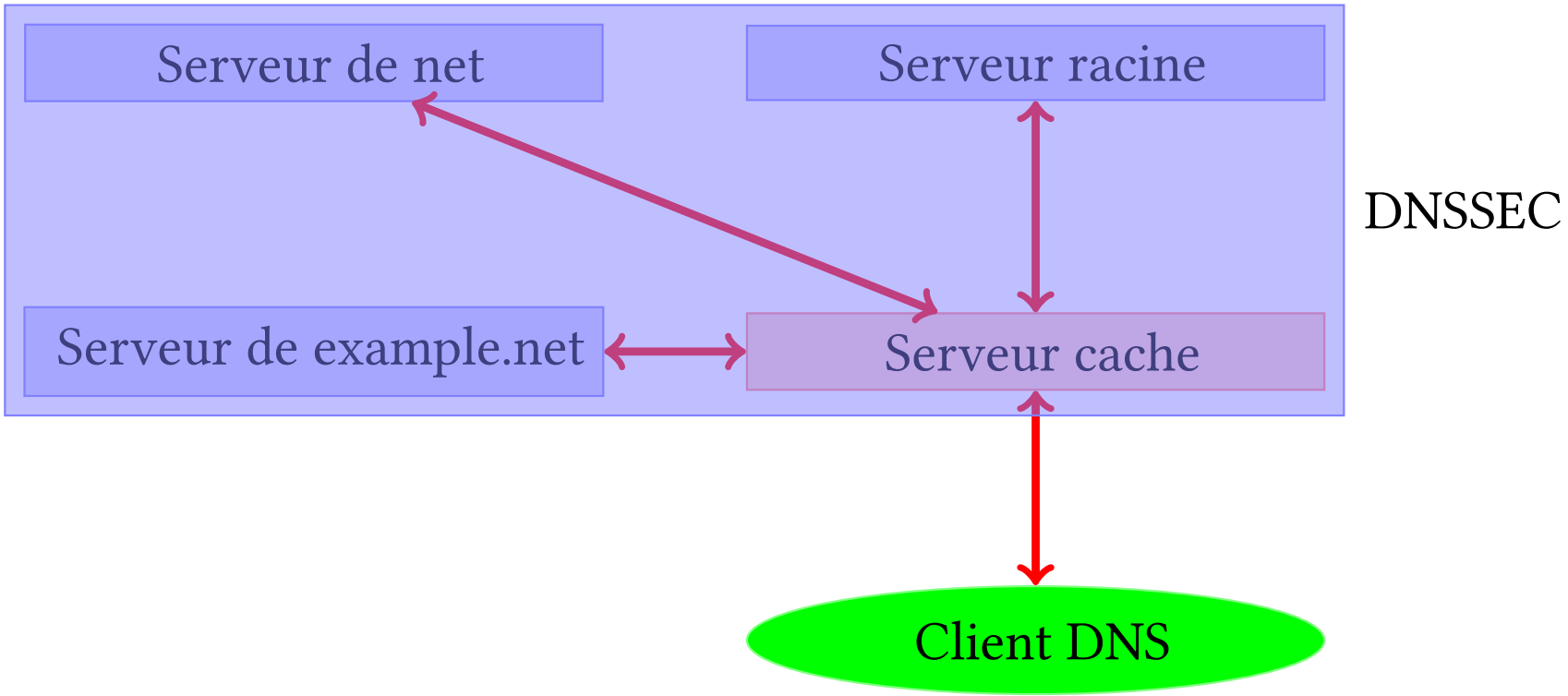
Signature en ligne

- induit une charge supplémentaire
- la clé privée est présente sur le serveur DNS
- l'énumération peut être supprimée avec le NSEC-narrow
  - le serveur signe dynamiquement des NSEC réduits portant sur le nom demandé

Serveurs powerdns & phreebird

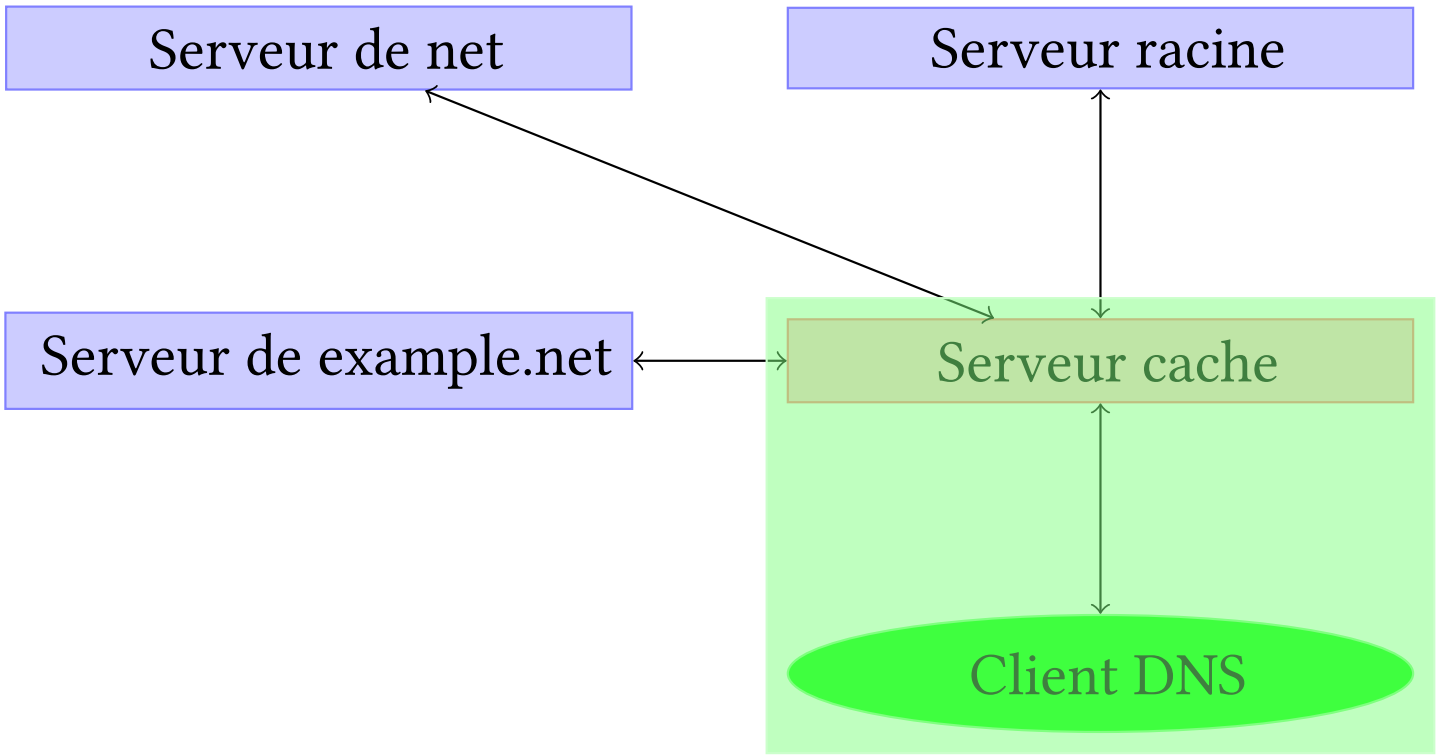
ANSSI

En résumé



- protection efficace des données en intégrité
- architecture de clés alternatives
- problèmatiques de gestion de clé, de temps, d'énumration ...
- utilisable dès aujourd'hui

Le dernier lien ?



Spécificités

- entre un client et son serveur cache (FAI, *hotspot*, entreprise etc.)
- canal physique peu protégé
- au plus proche de l'utilisateur final
- beaucoup d'information dans les requêtes DNS

TSIG : *Transaction SIGnature* - RFC 2845 (mai 2000)

Constat

- DNSSEC intéressant jusqu'au serveur cache
- trop gourmand sur les *stub resolvers* de l'époque

TSIG a été conçu pour compléter DNSSEC

- TSIG protège le lien : authentification du serveur
- cryptographie symétrique
- pas besoin de garder des clés en cache
- protection du lien jusqu'au cache, qui utilise DNSSEC

ANSSI

TSIG : concrètement

- identification des transactions par l'ajout d'un motif d'intégrité aux messages
- calcul d'un HMAC à l'aide d'une clé symétrique partagée

Quelques inconvénients

- gestion des clés secrètes, pas de solution préconisée dans la RFC
- uniquement intégrité, pas de chiffrement
- originalement HMAC-MD5 (maintenant jusqu'à SHA2)
- utilisé uniquement pour la synchronisation et les mises à jour dynamiques
- pas d'implémentation dans les *stub-resolvers*

ANSSI

TSIG : exemple

```
$ dig ANY www.ssi.gouv.fr @192.168.24.8

;; Got answer :
;; ->>HEADER<<- opcode : QUERY, status : REFUSED, id : 29215
;; flags : qr rd ; QUERY : 1, ANSWER : 0, AUTHORITY : 0, ADDITIONAL : 0
;; WARNING : recursion requested but not available

;; QUESTION SECTION :
;www.ssi.gouv.fr. IN ANY

$ dig -y HMAC-SHA512 :stub-key :dGhlYW5zd2VyaXM0MiE= ANY www.ssi.gouv.fr

;; ANSWER SECTION :
www.ssi.gouv.fr. 80 IN TXT "80 postes en 2012"
www.ssi.gouv.fr. 80 IN TXT "L'ANSSI recrute !"
www.ssi.gouv.fr. 80 IN TXT "http ://www.ssi.gouv.fr/fr/anssi/emploi"
www.ssi.gouv.fr. 80 IN A 192.51.100.80
```

ANSSI

DNSCurve

Objectif protéger les messages DNS (intégrité, confidentialité)

En résumé

- courbe elliptique de D. J. Bernstein
- message spécifique sur UDP ou dans un enregistrement TXT
- protection opportuniste
  - la clé publique du serveur est publiée dans un enregistrement NS
  - seul le serveur est authentifié ; la clé publique du client est dans la requête

Problèmes

- données pas protégées : empoisonnement de cache possible
  - c'est une alternative à TSIG, pas à DNSSEC
- pas de prise en compte des configurations maîtres & esclaves
- révocation de clé uniquement par le bureau d'enregistrement

ANSSI

Résolution de nom à l'aide d'un annuaire LDAP

Contexte : Intranet, réseau fortement maîtrisé, plate-forme GNU

Comment ?

- méthode `ldap` dans `nsswitch.conf` (bibliothèque `nss-ldapd`)
- *daemon* `nscld` fait des requêtes LDAP standards

Intérêt

- support natif de TLS, intégration à l'IGC
- authentification via l'annuaire du système d'information

Inconvénients

- plus coûteux que DNS (réseau et serveurs)
- gestion d'une IGC et d'un annuaire

ANSSI

Conclusion

DNSSEC

- une solution efficace pour protéger les enregistrements
- complexifie la gestion d'une zone avec le renouvellement des clés et des signatures
- offre une architecture de clés alternative intéressante pour stocker des empreintes de clés d'autres protocoles : SSH & TLS

Dernier lien

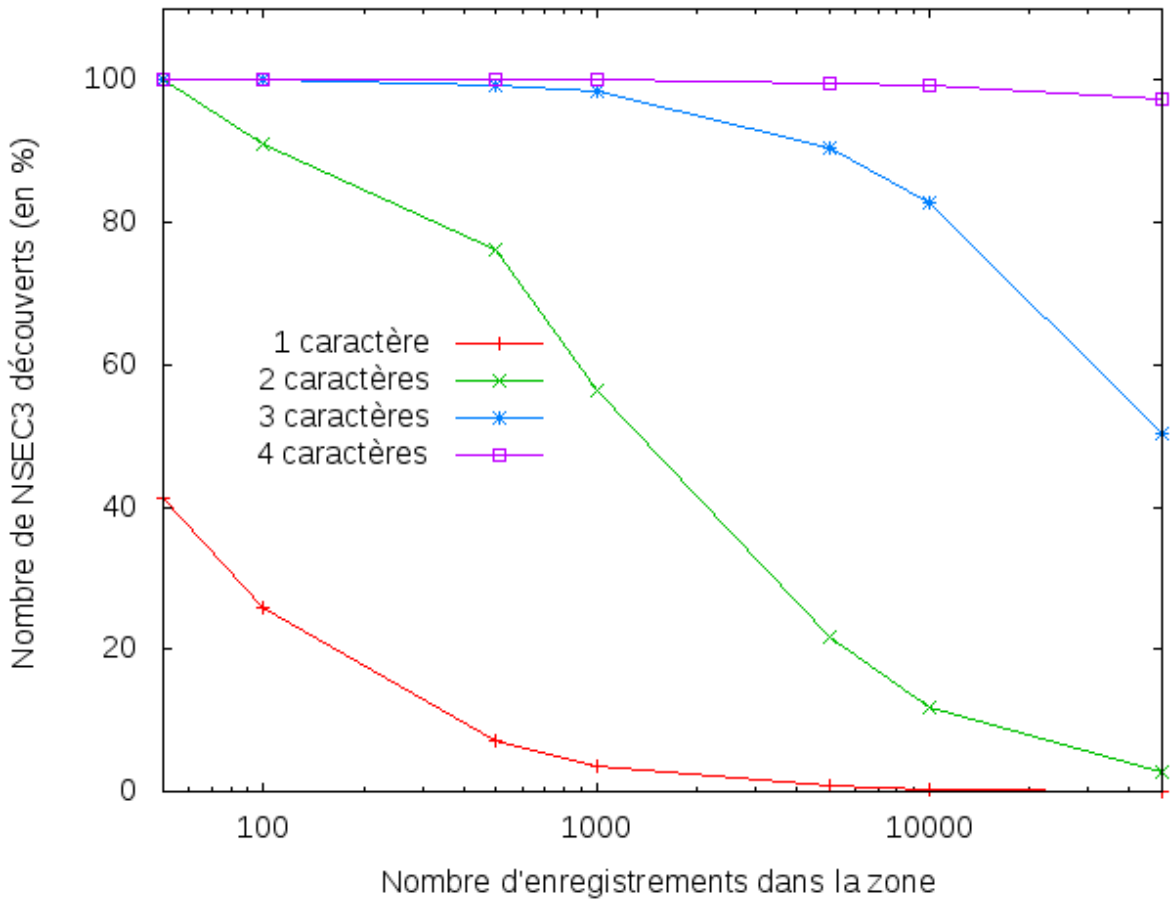
- nombreuses solutions théoriques : TSIG, TKEY, DNSCurve, DTLS, TCP ...
- en pratique seul le couple TSIG/libnss est utilisable sous Linux
  - des proxy DNS locaux sont envisageables pour les autres OS
- solution alternative avec un serveur DNSSEC validant sur chaque machine

ANSSI

RFC 2845 (TSIG)

> 1.2. One difficulty with the [RFC2535] scheme is that common DNS implementations include simple "stub" resolvers which do not have caches. Such resolvers typically rely on a caching DNS server on another host. It is impractical for these stub resolvers to perform general [RFC2535] authentication and they would naturally depend on their caching DNS server to perform such services for them. To do so securely requires secure communication of queries and responses. [RFC2535] provides public key transaction signatures to support this, but such signatures are very expensive computationally to generate. In general, these require the same complex public key logic that is impractical for stubs. This document specifies use of a message authentication code (MAC), specifically HMAC-MD5 (a keyed hash function), to provide an efficient means of point-to-point authentication and integrity checking for transactions.

L'énumération NSEC3 est-elle difficile ?



CVE de bind & DNSSEC

1	CVE-2011-1910	26/05/2011	Large RRSIG RRsets and Negative Caching can crash named
2	CVE-2011-1907	05/05/2011	RRSIG Queries Can Trigger Server Crash When Using Response Policy Zones
3	CVE-2010-3613	01/12/2011	BIND : cache incorrectly allows a ncache entry and a rrsig for the same type
4	CVE-2010-0213	15/07/2011	RRSIG query handling bug in BIND 9.7.1

l'ANSSI recrute

http ://www.ssi.gouv.fr/fr/anssi/emploi

ANSSI