



**Universidad de Buenos Aires**  
**Facultad de Ingeniería**  
**Departamento de Informática**



***Organización de Datos (75.06)***

**Voto Electrónico**  
**Vigenere y Kasiski**

Cuatrimestre y año: 2<sup>do</sup> Cuatrimestre 2011

Docente a cargo del TP: Nicolás Pablo Fernández Theillet

Grupo: Lamas

Fecha de Entrega: 2011-11-03

Integrantes:

| <b><i>Padrón</i></b> | <b><i>Nombre</i></b>         | <b><i>Email</i></b>               |
|----------------------|------------------------------|-----------------------------------|
| 91187                | Gonzalez Durand, Juan Manuel | jmanuel.gonzalez.durand@gmail.com |
| 90762                | Ostrowsky, Gabriel           | gaby.ostro@gmail.com              |
| 90728                | Schenkelman, Damián          | damian.schenkelman@gmail.com      |
| 91045                | Torrado, Alejandro           | aletorrado@gmail.com              |
| 90884                | Zamudio, Gonzalo             | ahogadosderazon@gmail.com         |

Índice

Cifrado de Vigenere..... 3

    Encriptación de Reportes ..... 3

    Implementación Vigenere ..... 3

Ataque de Kasiski..... 4

    Ejemplos ..... 4

## Cifrado de Vigenere

En esta sección se detalla la implementación del cifrado de Vigenere utilizada para el trabajo práctico y el uso de la misma para encriptar y desencriptar reportes.

### Encriptación de Reportes

1. Cuando se quiere guardar un reporte en disco, se pide una clave al usuario. La misma puede tener la cantidad de caracteres que el usuario desee, siempre que estos sean imprimibles. Para este ejemplo supongamos que la clave es **MICLAVE**.
2. Una vez que el usuario ingresa su clave, se encripta el reporte. Para mantener la estructura del reporte y tener la posibilidad de hacer el ataque de Kasiski, solo se encriptan las letras. Por ejemplo, si tenemos el siguiente reporte, solo los **caracteres resaltados** se encriptaran:

| Reporte sin encriptar                    | Reporte encriptado                       |
|--|--|
| DISTRITO: YUMA                           | PQUERDXA: GWXA                           |
| GANADOR DE ELECCION: GOBERNADOR PROV     | BEZIFZR YI QTGNCDZS: OQMEMRMLQC PMSH     |
| NRO 107, 10-12-2011                      | VTZ 107, 10-12-2011                      |
| LISTA: LISTA 3 VOTOS: 18                 | LDWFI: NTSOE 3 HWVZS: 18                 |
| GANADOR DE ELECCION: INTENDENTE DE YUMA, | BEZIFZR YI QTGNCDZS: QPEEIHQVVP DZ CGUC, |
| 15-11-2011                               | 15-11-2011                               |
| LISTA: LISTA 3 VOTOS: 13                 | WINXM: TKDTV 3 ZABQD: 13                 |
| GANADOR DE ELECCION: PRESIDENTE NRO 6    | GVRMLQC DZ IXMENIJR: BZGDIYIZBG YRJ 6    |
| FECHA: 12-11-2011                        | JQKJL: 12-11-2011                        |
| LISTA: LISTA 2 VOTOS: 20                 | LDWFI: NTSOE 2 HWVZS: 20                 |

Los reportes se guardan en el directorio **Files/Reports**.

### Implementación Vigenere

Para el cifrado, utilizamos el siguiente algoritmo:

1. Apareamos cada letra del mensaje ( $m_i$ ) con una letra de la clave ( $k_i$ ).
2. Obtenemos el símbolo relacionado del criptograma  $c_i$  de la forma:  
$$\text{crypted} = ((\text{message}[i] + \text{key}[j]) \% 26) + 'A';$$

En el caso de la clave **MICLAVE**, y la palabra **DISTRITO** las operaciones y el resultado serían:

$$\begin{array}{r} \text{DISTRITO} \\ + \quad \text{MICLAVEM} \\ \% \quad 26 \ 26 \ 26 \ 26 \ 26 \ 26 \ 26 \ 26 \\ + \quad \text{AAAAAAAA} \\ \hline \text{PQUERDXA} \end{array}$$

## Ataque de Kasiski

En esta sección continuación se detalla la implementación del ataque de Kasiski. El mismo es llevado a cabo en distintas etapas secuenciales, siendo el resultado de cada una de ellas necesario para la realización de la siguiente.

**Nota:** Esto solo considera a los caracteres que son letras mayúsculas, ya que son los únicos que fueron encriptados.

1. El ataque comienza con un criptograma a atacar y una longitud de n-gramas (si la longitud es 3 son trigramas, 4 tetragramas, etc.) para analizar a dicho criptograma.
2. Busca la posición de todos los n-gramas repetidos en el criptograma. Por ejemplo, si  $n = 3$ , se buscan conjuntos de 3 letras repetidos en diferentes posiciones del criptograma.
3. Calcula la distancia entre repeticiones de cada n-grama. Por ejemplo si ABC aparece en las posiciones 1, 7 y 33, las distancias serán 6 y 26.
4. Obtiene todos los divisores de cada una de las distancias y cuenta las apariciones de cada divisor.
5. Ordena los divisores según su cantidad de apariciones. Considera a los primeros tres longitudes candidatas de la clave.

Los pasos detallados a continuación son específicos para un reporte de Lista, el cual tiene su mayor longitud en una elección presidencial.

Estos pasos se repiten para cada una de las tres posibles longitudes candidatas o hasta que intentando con alguna de todas las posibles claves de una longitud se encuentren las palabras "**DISTRITO**" y "**VOTOS**" en el mensaje descifrado (con claves largas, es posibles que con más de una clave candidata se obtengan estas palabras, ver [Ejemplos](#)).

Sea  $L$  la longitud propuesta para una clave:

1. Crea  $L$  cadenas de caracteres vacías y las enumera de 0 a  $L-1$ .
2. Crea  $2^L$  cadenas vacías, las que considerara como claves posibles.
3. Recorre el criptograma. Sea  $i$  la posición actual en el criptograma, agrega a la cadena  $(i \% L)$  la letra actual.
4. Para cada cadena  $(0 \dots L-1)$ , sea  $s$  la posición de la cadena actual:
  - a. Calcula la cantidad de apariciones de cada carácter. Sea  $c$  el carácter más frecuente.
  - b. Considera que  $c$  puede ser 'O' o 'T' (las palabras que se repiten en el reporte son **DISTRITO** y **VOTOS**, en las cuales esas letras son las más frecuentes).
  - c. A partir de  $c$  y 'O' obtiene una posible letra de la clave (la de la posición  $s$ ). Lo mismo para  $c$  y 'L'.
  - d. Agrega ambas posibilidades a las claves posibles pertinentes.

**Aclaración:** Al finalizar este paso (4) se cuenta con  $2^L$  claves posibles.

5. Para cada clave, se descifra el mensaje y se imprime por pantalla si encuentra las palabras "**DISTRITO**" y "**VOTOS**".

## Ejemplos

El ataque explicado arriba funciona bien en casos que la clave está formada por letras mayúsculas. No contempla otros casos (aunque podría), pero debería considerar una cantidad mayor a  $2^L$  de claves posibles (sería mucho mayor ya que cambiaría la base a por ejemplo 6).

En la entrega se pueden ver el caso exitoso en los siguientes informes:

- kasiski
- kasiski2

**kasiski:** La clave es **MICLAVE**. Como se puede ver, la primera longitud propuesta es 7 caracteres y una de las 128 claves intentadas es la correcta, por lo que el criptograma se puede descifrar correctamente.

```

damian@ubuntudev: ~/datos11/ElectronicElections
File Edit View Search Terminal Help
12. Atacar reporte (Kasiski)
13. Romper RSA
14. Volver
12
Nombre archivo? kasiski

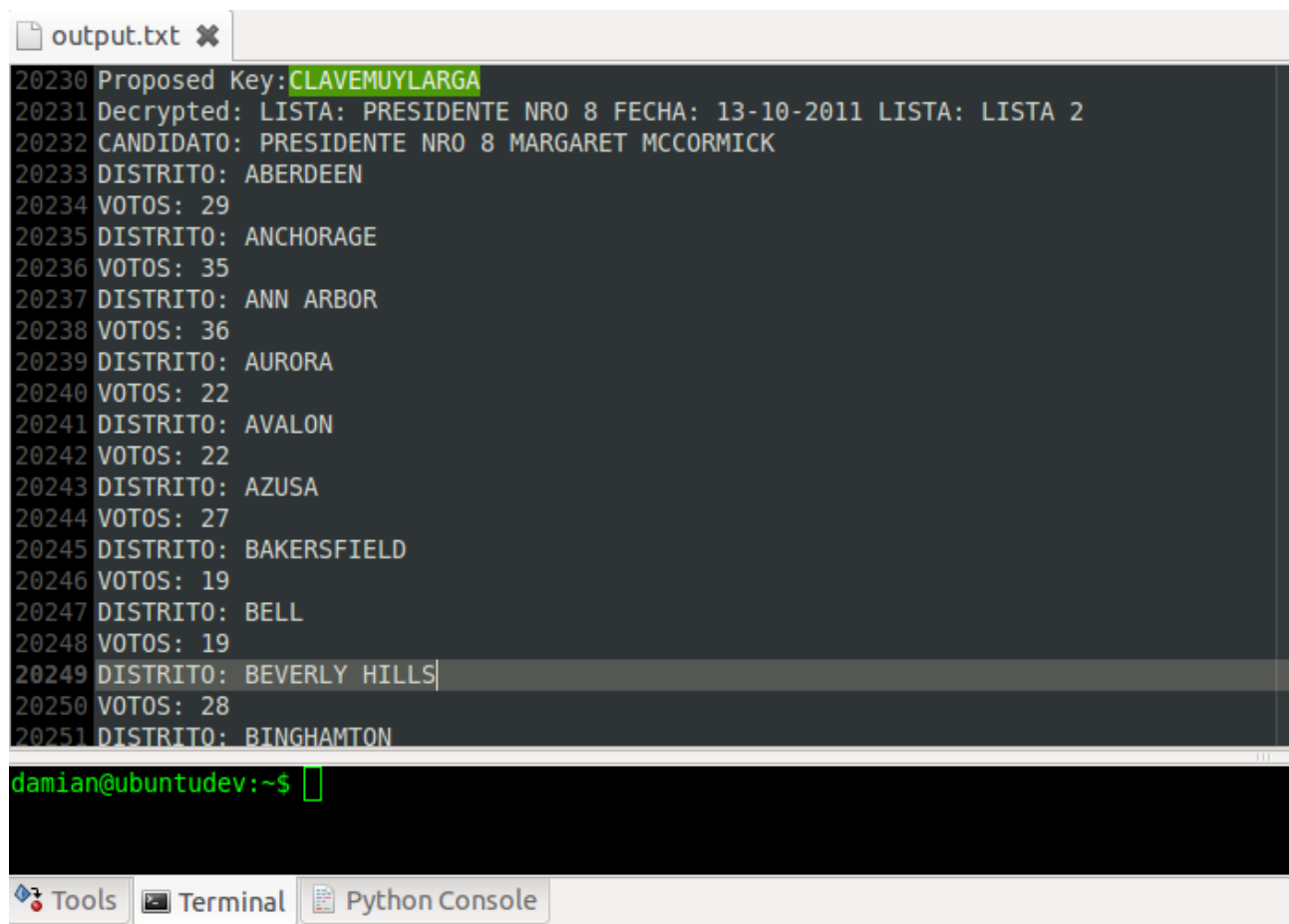
Using Key Length: 7
Proposed Key:MICLAVE
Decrypted: LISTA: PRESIDENTE NRO 6 FECHA: 12-11-2011 LISTA: LISTA 2
CANDIDATO: PRESIDENTE NRO 6 MACON RASMUSSEN
DISTRITO: ABILENE
VOTOS: 10
DISTRITO: ALEXANDRIA
VOTOS: 17
DISTRITO: ANDERSON
VOTOS: 9
DISTRITO: ARTESIA
VOTOS: 13
DISTRITO: ASHEVILLE
VOTOS: 24
DISTRITO: ATLANTA
VOTOS: 16
DISTRITO: ATTLEBORO
VOTOS: 16

```

**kasiski2:** La clave es CLAVEMUYLARGA. Este caso es más interesante porque como la clave es más larga que las palabras "**DISTRITO**" y "**VOTOS**", hay más de una clave que puede hacer que estas palabras aparezcan en el reporte. Como varias posibilidades serán mostradas por pantalla, una forma simple de verificar que el ataque fue exitoso es redirigiendo la entrada de la consola antes de correr el programa:

| Bash                               |
|------------------------------------|
| ./ElectronicElections   tee output |

Abriendo el archivo output podemos hacer CTRL+F ("CLAVEMUYLARGA") y se verá lo siguiente en la línea 20230:



```
output.txt ✕
20230 Proposed Key: CLAVEMUYLARGA
20231 Decrypted: LISTA: PRESIDENTE NRO 8 FECHA: 13-10-2011 LISTA: LISTA 2
20232 CANDIDATO: PRESIDENTE NRO 8 MARGARET MCCORMICK
20233 DISTRITO: ABERDEEN
20234 VOTOS: 29
20235 DISTRITO: ANCHORAGE
20236 VOTOS: 35
20237 DISTRITO: ANN ARBOR
20238 VOTOS: 36
20239 DISTRITO: AURORA
20240 VOTOS: 22
20241 DISTRITO: AVALON
20242 VOTOS: 22
20243 DISTRITO: AZUSA
20244 VOTOS: 27
20245 DISTRITO: BAKERSFIELD
20246 VOTOS: 19
20247 DISTRITO: BELL
20248 VOTOS: 19
20249 DISTRITO: BEVERLY HILLS|
20250 VOTOS: 28
20251 DISTRITO: BINGHAMTON

damian@ubuntudev:~$
```

Tools Terminal Python Console