Analyse du système de vote en ligne Neovote

Enka Blanchard

CNRS,
Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (UPHF),
Centre Internet et Société (CNRS)

www.koliaza.com

Travail commun avec
Antoine Gallais, Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (UPHF)
Emmanuel Leblond, Scille SAS
Djohar Sidhoum-Rahal, Centre de Droit Pénal et de Criminologie, Université Paris Nanterre
Juliette Walter, Unite Live

24ème Rencontres Francocophones sur les Aspects Algorithmiques des Télécommunications
ALGOTEL 2022 1/06/2022

Neovote (l'entreprise)

Neovote :

- Fondée en 2007, discrète jusqu'à 2017
- Acteur marcheur du marché français (10k votes par an, 245+ clients publics et privés)
- Au moins 24 marchés publics allant au moins jusqu'à 1.28M€

Système utilisé dans 3 primaires de l'élection présidentielle de 2022 :

- Europe-Ecologie-Les-Verts (EELV) fin septembre
- Les Républicains (LR) début décembre
- **Primaire Populaire** (PP) fin janvier

En procès-bâillon contre le CNRS, le CCSD (Hal) et plusieurs Universités à cause de cet article

Travaux précédents et parallèles

Critiques générales sur la Primaire Populaire :

- Inscriptions multiples (aussi pour LR)
- Méthode de décompte (jugement majoritaire)

Une seule analyse publique par *de Barros, Gergouil, Grelard and Thibault* :

- Élections professionnelles à l'Université de Bordeaux
- Problèmes avec ESMTPS
- Inscription peu sécurisée, anonymat faible

Lanceur d'alerte pour le scrutin EELV

Arguments de Neovote et transparence

Annoncé sur le site de Neovote :

- 10k votes sans problème technique
- Homologué par de nombreuses institutions (Sénat, Assemblée Nationale, DGSI...)
- Pas de code externe, Debian modifiée et pile de sécurité (dont protocoles de communication) maison
- Déployées dans des centres SecNumCloud but n'utilise pas de ressources cloud
- Vocabulaire non-standard (urnes aléatoires, modèles géométriques, "pas de mélangeur")
- Pas de transparence "pour des raisons de secret défense", aucune idée des modèles, algorithmes, protocoles ou code utilisés sur les serveurs

Méthodologie

Plusieurs questions méthodologiques et éthiques :

- Prendre le risque d'affecter la légitimité lors d'élections majeures ?
- Comment analyser sans affecter le scrutin ?

Décisions:

- Inscriptions et comportements normaux (pas de tentative de votes multiples)
- Enregistrement complet, téléchargement et analyse du code public
- Comparaison du code avec codes précédents

En parallèle : prévenir les institutions (CNIL, ANSSI) et Neovote

Réglementations en vigueur

Règles de la CNIL (mises à jour en 2019) :

- 1-07 : Assurer l'étanchéité totale entre l'identité de votant et l'expression de son vote pendant toute la durée du traitement.
- 1-11 : S'assurer que le dépouillement de l'urne puisse être vérifié a posteriori.
- 2-06 : Utiliser un système d'information mettant en œuvre les mesures de sécurité physique et logique recommandées par les éditeurs et l'Agence nationale de la sécurité des systèmes d'information.
- 2-07 : Assurer la transparence de l'urne pour tous les électeurs.
- 3-02 : Permettre la transparence de l'urne pour tous les électeurs à partir d'outils tiers.

Règles de l'ANSSI (2.2.5 et 2.2.6) : éviter les protocoles maison, seulement utiliser des bibliothèques connues, testées et maintenues :

Il est impératif de n'employer que des bibliothèques éprouvées bénéficiant d'un suivi de leur sécurité pour tout appel à des mécanismes cryptographiques.

Analyse de code

Un problème : l'obfuscation

- Recharger la page change les noms de fonctions/variables
- Un schéma de renommage subsiste
- La structure est conservée, comme les chaînes de caractères
- La portée empêche parfois l'obfuscation

On compare les structures et chaînes de caractères

AES_encrypt

```
AES_Encrypt_process
(data: Uint8Array):Uint8Array
{if (!is_bytes(data)) throw new
  TypeError("data isn't of expected type");

  let asm = this.asm;
  let heap = this.heap;
  let amode = AES_asm.ENC[this.mode];
  let hpos = AES_asm.HEAP_DATA;
  let pos = this.pos;
  let len = this.len;
  let dpos = 0;
  let dlen = data.length  0;
  let rpos = 0;
  let rlen = (len + dlen) & -16;
  let wlen = 0;

  let result = new Uint8Array(rlen);

      ⋮
```

```
.xTpmDHxL=function
($xTpmppgF)
{if(!xTpmppDL($xTpmppgF)){throw new
TypeError("data isn't of expected type");}

var $xTpmpDmV=this.$xTpmpDmV;
var $xTpmppgm=this.$xTpmppgm;
var $xTpmpDLT=xTpmpDNN.xTpmDHNL[this.$xTpmpDYs];
var $xTpmppgg=xTpmpDNN.xTpmDHxH;
var $xTpmpDYV=this.$xTpmpDYV;
var $xTpmprNV=this.$xTpmprNV;
var $xTpmppHx=0;
var $xTpmppHs=$xTpmppgF.length0;
var $xTpmpDLY=0;
var $xTpmpDLL=($xTpmprNV+$xTpmppHs)&-16;
var $xTpmppHg=0;

var $xTpmpDLD=new Uint8Array($xTpmpDLL);

      ⋮
```

RSAES-PKCS

```
export function getNonZeroRandomValues
(buf: Uint8Array)
{getRandomValues(buf);
  for (let  i = 0;
  i < buf.length;  i++) {
    let byte = buf[i];
    while (!byte) {
      const octet = new Uint8Array(1);
      getRandomValues(octet);
      byte = octet[0];
    }
    buf[i] = byte;}}
```

```
var xTpmpDDx=function
(xTpmpDpH)
{xTpmpDpD(xTpmpDpH);
for(var $xTpmprNs=0;
$xTpmprNs<xTpmpDpH.length;$xTpmprNs++){
var $xTpmpDDW=xTpmpDpH[$xTpmprNs];
while(!$xTpmpDDW){
var $xTpmpDDV=new Uint8Array(1);
xTpmpDpD($xTpmpDDV);
$xTpmpDDW=$xTpmpDDV[0];
}
xTpmpDpH[$xTpmprNs]=$xTpmpDDW;}
```

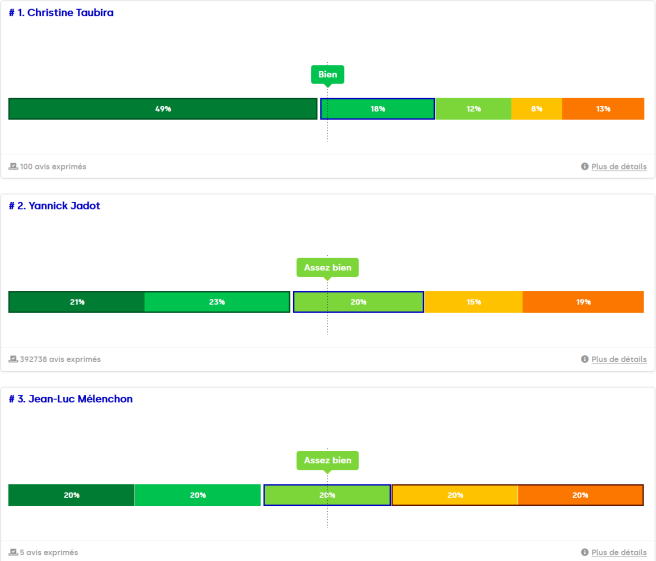
Publication des résultats

Protocole (selon ex-membre de PP) :

- Données brutes envoyées par Neovote à : autorité électorale, expert CNIL, huissier
- Confirmation indépendante que tout est conforme
- Résultats bruts restent privés
- Décompte final calculé et publié (en ligne et conférence de presse)

Vulnérabilités car pas de vérification E2E et pas de transparence

Scrutin de la Primaire Populaire



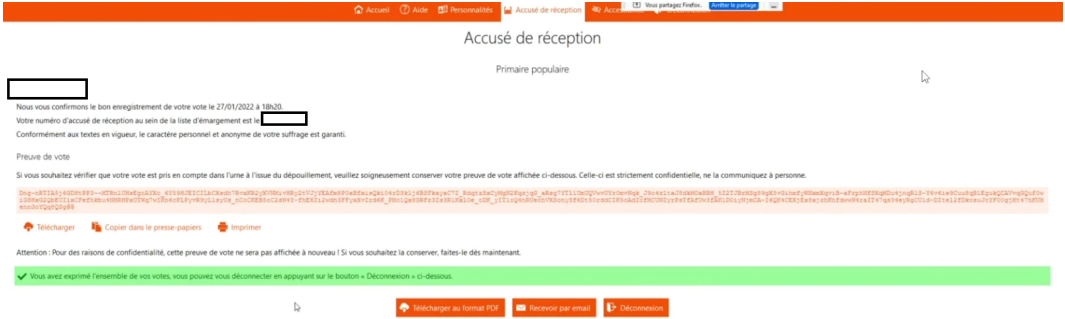
Possibilité de vérifier son vote

Vérification

- “Vous pourrez vérifier que votre vote est bien pris en compte”
- Preuve de vote (longue chaîne alphanumérique dans le navigateur)
- Interdit de partager la preuve de vote
- Facile de rater le téléchargement

Fonctionnel pour EELV, pour PP pas d’information sur comment vérifier

Possibilité de vérifier son vote



Structure de la vérification

Structure :

- Chaque preuve a 5 hachés (chiffrés côté client avec clé publique constante)
- L'urne est une liste de (bulletin, haché)
- Hachés supplémentaires dans un fichier annexe

Protocole de vérification :

- Supprimer les hachés supplémentaires
- Vérifier que les hachés restant sont dans l'urne
- Faire déchiffrer l'urne par le serveur (RSA)
- Compter les votes

Attaques sur l'urne

Vulnérabilité sur l'urne

- Urne non signée
- Possibilité de créer une fausse urne indistinguable de la vraie avec données arbitraires.

Vulnérabilité sur la preuve

- Preuves non signées
- Attaque par faux reçu permettant d'attaquer la légitimité du scrutin

Attaques sur l'anonymat

Spéculatif car pas de code serveur disponible mais la structure des hachés indique que :

- Si aucun haché supplémentaire dans l'urne (tous dans le fichier annexe), possibilité de prouver son vote
- Si seulement certains hachés sont présents, possibilité de prouver comment on n'a pas voté

Si les hachés ne sont pas dans le fichier annexe :

- Sans partage de preuve de vote : possibilité pour les organisateurs de modifier arbitrairement le scrutin par réutilisation de hachés
- Avec partage : séquence de preuves permettant de prouver comment des groupes ont voté

Jurisprudence

Plusieurs décisions de justice

- Cour d'Appel : annulation d'un vote en 2019 car trop faibles garanties d'anonymat
- Cour de Cassation : l'expertise *in abstracto* est suffisante, pas besoin d'expertiser le système à chaque utilisation
- Recommandations en terme de vie privée trop faibles (2-facteurs malgré données publiques)

Conclusion

On observe une multiplication des systèmes de vote électronique (Voxaly, Neovote, Gedivote)

L'appareil juridique ne semble pas adéquat pour imposer un haut niveau de sécurité

Comment passer d'une exigence d'expertise (idéalement indépendante) à l'exigence du respect de standards internationaux et à son application en pratique ?