ECS 235B, Lecture 18

February 20, 2019

Break-the-Glass Policies

- Motivation: when security requirements conflict, some access controls may need to be overwritten in an unpredictable manner
  - Example: a doctor may need access to a medical record to treat someone, yet that person is unable to give consent (without which access would be denied)
- User overrides the denial
  - Controls notify some people about the override
  - Controls log override for later audit

Example: Rumpole

- Implements a break-the-glass policy
- *Evidential rules*: how to assemble evidence to create context for request
- *Break-glass rules*: define permissions
    - Includes constraints such as obligations to justify need for actions
- *Grant policies*: how rules are combined to determine whether to grant override

Example: Rumpole Enforcement Model

- *Request*: subject, desired action, resource, obligations acceptable to subject
- Decision point:
  - Grants request
  - Denies request
  - Returns request with set of obligations subject must accept; subject then can send a new request with that set of obligations, if they are acceptable

Key Points

- Hybrid policies deal with both confidentiality and integrity
  - Different combinations of these
- ORCON model neither MAC nor DAC
  - Actually, a combination
- RBAC model controls access based on functionality
- Break-the-glass model handles exceptional circumstances that the access control model does not account for

Information Flow

- Basics and background
  - Entropy
- Non-lattice flow policies
- Compiler-based mechanisms
- Execution-based mechanisms
- Examples
  - Privacy and cell phones
  - Firewalls

Basics

- Bell-LaPadula Model embodies information flow policy
  - Given compartments *A*, *B*, info can flow from *A* to *B* iff *B dom A*
- So does Biba Model
  - Given compartments *A*, *B*, info can flow from *A* to *B* iff *A dom B*
- Variables *x*, *y* assigned compartments $\underline{x}$, $\underline{y}$ as well as values
  - Confidentiality (Bel-LaPadula): if $\underline{x}$ = *A*, $\underline{y}$ = B, and *B dom A*, then *y* := *x* allowed but not *x* := *y*
  - Integrity (Biba): if $\underline{x}$ = *A*, $\underline{y}$ = B, and *A dom B*, then *x* := *y* allowed but not *y* := *x*
- From here on, the focus is on confidentiality (Bell-LaPadula)
  - Discuss integrity later

All About Entropy

- Random variables
- Joint probability
- Conditional probability
- Entropy (or uncertainty in bits)
- Joint entropy
- Conditional entropy
- Applying it to secrecy of ciphers

Random Variable

- Variable that represents outcome of an event
  - *X* represents value from roll of a fair die; probability for rolling *n*: $p(=n) = 1/6$
  - If die is loaded so 2 appears twice as often as other numbers, $p(X=2) = 2/7$ and, for $n \neq 2$, $p(X=n) = 1/7$
- Note: $p(X)$ means specific value for *X* doesn't matter
  - Example: all values of *X* are equiprobable

Joint Probability

- Joint probability of X and Y, $p(X, Y)$, is probability that X and Y simultaneously assume particular values
  - If X, Y independent, $p(X, Y) = p(X)p(Y)$
- Roll die, toss coin
  - $p(X=3, Y=\text{heads}) = p(X=3)p(Y=\text{heads}) = 1/6 \times 1/2 = 1/12$

Two Dependent Events

- X = roll of red die, Y = sum of red, blue die rolls

$p(Y=2) = 1/36$ $p(Y=3) = 2/36$ $p(Y=4) = 3/36$ $p(Y=5) = 4/36$

$p(Y=6) = 5/36$ $p(Y=7) = 6/36$ $p(Y=8) = 5/36$ $p(Y=9) = 4/36$

$p(Y=10) = 3/36$ $p(Y=11) = 2/36$ $p(Y=12) = 1/36$

- Formula:

$p(X=1, Y=11) = p(X=1)p(Y=11) = (1/6)(2/36) = 1/108$

Conditional Probability

- Conditional probability of *X* given *Y*, *p*(*X* | *Y*), is probability that *X* takes on a particular value given *Y* has a particular value
- Continuing example …
  - *p*(*Y*=7 | *X*=1) = 1/6
  - p(*Y*=7 | *X*=3) = 1/6

Relationship

- $p(X, Y) = p(X \mid Y)\, p(Y) = p(X)\, p(Y \mid X)$
- Example:

  $p(X=3,Y=8) = p(X=3|Y=8)\, p(Y=8) = (1/5)(5/36) = 1/36$
- Note: if *X*, *Y* independent:

  $p(X|Y) = p(X)$

Entropy

- Uncertainty of a value, as measured in bits
- Example: *X* value of fair coin toss; *X* could be heads or tails, so 1 bit of uncertainty
  - Therefore entropy of *X* is $H(X) = 1$
- Formal definition: random variable *X*, values $x_1, \ldots, x_n$; so $\sum_i p(X = x_i) = 1$; then entropy is:

$$H(X) = -\sum_i p(X=x_i) \lg p(X=x_i)$$

Heads or Tails?

- $H(X) = -\, p(X\text{=heads}) \lg p(X\text{=heads}) - \text{p}(X\text{=tails}) \lg p(X\text{=tails})$

  $= -\,(1/2) \lg (1/2) - (1/2) \lg (1/2)$

  $= -\,(1/2)\,(-1) - (1/2)\,(-1) = 1$
- Confirms previous intuitive result

n-Sided Fair Die

$H(X) = -\Sigma_i\, p(X = x_i) \lg p(X = x_i)$

As $p(X = x_i) = 1/n$, this becomes

$H(X) = -\Sigma_i\, (1/n) \lg (1/ n) = -n(1/n)\, (-\lg n)$

so

$H(X) = \lg n$

which is the number of bits in n, as expected

Ann, Pam, and Paul

Ann, Pam twice as likely to win as Paul

W represents the winner. What is its entropy?

- w_1 = Ann, w_2 = Pam, w_3 = Paul
- $p(W=w_1) = p(W=w_2) = 2/5$, $p(W=w_3) = 1/5$

- So $H(W) = -\Sigma_i\, p(W=w_i) \lg p(W=w_i)$

  $= -(2/5) \lg (2/5) - (2/5) \lg (2/5) - (1/5) \lg (1/5)$

  $= -(4/5) + \lg 5 \approx -1.52$
- If all equally likely to win, $H(W) = \lg 3 \approx 1.58$

Joint Entropy

- X takes values from $\{ x_1, \ldots, x_n \}$, and $\Sigma_i\, p(X=x_i) = 1$
- Y takes values from $\{ y_1, \ldots, y_m \}$, and $\Sigma_i\, p(Y=y_i) = 1$
- Joint entropy of X, Y is:

$$H(X, Y) = -\Sigma_j\, \Sigma_i\, p(X=x_i, Y=y_j) \lg p(X=x_i, Y=y_j)$$

Example

X: roll of fair die, Y: flip of coin

As X, Y are independent:

$p(X{=}1, Y{=}\text{heads}) = p(X{=}1)\ p(Y{=}\text{heads}) = 1/12$

and

$H(X, Y) = -\sum_j \sum_i p(X{=}x_i, Y{=}y_j) \lg p(X{=}x_i, Y{=}y_j)$

$= -2\ [\ 6\ [\ (1/12) \lg (1/12)\ ]\ ] = \lg 12$

Conditional Entropy

- X takes values from $\{ x_1, \ldots, x_n \}$ and $\Sigma_i\, p(X{=}x_i) = 1$
- Y takes values from $\{ y_1, \ldots, y_m \}$ and $\Sigma_i\, p(Y{=}y_i) = 1$
- Conditional entropy of X given $Y{=}y_j$ is:

$$H(X \mid Y{=}y_j) = -\Sigma_i\, p(X{=}x_i \mid Y{=}y_j) \lg p(X{=}x_i \mid Y{=}y_j)$$

- Conditional entropy of X given Y is:

$$H(X \mid Y) = -\Sigma_j\, p(Y{=}y_j)\, \Sigma_i\, p(X{=}x_i \mid Y{=}y_j) \lg p(X{=}x_i \mid Y{=}y_j)$$

Example

- X roll of red die, Y sum of red, blue roll
- Note $p(X=1|Y=2) = 1$, $p(X=i|Y=2) = 0$ for $i \neq 1$
  - If the sum of the rolls is 2, both dice were 1
- Thus

$$H(X|Y=2) = -\Sigma_i\, p(X=x_i|Y=2) \lg p(X=x_i|Y=2) = 0$$

Example (*con’t*)

- Note $p(X{=}i, Y{=}7) = 1/6$
  - If the sum of the rolls is 7, the red die can be any of 1, …, 6 and the blue die must be 7–roll of red die
- $H(X \mid Y{=}7) = -\sum_i p(X{=}x_i \mid Y{=}7) \lg p(X{=}x_i \mid Y{=}7)$

  $= -6\ (1/6) \lg (1/6) = \lg 6$

Perfect Secrecy

- Cryptography: knowing the ciphertext does not decrease the uncertainty of the plaintext
- $M = \{ m_1, \ldots, m_n \}$ set of messages
- $C = \{ c_1, \ldots, c_n \}$ set of messages
- Cipher $c_i = E(m_i)$ achieves *perfect secrecy* if $H(M \mid C) = H(M)$

Entropy and Information Flow

- Idea: info flows from *x* to *y* as a result of a sequence of commands *c* if you can deduce information about *x* before *c* from the value in *y* after *c*
- Formally:
  - *s* time before execution of *c*, *t* time after
  - $H(x_s \mid y_t) < H(x_s \mid y_s)$
  - If no *y* at time *s*, then $H(x_s \mid y_t) < H(x_s)$

Example 1

- Command is $x := y + z$; where:
  - $0 \le y \le 7$, equal probability
  - $z = 1$ with prob. 1/2, $z = 2$ or 3 with prob. 1/4 each
- s state before command executed; t, after; so
  - $H(y_s) = H(y_t) = -8(1/8) \lg (1/8) = 3$
  - $H(z_s) = H(z_t) = -(1/2) \lg (1/2) - 2(1/4) \lg (1/4) = 1.5$
- If you know x_t, y_s can have at most 3 values, so $H(y_s \mid x_t) = -3(1/3) \lg (1/3) = \lg 3 \approx 1.58$
  - Thus, information flows from y to x

Example 2

- Command is

  if $x = 1$ **then** $y := 0$ **else** $y := 1$;

  where x, y equally likely to be either 0 or 1
- $H(x_s) = 1$ as x can be either 0 or 1 with equal probability
- $H(x_s \mid y_t) = 0$ as if $y_t = 1$ then $x_s = 0$ and vice versa
  - Thus, $H(x_s \mid y_t) = 0 < 1 = H(x_s)$
- So information flowed from x to y

Implicit Flow of Information

- Information flows from *x* to *y* without an *explicit* assignment of the form $y := f(x)$
  - $f(x)$ an arithmetic expression with variable *x*
- Example from previous slide:

  if $x = 1$ **then** $y := 0$ **else** $y := 1$;

- So must look for implicit flows of information to analyze program

Notation

- $\underline{x}$ means class of x
  - In Bell-LaPadula based system, same as “label of security compartment to which x belongs”
- $\underline{x} \le \underline{y}$ means “information can flow from an element in class of x to an element in class of y
  - Or, “information with a label placing it in class $\underline{x}$ can flow into class $\underline{y}$”

Information Flow Policies

Information flow policies are usually:

- reflexive
  - So information can flow freely among members of a single class
- transitive
  - So if information can flow from class 1 to class 2, and from class 2 to class 3, then information can flow from class 1 to class 3

Non-Transitive Policies

- Betty is a confident of Anne
- Cathy is a confident of Betty
  - With transitivity, information flows from Anne to Betty to Cathy
- Anne confides to Betty she is having an affair with Cathy's spouse
  - Transitivity undesirable in this case, probably

Non-Lattice Transitive Policies

- 2 faculty members co-PIs on a grant
  - Equal authority; neither can overrule the other
- Grad students report to faculty members
- Undergrads report to grad students
- Information flow relation is:
  - Reflexive and transitive
- But some elements (people) have no “least upper bound” element
  - What is it for the faculty members?

Confidentiality Policy Model

- Lattice model fails in previous 2 cases
- Generalize: policy $I = (SC_I, \leq_I, join_I)$:
  - SC_I set of security classes
  - $\leq_I$ ordering relation on elements of SC_I
  - $join_I$ function to combine two elements of SC_I
- Example: Bell-LaPadula Model
  - SC_I set of security compartments
  - $\leq_I$ ordering relation *dom*
  - $join_I$ function *lub*

Confinement Flow Model

- $(I, O, confine, \rightarrow)$
  - $I = (SC_I, \leq_I, join_I)$
  - O set of entities
  - $\rightarrow$: $O \times O$ with $(a, b) \in \rightarrow$ (written $a \rightarrow b$) iff information can flow from a to b
  - for $a \in O$, $confine(a) = (a_L, a_U) \in SC_I \times SC_I$ with $a_L \leq_I a_U$
    - Interpretation: for $a \in O$, if $x \leq_I a_U$, information can flow from x to a, and if $a_L \leq_I x$, information can flow from a to x
    - So a_L lowest classification of information allowed to flow out of a, and a_U highest classification of information allowed to flow into a

Assumptions, *etc*.

- Assumes: object can change security classes
  - So, variable can take on security class of its data
- Object *x* has security class $\underline{x}$ currently
- Note transitivity *not* required
- If information can flow from *a* to *b*, then *b* dominates *a* under ordering of policy *I*:

  $(\forall\, a, b \in O)[\, a \rightarrow b \Rightarrow a_L \leq_I b_U \,]$

Example 1

- SC_I = { U, C, S, TS }, with U $\leq_I$ C, C $\leq_I$ S, and S $\leq_I$ TS
- $a, b, c \in O$
  - confine(a) = [C, C]
  - confine(b) = [S, S]
  - confine(c) = [TS, TS]
- Secure information flows: $a \rightarrow b$, $a \rightarrow c$, $b \rightarrow c$
  - As $a_L \leq_I b_U$, $a_L \leq_I c_U$, $b_L \leq_I c_U$
  - Transitivity holds

Example 2

- SC_I, $\leq_I$ as in Example 1
- $x, y, z \in O$
  - confine(x) = [C, C]
  - confine(y) = [S, S]
  - confine(z) = [C, TS]
- Secure information flows: $x \rightarrow y$, $x \rightarrow z$, $y \rightarrow z$, $z \rightarrow x$, $z \rightarrow y$
  - As $x_L \leq_I y_U$, $x_L \leq_I z_U$, $y_L \leq_I z_U$, $z_L \leq_I x_U$, $z_L \leq_I y_U$
  - Transitivity does not hold
    - $y \rightarrow z$ and $z \rightarrow x$, but $y \rightarrow z$ is false, because $y_L \leq_I x_U$ is false