Lucas Harrison

LMH2538

lucash

harrisonlucas@utexas.edu

Lecture 53

1. Why is it important for a digital signature to be non-reusable?

It should be 'bound' to a document or transaction. This prevents the copying and reusing of a signature for an unauthorized purpose.

2. Why is it the hash of the message typically signed, rather than the message itself?

Because public key encryption is expensive and the message can be an arbitrary length where the hash is a fixed size.

3. What assurance does R gain from the interchange on slide 4?

That the message is authentic because the message is encrypted with ks-1. That means that only s could have encrypted the message. It becomes tamper proof as well because no one except r can get into the actual message because it requires r's private key.

Lecture 54

1. What is the importance of certificate authorities?

They are important because they can be used as a third party to establish trust between two entities that are mutually suspicious of each other.

2. In the example on slide 5, why does X sign the hash of the first message with its private key?

X uses its private key as a form of authentication. X is known and trusted and hashes the first message of Y and its public key into a message that is sealed by its private key. Now when Y wants to send a message, it can send its message along with X's certificate and if the hashes match, its likely nothing was tampered with. All this assumes X is trusted and already has its identity and public key established.

3. Why is it necessary to have a hash of Y and Ky?

Because the chances of a spoofer using fake values for Y and Ky producing the same hash as the actual values of Y and Ky are very low. Therefore, when Y presents the certificate to another party, that party can use X's public key to compare the 'correct hash' as defined by X and the one it just received from Y. If they match then the new party can be relatively assured that Y is who they say they are and not a scam.

4. What would happen if Z had a public key for X, but it was not trustworthy?

Then the entire purpose is undermined. Without X being trustworthy, then there is nothing to be gained by having X verify Y's identity.

Lecture 55

1. What happens at the root of a chain of trust?

The root of a chain of trust needs to be an unimpeachable authority. If that root is no longer trustworthy, every certificate that leads back to it is no longer valid.

2. Why does an X.509 certificate include a “validity interval”?

The validity interval is important because it prevents abuse/unauthorized reuse of a certificate. It also makes it so credentials are checked regularly which provides more validity to the certificate.

3. What would it mean if the hash and the received value did not match?

It would mean that at least one of the fields did not match its original value and therefore should not be trusted. It could be harmless but it could also be an intentional attack. For example, if someone changed the validity interval of an expired certificate so that it was no longer expired and sent it the hashes should be different and would warn the receiver not to trust that certificate.

Lecture 56

1. What are some protocols previously discussed?

2. What may happen if one step of a protocol is ignored?

The desired results might not occur. It could be vulnerable to attack.

3. Why must the ciphers commute in order to accomplish the task in slide 4?

Because you can’t reach inside the outer encryption to undo the inner encryption. Most protocols/encryption algorithms don’t have that property.

4. Describe how an attacker can extract M from the protocol in slide 6.

An attacker could gather all three of the messages. Then, the attacker would xor message 2 and 3 leaving ka. Then the attacker would xor that with message 1 leaving m.

5. Describe how an attacker can extract Ka from the protocol in slide 6.

Once the attacker has m as described above, they could xor m with message 1 and get ka.

6. Describe how an attacker can extract Kb from the protocol in slide 6.

Once you have m and ka, the attacker could xor them and then xor that with message 2 leaving kb.

7. Why are cryptographic protocols difficult to design and easy to get wrong?

They are tough to prove correctness when trying to deal with everything bad that could happen. Many problems with them are very subtle.

Lecture 57

1. Explain the importance of protocols in the context of the internet.

Protocols are important to the internet because there has to be a fixed way messages are sent and received between parties across the internet. Devices are so different and it doesn’t make sense to leave anything device specific. Instead, there is a protocol in place to make sure all the messages are interpreted in a uniform manner.

2. Explain the importance of cryptographic protocols in the context of the internet.

Cryptographic protocols are important for obvious reasons such as integrity, authenticity, and confidentiality. These along with several other properties are what allow business, banking, shopping, etc. to take place on the internet.

3. What are the assumptions of the protocol in slide 6?

That there are public keys in place for A and B and each has access to these keys.

4. What are the goals of the protocol in slide 6?

To share a secret key between A and B.

5. Are the goals of the protocol in slide 6 satisfied? Explain.

Yes the goals are satisfied. Both A and B now have a secret key K that they can use for securely sending messages between them.

6. How is the protocol in slide 6 flawed?

Professor showed us in class. Basically there is a way for a third party who has access to the public keys of A and B to copy A's original message to B and resend it encrypted with its private key. Then when B responds, the third party can gain access to its private by using A's public key. Not sure if I explained that correctly…but the flaw is that there is a way for a third party to use this protocol to gain access to a private key.

Lecture 58

1. Why is it important to know if a protocol includes unnecessary steps or messages?

Unnecessary steps are important to identify because it may be possible for an attacker to glean information off of these messages or steps. Also, these unnecessary steps may not be checked as rigorously and changes to the original message may go unnoticed. This could open up some new forms of attacks.

2. Why is it important to know if a protocol encrypts items that could be sent in the clear?

The encrypted information may not be available at appropriate times to the receiver. It adds extra steps of decryption which could slow down the protocol.

Lecture 59

1. Why might it be difficult to answer what constitutes an attack on a cryptographic protocol?

It could be difficult to classify what information is sensitive. Is someone gathering all a system's public keys really that threatening? Is someone copying all your encrypted messages bad? It could open the door to replay attacks.

2. Describe potential dangers of a replay attack.

The attacker could confuse or disrupt normal operations of the system. Replaying messages could be used to glean information on what the contents of a message is or what action is taken when that message is received.

3. Are there attacks where an attacker gains no secret information? Explain.

Yes. An interleaving attack or replay attack could lead to no new secret information gained. However, it could disrupt normal operations of a system and give the attacker power in that way.

4. What restrictions are imposed on the attacker?

That their messages can't be arbitrary. It's assumed they can see all traffic but the protocol should be robust enough to deal with a determined attacker.

5. Why is it important that protocols are asynchronous?

So that way systems can operate normally until they receive a message. Then when they do they know how to handle the message and respond.

Lecture 60

1. Would the Needham-Schroeder protocol work without nonces?

I think it could still work in that A could still determine if the message returned from B is fresh and not a replay message. However, since timestamps are predictable, I imagine it would leave the protocol much more vulnerable to attack.

2. For each step of the NS protocol, answer the two questions on slide 5.

1: A is telling S that it wants to talk to B securely. S believes that A and B want to talk. 2: S returns the message to A and supplies it a key. The whole message is encrypted with Kas and then part of the message that A is going to forward to B is encrypted with Kbs. A believes the key Kab is fresh and new. 3: A forwards a copy of S's message that's encrypted with Kbs. B believes that the key received from A is from S since its encrypted with Kbs. Therefore, B believes its okay to talk to A with Kab. 4: B sends A a nonce encrypted with their key. A believes B received the message and decrypted it using Kbs so it now has Kab. 5: A returns the nonce -1. B now believes that A has the key Kab because it would be needed to decrypt Nb and then perform an operation on it and then encrypt using Kab and send back to B.

Lecture 61

1. As in slide 5, if A's key were later changed, after having Kas compromised, how could A still be impersonated?

Once you have Kas you can open communication with any other entity on the system so once that is compromised, you can continue to impersonate A.

2. Is it fair to ask the question of a key being broken?

Sort of. Key security isn't necessarily a problem for the protocol. The protocol assumes the private keys remain private. That assumption is an extremely important and difficult thing to implement.

3. How might you address these flaws if you were the protocol designer?

By protecting more steps with nonces to make sure they're 'fresh' messages.

Lecture 62

1. What guarantees does Otway-Rees seem to provide to A and B?

It seems that each of the messages is guaranteed to be fresh. They are wrapped around a session identifier, M, and also uses nonces.

2. Are there guarantees that Needham-Schroeder provides that Otway-Rees does not or vice versa?

Replay messages seem to not be an alternative in Otway-Rees due to the session identifier. If there was a way to verify that session M was current and valid when receiving messages then A and B could thwart any replayed messages.

3. How could you fix the flawed protocol from slide 4?

Lecture 63

1. Why is the verification of protocols important?

It is important to know that your protocol is secure. However, verifying a protocol acts the way you want it to is difficult because it is tough to determine everything a spy might try and do.

2. What is a belief logic?

It allows one to reason out what different users in the protocol should expect to believe during different phases of the protocol. This allows short abstract proofs but may miss some important flaws in the protocol.

3. A protocol is a program; where do you think beliefs come in?

Beliefs are assumptions that can represent an initial state or the outcome of a transition between states.

Lecture 64

1. What is a modal logic?

Standard, proposition, and predicate logic with some extra primitives and rules of inference to reason about a particular domain, ie belief.

2. Explain the intuition behind the message meaning inference rule.

The intuition here is that if A and B have a secret key known only by them and A sees a message encrypted with this key, it must be the case that B said the message.

3. Explain the intuition behind the nonce verification inference rule.

The intuition here is that if A receives a fresh message, X, from B, then A believes that B believes X.

4. Explain the intuition behind the jurisdiction inference rule.

The intuition here is that if B is has more authority on something than X and B agrees with X, then A believes X.

5. What is idealization and why is it needed?

The goal is to omit parts of the message that do no impact the beliefs of the recipients. It is needed to get from protocol steps to logical inferences.

Lecture 65

1. Why do you think plaintext is omitted in a BAN idealization?

Because it doesn't do anything to affect the beliefs of the recipient. Plain text messages don't change the way the system is viewing things.

2. Some idealized steps seem to refer to beliefs that will happen later in the protocol. Why would that be?

In order for the beliefs to remain consistent, they sometimes appear earlier in the idealized protocol.

3. One benefit of a BAN proof is that it exposes assumptions. Explain that.

For example, if when creating an idealized version of the protocol, you realize that everybody believes S. That is an assumption that has huge consequences for the system if it ever becomes the case that S shouldn't be believed. Now you can use that assumption and make sure it is implemented in a way that it always remains true. These types of assumptions can help you find weaknesses in your protocol and address them.