CS361 Intro to Computer Security Daniel Rosenwald
Young, Bill July 3rd, 2014

EID: dpr447
CS login: randose
Email: danielrosenwald@gmail.com

Week 4

Lecture 53

1. Digital signatures, just like real signatures, should be non-reusable so that they cannot be used for transactions which the signator did not actually authorize.
2. Because public-key encryption is very expensive, so for variable-sized messages we use the hash.
3. R sees that it actually came from S and R is the only one who can remove the outer layer of encryption.

Lecture 54

1. They vouch for the binding of an agent to its key.
2. In order to assure that the values weren't changed.
3. The hash helps us check for changed values.
4. Then X would not be a legitimate certifying agent.

Lecture 55

1. The root has a company whose business it is to verify the certification of agents. They have a monetary incentive to be thorough, so we can trust them.
2. To make sure that these certificates are only valid for a certain amount of time – in other words, they expire.
3. If those values did not match, it would mean that the certificate is invalid.

Lecture 56

1. TCP/IP
2. Then the security of the whole transaction could be compromised.
3. The ciphers have to commute so that we can encrypt the message in either order.
4. XOR the message with the XOR of the first two.
5. XOR the last one with the second one.
6. XOR the first one with the second one.
7. They're hard to design because there are very few commutable encryption methods and so there aren't many options and ways to get around flaws.

Lecture 57

1. Almost everything that happens on the Internet is through a protocol. Protocols are simply structured dialogue, and when sending information between systems, the protocol is what allows those communications to be sent and heard correctly and successfully.

2. Likewise, cryptographic protocols are important for the same reasons plus unicity, integrity, authenticity, confidentiality, non-repudiation of origin, and non-repudiation of receipt.
3. Assumes that there's a public-key infrastructure in place, and they each have their own keys in place.
4. The goals are to make sure A knows he's talking to B and vice versa.
5. The goals seem to be satisfied, however I feel as though there is not secure enough of a system in place.
6. Perhaps the flaw is that B is given too much leeway and can change the contents of the message when he sends it back. Or, that B will not know that A has received the response message, so an endless cycle of confirmations must be sent.

Lecture 58

1. Because it would be a waste of time to include unnecessary steps – it wouldn't be as efficient as possible.
2. Similarly, if items are being encrypted that could be sent in the clear, then you're wasting time.

Lecture 59

1. It's difficult to answer what constitutes an attack because we don't know all of the mechanisms which attackers might use to slip through the cracks.
2. Replay attack could be dangerous if secret government data is leaked and then replayed for public hearing, like a military strike.
3. Yes, for example an interleaving attack can screw up the line of communication between two parties without actually gaining information.
4. Attackers have some limits.
5. Protocols must be asynchronous because of the nature of message-sending – when I receive a message, I start my run of the protocol, whereas you started yours when you sent the message.

Lecture 60

1. Yes, but it would need something that with high probability has not been used before.
2. For message 1, the sending is trying to say here's my key, B's key, and a nonce, and then the receiver should believe it should start to make a key. For message 2, S says here's some keys, you know what to do, and A knows what to do. So for message 3, it is sending B the Kbs encrypted key, and B should see that this is step 3 of the NS protocol and decrypt it. For message 4, B goes on to encrypt a new nonce and encrypts it with the new session key and sends to A to demonstrate that it has the key, while A should know that this means B has the key. Finally, A sends the nonce - 1 back to B with the session key, showing that A has the key and can decrypt and encrypt with it.

Lecture 61

1. A could still be impersonated by using the same key on the old session.

2. Yes, because it is in the scope of the protocol.
3. I'd considering nonce-protecting step 3.

Lecture 62

1. Provides that both parties are authenticated throughout the whole process.
2. OR doesn't guarantee that both parties know they other has the key, while NS doesn't provide authentication throughout.
3. You would implement an extra layer of encryption.

Lecture 63

1. Verifying protocols is important to ensure that they are secure and not vulnerable to attack.
2. A belief logic is a system that allow reasoning about what principals within the protocol should be able to infer from the messages they see. In other words, it's a formal system for reasoning about beliefs.
3. Beliefs come in where the code leaves off – the belief is the theory behind the protocol.

Lecture 64

1. Modal logic has primitives and operators.
2. It simply follows that if a K is used between A and B, and A receives a message with that key, he assumes it's from B.
3. This is simply nonce verification.
4. This is the commutative property of experience.
5. Idealization turns the message sent into its semantics.

Lecture 65

1. I think plaintext is omitted in a BAN idealization because it can be forged.
2. Because we need to establish a set of initial beliefs that come into play later.
3. BAN doesn't leave anything to be implicit. Every single thing that happens is attempted to be exposed explicitly to ensure that no security flaws are allowed.