La Sécurité de vote par internet

Présentée par : hind magat

Encadré par : monsieur sadikki

Plan de travail:

1-Introduction

2- Les deux propriétés principaux de vote électronique

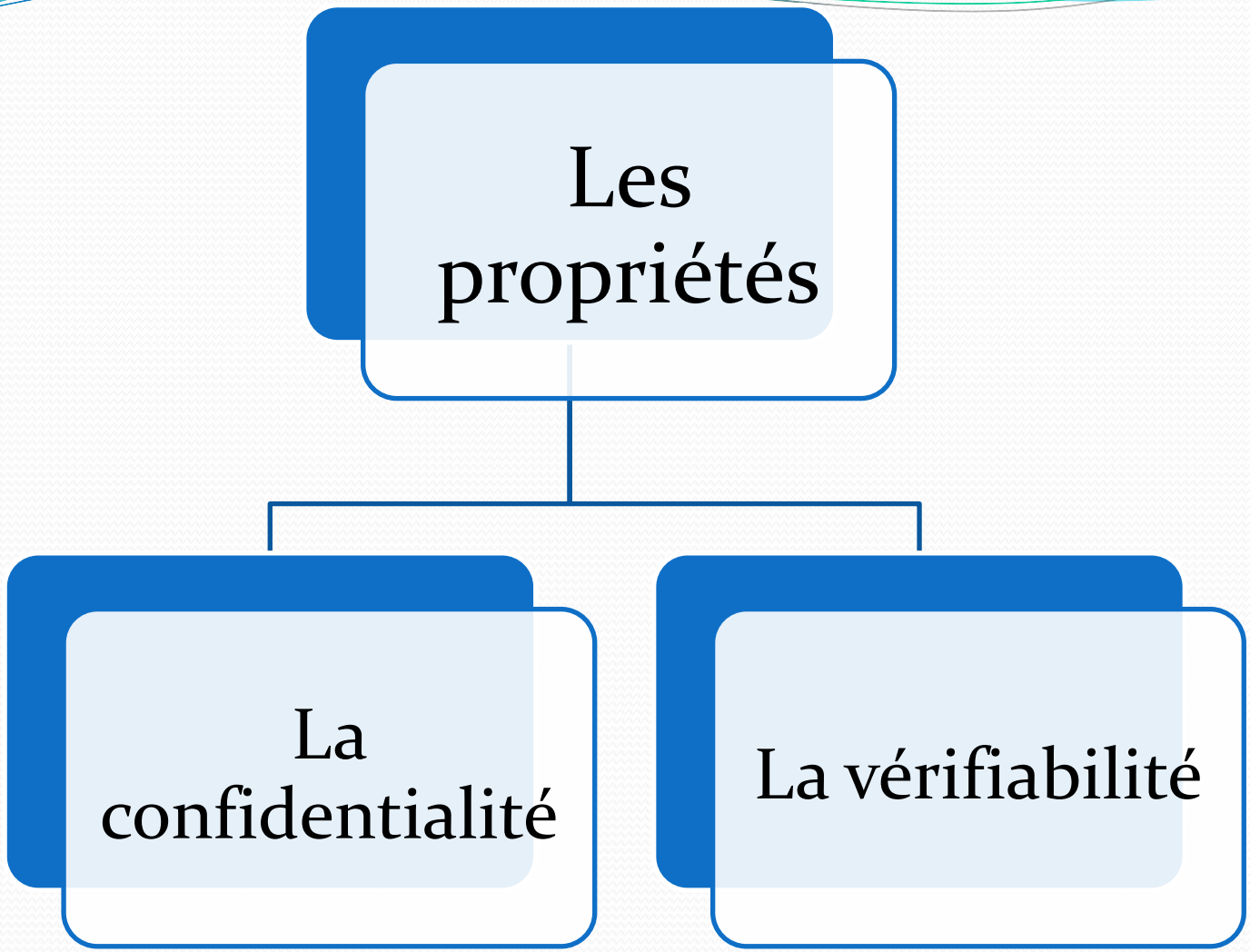
3-Modélisation mathématique :**le principe de el gamal**

4- Etude informatique

5-Exemple d'un protocole de vote: Belenios

introduction

Pour montrer la confidentialité vers le vote électronique ,il faut comprendre le principe de el gamal . Nous allons donc faire une étude théorique et expérimentale de ce principe afin de minimiser l'accusation de ce vote de fraude.


Les propriétés
La confidentialité
La vérifiabilité

Un point de vue sur mon sujet

La confidentialité

Le principe de el gamal: cryptage , décryptage

Le problème de logarithme discrète

Modélisation mathématique

a) **Préparation de clé**:

Définition:

soit G un groupe cyclique et x $\in$G

on dit que x est d'ordre fini si il existe p un entier non nul tell que $\mathbf{x^p = e}$

le plus petit p qui vérifie $\mathbf{x^p = e}$ s'appelle l'ordre de x.

Soit H un groupe cyclique choisi

Soit g∈ H et p son ordre qui est très grand

On choisit $1 \leq x \leq p - 1$

on calcule $y = g^x \ [p]$

Donc **la clé privée : x**

la clé publique : y

b- Cryptage:

Soit la clé public y

Soit v un vote

Soit une clé éphémère $k \in H$ telle que $pgcd(k , p-1)=1$

on calcule $r= g^{k} [p]$

$c= v \cdot y^{k} [p]$

Le vote crypté est:

(r , c)

Décryptage:

Soit v un vote qu'on veut décrypter

Soit (r, c) le vote crypté

Pour décrypter on calcule :

$$Dx(v) = r / c^{x}$$

Vérification :

$$Dx(v) = c/r^{x}$$

$$= v.\, y^{k}[p] \,/ g^{k\,x}[p]$$

$$= v(g^{x})^{k}\,[p] \,/ (g^{k})^{x}\,[p]$$

$$= v$$

Car $y = g^{x}\,[p]$

La preuve de la sécurité de ce vote :

Logarithme discrète:

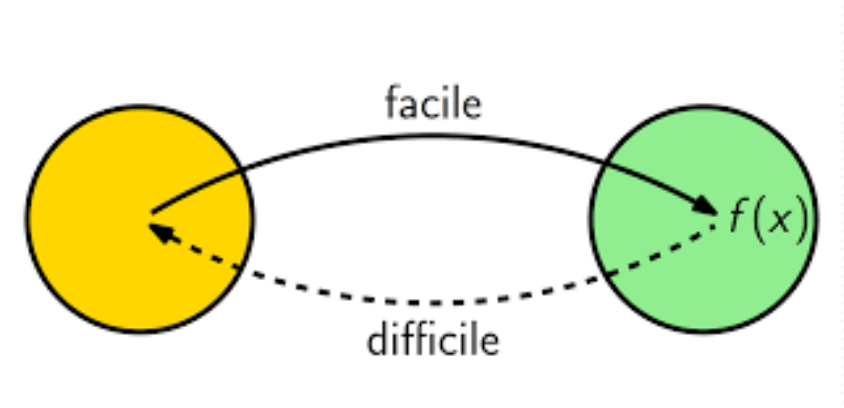
Soit g , p deux paramètres choisis

Il est difficile de trouver x telle que :

$y=g^{x}\ [p]$ car y , p , x sont des nombres très grands

La confidentialité de vote:

soit **x** la clé privée et **$f(x)=g^x[p]$** la clé public



L'étude informatique:

```
1
2
3   # un programme qui traite si un nombre est
    premier
4
5   from random import*
6   def estpremier(n):
7       for k in range (2,n//2+1):
8           if n%k==0:
9               return False
10      return True
11
12
13
14
15
16
```

```
14
15
16
17  #un programme qui renvoie le pgcd de deux
    nombres
18
19  def pgcd(a,b):
20      r=a%b
21      while r!=0:
22          a=b
23          b=r
24          r=a%b
25      return b
26
27
28
```

```
27
28
29
30  # un programme qui teste si un nombre est
    premier
31
32  def chercherpremier(g):
33      x=randint(0,1000)
34      print("x",x)
35      p=2
36      while True:
37          if estpremier(p) and p>g and p>x:
38              return p,x
39          p=p+1
40
```

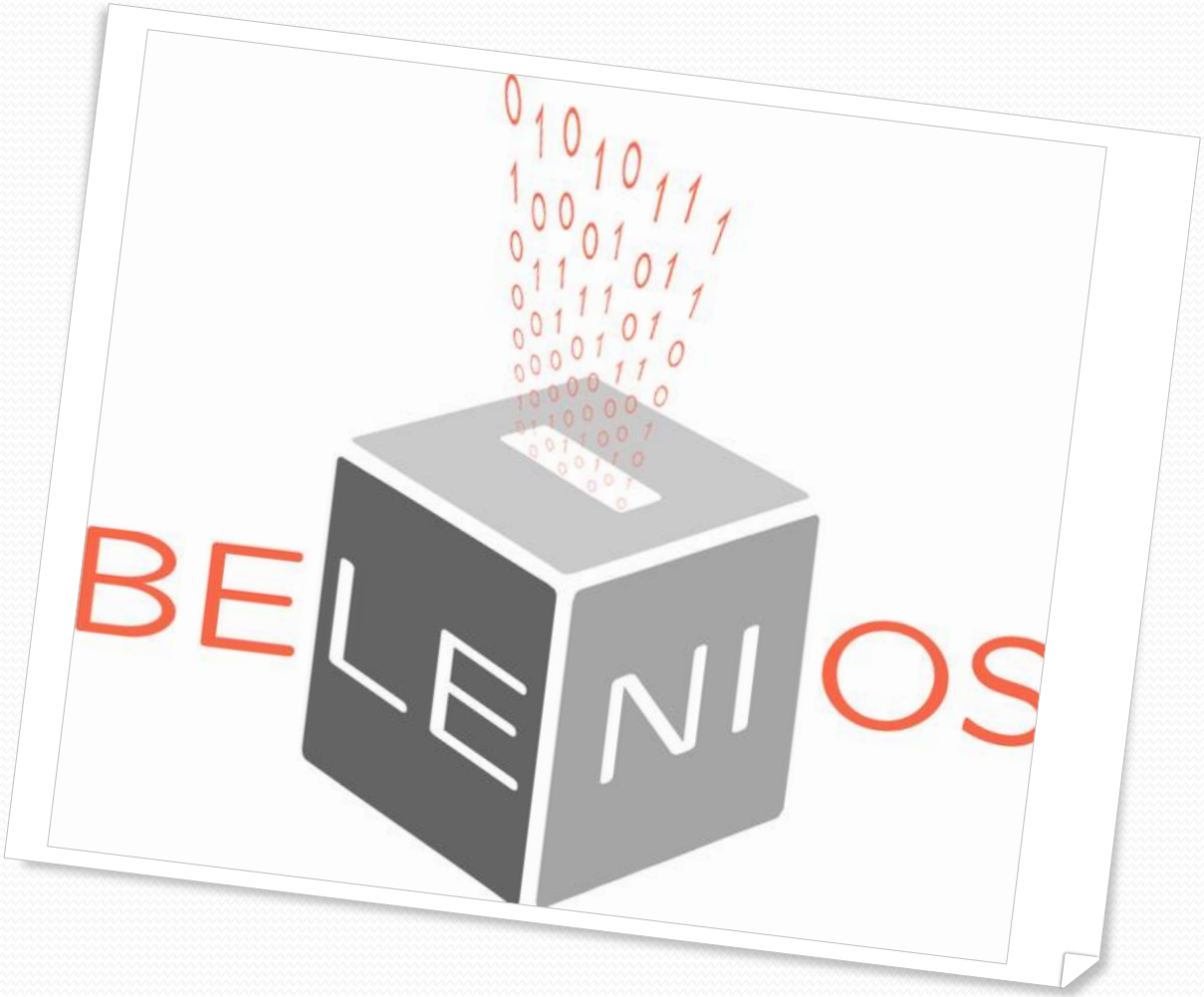
```
42
43
44
45
46
47
48  # un programme  qui renvoie la clé privée
49
50  def clepublic(g,p,x):
51      return (g**x)%p
52
53
54
55
56
57
```

```
57
58
59  # un programme qui renvoie le messsage ou  le
    vote crypté
60
61  def crypter(m,g):
62      p,x=chercherpremier(g)
63      print("p=",p,"x=",x)
64      y=clepublic(g,p,x)
65      while True:
66          k=randint(2,1000)
67          print("k=",k)
68          if pgcd(k,p-1):
69              break
70      r=(g**k)%p
71      c=(m*y**k)%p
72      return (r,c)
73
```

```
>>> crypter(5,3)
x= 280
p= 281 x= 280
k= 566
(167, 5)
```

Un exemple d'un protocole de vote assure :

- La vérifiabilité
- La confidentialité



Le processus sécurisé:

L' opérateur de dépouillement: choisi **(Y, X)** pour chaque votant

L'électeur : crypte son choix **Vi**

L'opérateur de calcul : reçoit chaque nombre crypté **Y(Vi)**

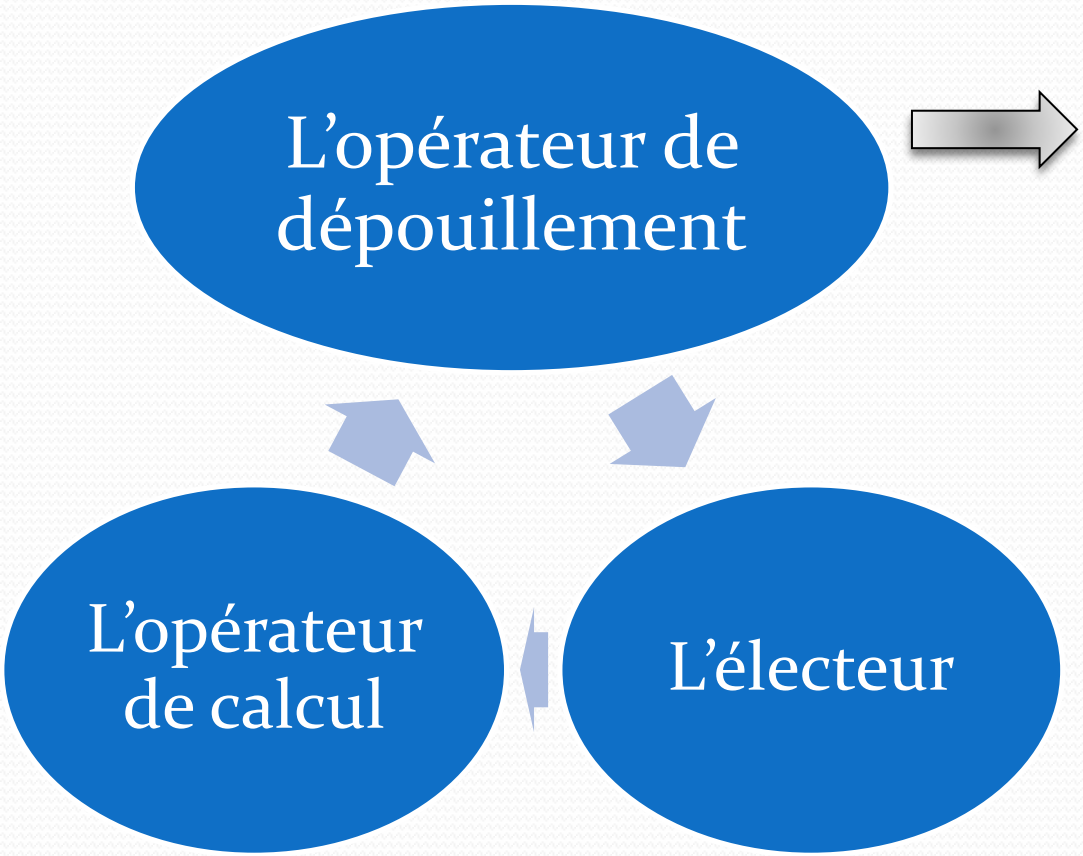
Et calcule **Y(V1+V2…V n)=Y(V1)* Y(V2)…*Y(V n)=P**

L'opérateur de dépouillement: décrypte P par X

X(P)=X(Y(V1) *Y(V2)…. *Y(V n))=X(Y(V1+V2….+V n))

= V1+V2…+V n = nombre de oui

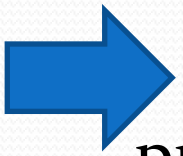
Le cycle:



Diffuse le résultat

Conclusion

Le problème de logarithme discret présent dans el gamal empêche l'accès à la clé privée d'un électeur



La confidentialité est donc prouvée à l'aide de ce problème