Criptografía y Seguridad Computacional

1. Introducción y Conceptos Básicos

Fernando Krell Loy

2 de Marzo 2016

Datos del Curso

- Profesor: Fernando Krell
- e-mail: fkrell@gmail.com
- Horario clases: Mie. 15:30 - 18:20
- Horario consultas: Mie 18:20 - 19:20
- Ayudantes: TBA
- Evaluación:
  - 6 tareas - Soluciones en PDF en LaTeX. 60%
  - 2 pruebas.20 %
  - 1 examen. 20 %

Criptografía vs Seguridad Computacional

Seguridad Computacional:

- Seguridad de redes
- Análisis de código
- Seguridad en sistemas operativos
- Sistemas de detección de intrusos
- Cloud computing security
- ...
- y... criptografía

$$\text{CRYPTO} \subset \text{SECURITY}$$

Criptografía vs Seguridad Computacional

Seguridad Computacional:

- Seguridad de redes
- Análisis de código
- Seguridad en sistemas operativos
- Sistemas de detección de intrusos
- Cloud computing security
- ...
- y... criptografía

$$\text{CRYPTO} \subset \text{SECURITY}$$

¿Qué es la criptografía?

Visión clásica de la Criptografía

Privacidad del mensage: Sólo Alice y Bob pueden entender el mensaje.
Autentificación: Bob sabe que el mensage no fue modificado.

Hoy en día, cripto es mucho más

- Cifrados basados en identidad
- Cifrados basados atributos
- Cifrado funcional (puedo descifrar $f(m)$ y nada mas).
- Búsqueda privada de datos
- Computación segura:
  - Encriptación homomorphica.
  - Obfuscación de programas.
- y más

Revisión de Probabilidad Discreta

- Sea S un conjunto *discreto*, y $p(\cdot) : S \to [0, 1]$ una función.
- p define un espacio de probabilidades si $\sum_{a \in S} p(a) = 1$.
- Un evento E es un subconjunto de S. $\Pr[E] \stackrel{def}{=} \sum_{e \in E} p(e)$

Definition

Eventos E_1 y E_2 son *independientes* si $\Pr[E_1 \vee E_2] = \Pr[E_1] + \Pr[E_2]$

Propiedades

- $\Pr[E_1 \vee E_2] = \Pr[E_1] + \Pr[E_2] - \Pr[E_1 \wedge E_2]$
- $\Pr[E_1 \vee E_2] \leq \Pr[E_1] + \Pr[E_2]$ (cota de unión, $=$ sólo si son ind.)

Propiedades

- $\Pr[E_1 \vee E_2] = \Pr[E_1] + \Pr[E_2] - \Pr[E_1 \wedge E_2]$
- $\Pr[E_1 \vee E_2] \leq \Pr[E_1] + \Pr[E_2]$ (cota de unión, $=$ sólo si son ind.)
- $\Pr[\bigvee_{i=1}^{k} E_i] \leq \sum_{i=1}^{k} \Pr[E_i]$ (cota de unión generalizada)

Propiedades

- $\Pr[E_1 \vee E_2] = \Pr[E_1] + \Pr[E_2] - \Pr[E_1 \wedge E_2]$
- $\Pr[E_1 \vee E_2] \leq \Pr[E_1] + \Pr[E_2]$ (cota de unión, $=$ sólo si son ind.)
- $\Pr[\bigvee_{i=1}^{k} E_i] \leq \sum_{i=1}^{k} \Pr[E_i]$ (cota de unión generalizada)
- Sea $\{E_i\}_{i=1}^{k}$ tales que $E_i \cap E_j = \emptyset \forall i \neq j$ y $\sum_{i=1}^{k} \Pr[E_i] = 1$ (una partición del espacio de probabilidades). Entonces $\forall$ evento F:

$$\Pr[F] = \sum_{i=1}^{k} \Pr[F \wedge E_i]$$

Propiedades

- $\Pr[E_1 \vee E_2] = \Pr[E_1] + \Pr[E_2] - \Pr[E_1 \wedge E_2]$
- $\Pr[E_1 \vee E_2] \leq \Pr[E_1] + \Pr[E_2]$ (cota de unión, $=$ sólo si son ind.)
- $\Pr[\bigvee_{i=1}^{k} E_i] \leq \sum_{i=1}^{k} \Pr[E_i]$ (cota de unión generalizada)
- Sea $\{E_i\}_{i=1}^{k}$ tales que $E_i \cap E_j = \emptyset \forall i \neq j$ y $\sum_{i=1}^{k} \Pr[E_i] = 1$ (una partición del espacio de probabilidades). Entonces $\forall$ evento F:

$$\Pr[F] = \sum_{i=1}^{k} \Pr[F \wedge E_i]$$

- $\Pr[E_1|E_2] \stackrel{\text{def}}{=} \frac{\Pr[E_1 \wedge E_2]}{\Pr[E_2]}$ (probabilidad condicional)

$$\Rightarrow$$

Si E_1 y E_2 son independientes, $\Pr[E_1 \wedge E_2] = \Pr[E_1] \cdot \Pr[E_2]$

Propiedades

- $\Pr[E_1 \vee E_2] = \Pr[E_1] + \Pr[E_2] - \Pr[E_1 \wedge E_2]$
- $\Pr[E_1 \vee E_2] \leq \Pr[E_1] + \Pr[E_2]$ (cota de unión, $=$ sólo si son ind.)
- $\Pr[\bigvee_{i=1}^{k} E_i] \leq \sum_{i=1}^{k} \Pr[E_i]$ (cota de unión generalizada)
- Sea $\{E_i\}_{i=1}^{k}$ tales que $E_i \cap E_j = \emptyset \forall i \neq j$ y $\sum_{i=1}^{k} \Pr[E_i] = 1$ (una partición del espacio de probabilidades). Entonces $\forall$ evento F:

$$\Pr[F] = \sum_{i=1}^{k} \Pr[F \wedge E_i]$$

- $\Pr[E_1|E_2] \stackrel{\text{def}}{=} \frac{\Pr[E_1 \wedge E_2]}{\Pr[E_2]}$ (probabilidad condicional)

$$\Rightarrow$$

Si E_1 y E_2 son independientes, $\Pr[E_1 \wedge E_2] = \Pr[E_1] \cdot \Pr[E_2]$

Theorem (Bayes)

Si $\Pr[E_2] > 0$ *entonces* $\Pr[E_1|E_2] = \frac{\Pr[E_1] \cdot \Pr[E_2|E_1]}{\Pr[E_2]}$

Proof.

$$\begin{aligned}\Pr[E_1|E_2] &= \frac{\Pr[E_1 \wedge E_2]}{\Pr[E_2]} \\ &= \frac{\Pr[E_1] \cdot \Pr[E_2|E_1]}{\Pr[E_2]}\end{aligned}$$

□

Hemos usado probabilidad condicional 2 veces.

Notación asintótica

- $f(n) = O(g(n))$ si $\exists c, n'$ tales que $f(n) \leq c \cdot g(n)\ \forall n \geq n'$.
- $f(n) = \Omega(g(n))$ si $\exists c, n'$ tales que $f(n) \geq c \cdot g(n)\ \forall n \geq n'$.
- $f(n) = \Theta(g(n))$ si $f(n) = O(g(n))$ y $f(n) = \Omega(g(n))$
- $f(n) = o(g(n))$ si $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$
- $f(n) = \omega(g(n))$ si $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$
- $f(n) = \tilde{O}(g(n))$ si $f(n) = O(g(n) \cdot \log^c g(n))$ para alguna constante c.

Funciones negligibles

Definition

Una function $\mu : \mathbb{N} \to \mathbb{R}$ es negligible si $\forall$ polinomio $p(\cdot)$

$$p(\cdot)\mu(n) < 1/p(n)$$

para todo n suficientemente grande. (En otras palabras $\mu(n) = o(1/p(n))$ para todo polinomio $p(\cdot)$)

Uso: Construiremos esuquemas criptográficos que aseguran que la probabilidad de que el adversario "lo quiebre" es negligible.
Propiedades:

- negligible + negligible = negligible.
  Podremos componer algunos esquemas
- **negligible x polinomio = negligible**
  Si repetimos el adversario $p(n)$ veces, entonces su ventaja es a lo más $p(n) \cdot \mu(n) = \mathsf{negl}(n)$

Funciones negligibles

Definition

Una function $\mu : \mathbb{N} \to \mathbb{R}$ es negligible si $\forall$ polinomio $p(\cdot)$

$$p(\cdot)\mu(n) < 1/p(n)$$

para todo n suficientemente grande. (En otras palabras $\mu(n) = o(1/p(n))$ para todo polinomio $p(\cdot)$)

Uso: Construiremos esuquemas criptográficos que aseguran que la probabilidad de que el adversario "lo quiebre" es negligible.
Propiedades:

- negligible + negligible = negligible.
  Podremos componer algunos esquemas
- **negligible x polinomio = negligible**
  Si repetimos el adversario $p(n)$ veces, entonces su ventaja es a lo más $p(n) \cdot \mu(n) = \mathsf{negl}(n)$

Cifradores antiguos

- Cifrador del Cesar: Rotar alfabeto en 3 posiciones:

a	b	c	d	...	w	x	y	z
D	E	F	G	...	Z	A	B	C

Ejemplo:

Enc(`criptografia`) = `GVMTXSKVEJME`

- Cifrador de desplazamiento: Usar una *llave* en vez de cosntante.

$$\text{Enc}(k, X) = (X + k)$$

Ejercicio: Cual es la llave que descifra YNYYNIRRFGERPR

Cifradores antiguos

- Cifrador del Cesar: Rotar alfabeto en 3 posiciones:

a	b	c	d	...	w	x	y	z
D	E	F	G	...	Z	A	B	C

Ejemplo:

Enc(`criptografia`) = `GVMTXSKVEJME`

- Cifrador de desplazamiento: Usar una *llave* en vez de cosntante.

$$\text{Enc}(k, X) = (X + k)$$

Ejercicio: Cual es la llave que descifra YNYYNIRRFGERPR

- Cifrado mono-alfabético: Cada letra es reemplazada por otra.

a	b	c	d	...	w	x	y	z
T	X	A	F	...	G	Y	S	R

¿Cuál es la llave en este esquema?

Cifradores antiguos

- Cifrador del Cesar: Rotar alfabeto en 3 posiciones:

a	b	c	d	...	w	x	y	z
D	E	F	G	...	Z	A	B	C

Ejemplo:
Enc(`criptografia`) = `GVMTXSKVEJME`

- Cifrador de desplazamiento: Usar una *llave* en vez de cosntante.

$$\text{Enc}(k, X) = (X + k)$$

Ejercicio: Cual es la llave que descifra YNYYNIRRFGERPR

- Cifrado mono-alfabético: Cada letra es reemplazada por otra.

a	b	c	d	...	w	x	y	z
T	X	A	F	...	G	Y	S	R

¿Cuál es la llave en este esquema?

- Cifrador de Vignère: Usar diferentes desplazamientos, luego repetir.
  Ejemplo: Llave es 3 6 7 8. Enc(`seguridad`) =`VKNCUÑKIL`
  Tarea: describir como quebrar este cifrador.
  Caso1:El largo de la llave es conocido.
  Caso2:El largo de la llave es desconocido.

Cifradores antiguos

- Cifrador del Cesar: Rotar alfabeto en 3 posiciones:

a	b	c	d	...	w	x	y	z
D	E	F	G	...	Z	A	B	C

Ejemplo:
Enc(`criptografia`) = `GVMTXSKVEJME`

- Cifrador de desplazamiento: Usar una *llave* en vez de cosntante.

$$\text{Enc}(k, X) = (X + k)$$

Ejercicio: Cual es la llave que descifra YNYYNIRRFGERPR

- Cifrado mono-alfabético: Cada letra es reemplazada por otra.

a	b	c	d	...	w	x	y	z
T	X	A	F	...	G	Y	S	R

¿Cuál es la llave en este esquema?

- Cifrador de Vignère: Usar diferentes desplazamientos, luego repetir.
Ejemplo: Llave es 3 6 7 8. Enc(`seguridad`) =`VKNCUÑKIL`
Tarea: describir como quebrar este cifrador.
Caso1:El largo de la llave es conocido.
Caso2:El largo de la llave es desconocido.

Ahora criptografía *en serio*!

1. Principio de Kerckhoffs: Algoritmo *público*, llave secreta.
2. Seguridad demostrable: Teoremas y demostraciones.
3. $P = NP \Rightarrow$ no hay encriptación*. Adversario podría "*adivinar*" la llave y chequear que es la correcta.
4. Supuesto básico: $P \neq NP$. Es una condición necesaria, es suficiente? Problema abierto.
   - Necesitamos problema NP-completo difícil en el caso promedio. NP-completitud es sólo análisis de peor caso.
5. ¿Cuál condición es suficiente?
   - Depende para qué.
   - Criptografía simétrica: Funciones unidireccionales.
   - Criptografía asimétrica: Permutaciones con "puerta trasera".
   - Hay más ....

Temario

1. Seguridad Perfecta.
2. Criptografía simétrica.
   - 2.1 Definiciones.
   - 2.2 Generadores pseudo aleatoreos y "stream ciphers".
   - 2.3 Cifradores de Bloque. construcción prácticas (AES) y teórica.
   - 2.4 Modos de operacion
   - 2.5 Autenticación. MACs, HMAC, etc.
3. Criptografía asimétrica.
   - 3.1 Teoría de números.
   - 3.2 Cifradores asimétricos: RSA, ElGamal, Rabin, Paillier.
   - 3.3 Firmas digitales.
4. Seguridad de redes
   - 4.1 Infraestructura de clave pública.
   - 4.2 TLS, IPSEC, DNSSEC, ...
5. Tópicos avanzados
   - 5.1 Computación segura multipartita.
   - 5.2 Protocolos de nula divulgación.
   - 5.3 Cifradores homomórficos.
   - 5.4 Cifradores funcionales: IBE, ABE, etc.
   - 5.5 Obfuscadores de indistinguibilidad.