Lucas Harrison

LMH2538 lucash

harrisonlucas@utexas.edu

Lecture 17

1. If a computer system complies with the BLP model, does it necessarily comply with non-interference? Why or why not?

Yes it does because any MLS policy (BLP is a MLS policy) can be rewritten as an NI policy.

2. What would the NI policy be for a BLP system with subjects: A at (Secret: Crypto), B at (Secret: Nuclear)?

There would be no arrow between A and B. They could both read from lower levels of security but since they have different functional groups, they would not communicate directly.

3. Can covert channels exist in an NI policy? Why or why not?

No they cannot if it is a true NI policy. There cannot be a single case where L can see any activities of H. If L can see something, then a covert channel is possible, but the NI policy is not intact. The tough part is proving that a system is noninterference.

4. If be NI policy is *A*- > *B,* in a BLP system what combinations often levels "high" and "low" could A and B have?

It could have any combination. A or B could be high, and A or B could be low. NI policy only dictates which way information can flow and if it were the case of a system where info only flowed from high to low then A could be high and B be low or visa versa if the system was a BLP model.

Lecture 18

1. Why do NI policies better resemble metapolicies than policies?

It is difficult to prove that a system is noninterfering and an NI policy doesn't actually address any of the real system implementation details that a policy deals with.

2. What would be L's view often following actions: hl, 11, h2, h3, ... , hj, 12, 13, ... , lk

L would see l1, l2, l3, lk. Under no circumstance can that view be different based on what H is doing/not doing.

3. What is difficult about proving NI for realistic systems?

Its difficult to test every scenario that H could do and show that absolutely zero things are different from L's point of view based off of H's actions/inaction.

Lecture 19

1. Explain the importance of integrity in various contexts.

Integrity is important for any business. Customer's data shouldn't be able to be changed by anyone. If I could change someone's amazon account shipping address every time they ordered something, I could get a bunch of free stuff. If I could pretend to be the dean of UT, I could change all my grades and say I've paid full tuition every semester. There's tons of examples of integrity being crucial aspects of security.

2. Why would a company or individual opt to purchase commercial software rather than download a similar, freely available version?

It may have more integrity than a free version. That could come in the form of guarantees, tech support, or just overall support.

3. Explain the difference between separation of duty and separation of function.

Separation of duty requires several subjects to complete a part of a critical task. Separation of function prevents one subject from doing all the critical parts of a task.

4. What is the importance of auditing in integrity contexts?

It allows the potential for fraud detection and backup when a fraud occurs. Auditing makes your product accountable and reliable. It is one reason your product may be purchased over a free alternative because your auditing allows for some level of accountability to the customer.

5. What are the underlying ideas that raise the integrity concerns of Lipner?

Separation of function has the programmers write code on a non-productive system then transferred to the workforce who will use it. Auditing log this process and make sure they remain separate. Separation of duty makes sure managers and auditors have access to these logs to spread the accountability across several individuals.

6. Name a common scenario where integrity would be more important than confidentiality.

Banking. It's not the end of the world if my bank account balance can be seen by anyone. It is the end of the world if my bank account balance can be accessed by anyone.

Lecture 20

1. Give examples of information that is highly reliable with little sensitivity and information that is not so highly reliable but with greater sensitivity.

2. Explain the dominates relationships for each row in the table on slide 4.

Its very much the same as the BLP dominates relationship. Row one has an expert, physics and a student, physics. They have the same groups but the expert is higher than the student so that is the dominate subject. The next row, the level is novice trying to dominate an expert. That doesn't work. The third row is a student dominating a novice which works.

3. Construct the NI policy for the integrity metapolicy.

HL. We do not want information to flow up in integrity. Low level integrity info can't corrupt higher integrity info.

4. What does it mean that confidentiality and integrity are "orthogonal issues?"

It simply means that they are independent security concerns. So separate that they need their own labels so that each can be addressed fully.

Lecture 21

1. Why is Biba Integrity called the "dual" of the BLP model?

Because it is the opposite of BLP in practice. Anywhere you can write in BLP is a read in Biba. And anywhere you can read in BLP, you can write in Biba.

2. Why in the ACM on slide 5 is the entry for Subj3 - Obj3 empty?

Because Subj3 does not dominate Obj3 since Obj3's groups aren't a super set of Subj3's.

3. If a subject satisfies confidentiality requirements but fails integrity requirements of an object, can the subject access the object?

No that would invalidate the metapolicy of the system.

Lecture 22

1. What is the assumption about subjects in Biba's low water mark policy?

Writes are the same as in strict integrity where you can't write up. A subject can read anything but if reading down, the subjects level floats down to the level of the information being read.

2. Are the subjects considered trustworthy?

No they are not. On the off chance that you read bad info, your level floats down to the level at which the bad info came from.

3. Does the Ring policy make some assumption about the subject that the LWM policy does not?

It assumes the subject is competent enough to filter out bad information.

4. Are the subjects considered trustworthy?

Yes they are. They are trusted to not believe/use the bad information that may be at lower integrity levels.

Lecture 23

1. Are the SD and ID categories in Lipner's model related to each other?

No they are not. SD is a confidentiality category and ID is a integrity category.

2. Why is it necessary for system controllers to have to ability to downgrade?

Because they have to be able to change the label of an object from development to production.

3. Can system controllers modify development code/test data?

No they cannot.

4. What form of tranquility underlies the downgrade ability?

Weak tranquility. With a strong tranquility the ability to downgrade would be in violation of the metapolicy.

Lecture 24

1. What is the purpose of the four fundamental concerns of Clark and Wilson?

The overall goal of their four concerns is to maintain consistency across a large, even global, commercial system.

2. What are some possible examples ofCDis in a commercial setting?

A constrained data item could be groceries at HEB or tickets at a football game. These items need to be audited and maintain integrity.

3. What are some possible examples ofUDis in a commercial setting?

An examples of unconstrained data items could be coupons at HEB because once they are put out, they don't need to be audited and kept track of who is taking them.

4. What is the difference between certification and enforcement rules?

Enforcement rules are there to ensure the transactions are done with compliance to the certification rules. The certification rules are there to ensure the validity and legality of a transaction procedure.

5. Give an example of a permission in a commercial setting.

Permissions come in a set of triples. An example could be:

(Me, withdraw money, {ATMs, my bank account}])

Lecture 25

1. Why would a consultant hired by American Airlines potentially have a breach of confidentiality if also hired by United Airlines?

If the metapolicy allows reads to higher levels of security access, then it is very possible that while consulting for American Airlines, the employee could access sensitive information and then bring it over to United Airlines. Even if a read up wasn't allowed, what might be considered low level info for American Airlines could still be too sensitive to want it to get out to the competition.

2. In the example conflict classes, if you accessed a file from GM, then subsequently accessed a file from Microsoft, will you then be able to access another file from GM?

Yes because accessing a file inside a conflict class only prohibits access to other objects from companies in that conflict class. So you wouldn't be able to access Ford or Chrysler but since Microsoft is in another conflict class, accessing it has no effect on being able to access GM.

3. Following the previous question, what companies' files are available for access according to the simple security rule?

Anything in the company class of GM, Microsoft, and the a single company of the other conflict class.

4. What differences separate the Chinese Wall policy from the BLP model?

It is sensitive to what you've done in the past where as in the BLP model, you could dynamically change your access (assuming strong tranquility isn't in the metapolicy and there is a safe and organized way to do it) and then access more/less info.

Lecture 26

1. What benefits are there in associating permissions with roles, rather than subjects?

Its much easier to change them dynamically. By assigning permissions to all the tellers at a bank is a lot quicker than doing it on an individual basis for each teller.

2. What is the difference between authorized roles and active roles?

Authorized roles are allowed to be filled at various times but not necessarily all the time. Active roles are the roles that are currently occupied by the person.

3. What is the difference between role authorization and transaction authorization?

Role authorization makes sure any role that is active for a subject is a valid authorized role. Transaction authorization makes sure that a transaction a subject wants to do is valid for that active role.

4. What disadvantages do standard access control policies have when compared to RBAC?

They are much less flexible in terms of updating individual subjects permissions. Having a list of authorized roles makes it easier to switch between them and fully assess any security flaws associated with an individual. Also, by updating permisions on a per role basis is a lot easier than on an individual basis.

Lecture 27

1. Why would one not want to build an explicit ACM for an access control system?

It could get extremely large. By computing the permissions on the fly based off of a set of rules, it can save a lot of space.

2. Name, in order, the ACM alternatives for storing permissions with objects, storing permissions with subjects and computing permissions on the fly.

Access control list, capability based system, maintain a set of rules to compute accesses on the fly.

Lecture 28

1. What must be true for the receiver to interpret the answer to a ''yes'' or ''no'' question?

The reciever knows that the sender has one of the two possibilities but doesn't know which.

2. Why would one want to quantify the information content of a message?

3. Why must the sender and receiver have some shared knowledge and an agreed encoding scheme?

Without this, the recieve has no way to interpret the "yes" or "no". If they haven't determined if a 0 is a yes or a no, then the message is useless.

4. Why wouldn't the sender want to transmit more data than the receiver needs to resolve uncertainty?

Because it adds noise. The reciever can't tell when to start and stop and how to interpret. With extra data, its easier for noise to effect the actual message and throw off the reciever.

5. If the receiver knows the answer to a question will be ''yes,'' how many bits of data quantify the information content? Explain.

Zero. If the reciever knows the answer already then the question doesn't need to be asked. If the literal asking of the question is part of the answer, then one bit would be needed. Perhaps by asking the question, the reciever confirms an object exists, or is still on, or measures the timeing. Those are some cases where you would need that 1 bit even though you already know the answer.

Lecture 29

1. How much information is contained in each of the first three messages from slide 2?

N bits, 4 bits to convey the 10 digits, 7 bits to conver they 100 possibilites.

2. Why does the amound of information contained in ''The attack is at dawn'' depend on the receiver's level of uncertainty?

Because the message could be interpreted right to left or as digits or something completely different from "the attack is at dawn" if the reciever is uncertain about the way to decode the message. The message also doesn't specifiy things like what day, or how many troops. This can be a problem based on how uncertain the reciever is.

3. How many bits of information must be transmitted for a sender to send one of exactly 16 messages? Why?

Four because 2^4 is 16 and there are 16 possibilities.

4. How much information content is contained in a message from a space of 256 messages?

log2(256) = 8. 8 bits are sent but if the messages possible were known before hand then the amount of info contained in a message is arbitrary and depends on each messages corresponding info.

5. Explain why very few circumstances are ideal, in terms of sending information content.

Realisticaly, its not common for the reciever to know a complete set of messages that could be sent.

Lecture 30

1. Explain the difference between the two connotations of the term "bit."

One refers to the literal binary digit (a zero or a one). The other is refering to the information conveyed which is a continous stream.

2. Construct the naive encoding for 8 possible messages.

For 8 possible messages, then you would require 3 bits. It would be 000, 001, 010, 011, 100, 101, 110, 111

3. Explain why the encoding on slide 5 takes 995 + (5 * 5) bits.

Since message 10 will be sent almost always, it makes sense to assign that to a single bit. For example, if the sender sends the bit 0, then that corresponds to message 10. This cuts down the number of bits from 4 to 1 for 99.5% of the messages. The other messages are an extra bit long, however, so that way when a 1 is recieved we know we need to listen for another 4 bits. If that first bit is not a 1, then we know its message 10.

4. How can knowing the prior probabilities of messages lead to a more efficient encoding?

As demonstrated in the example above, knowing the probability of a predetermined message can let you assign small bit values that correspond to popular messages. This reduces the average amount of bits needing to be sent per message.

5. Construct an encoding for 4 possible messages that is worse than the naive encoding.

0, 10, 001, 011. Suppose each message is 25% of being sent. This would be worse than naive because the last two messages require an extra bit so now the average bit per message goes from 2 bits to greater than 2 bits.

6. What are some implications if it is possible to find an optimal encoding?

You would need to know some concrete probabilities of the messages. The messages would have to be predetermined, ie not random ascii messages.

Lecture 31

1. Name a string in the language consisting of positive, even numbers.

"2468462482468"

2. Construct a non-prefix-free encoding for the possible rolls of a 6-sided die.

1 = 1, 2 = 0, 3 = 01, 4 = 00, 5 = 11, 6 = 001. If the reciever gets a 1, it can't tell if that is referring to a '1' roll or to a '5' roll. They share the same prefix.

3. Why is it necessary for an encoding to be uniquely decodable?

If the decodings are not unique, then the reciever's understanding of the information is arbitrary since their is no way to differentiate between messages that have the same decoding.

4. Why is a lossless encoding scheme desirable?

Lossless is important because it allows the reciever to completely recover the original transmission. This makes sure there isn't any lost information and prevents ambiguity/confusion.

5. Why doesn't Morse code satisfy our criteria for encodings?

It isn't because it is not streaming since there is a break between dots. The reciever can't differentiate between a single dot referring to a an 'e' character or being one of several dots referring to an 's' character.

Lecture 32

1. Calculate the entropy of an 8-sided, fair die (all outcomes are equally likely).

Since they are all equally probable, h = -(log(1/8).

2. If an unbalanced coin is 4 times more likely to yield a tail than a head, what is the entropy of the language?

h = -(4/5*log(4/5) + 1/5*log(1/5)).

3. Why is knowing the entropy of a language important?

Knowing the entropy helps know how close to optimal your encoding scheme is. If you calculate the entropy and find that by switching your encoding scheme you lower your entropy, it is easier to determine if you have an optimal solution or not.

Lecture 33

1. Explain the reasoning behind the expectations presented in slide 3.

Since HH is the most likely outcome by far, using only one bit to send that message will result in .5 bits per flip when they are both H. This lowers the average bits per symbol since that is most likely. It outweighs the fact that TH and TT are 1.5 bits per flip over higher number of fips.

2. Explain why the total expected number of bits is 27 in the example presented in slide 4.

By adding up the total bits sent using the new encoding we see that only 27 bits are needed to convey the same message that would require 32 bits on a naive encoding. The efficieny is rather close to entropy and is a big step in the right direction.

3. What is the naive encoding for the language in slide 5?

000, 001, 010, 011, 100, 101

4. What is the entropy of this language?

5,6 = x. 3/4 is three times more likely than 5. 3/4 = 3x. 1/2 is twice as likely as 3/4.

1/2 = 6x. So thats 6x+6x+3x+3x+x+x = 20x.

h = -(6/20*log(6/20) + 6/20*log(6/20) + 3/20*log(3/20) + 3/20*log(3/20) + 1/20*log(1/20) + 1/20*log(1/20)).

5. Find an encoding more efficient than the naive encoding for this language.

0, 10, 110, 1110, 11110, 11111.

6. Why is your encoding more efficient than the naive encoding?

Its more efficient because it has a lower entropy and uses less bits because 1/2 are so much more likely.