Cheats, Liars, Posers and Thieves

Cryptographic Protocols for Insecure Networks

Rob Beezer
beezer@ups.edu

Department of Mathematics and Computer Science
University of Puget Sound

Digipen Institute of Technology
April 13, 2006

Encryption — Plaintext

The Declaration of Independence

When in the Course of human events it becomes necessary for one people to dissolve the political bands which have connected them with another and to assume among the powers of the earth, the separate and equal station to which the Laws of Nature and of Nature's God entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the separation.

Encryption — Ciphertext

```
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.1 (Cygwin)

hIwDeLDTlhkdcSkBBACRhTX548lcVPGPBot+zQ4IdZnKyzVRoZC9+QIyWj5fmhFZ
W05xi4T0bG/AAhIlD9eVuazq2F4sDpag2IcbbjwHmmFVuZ+fqfSVy6OYccoNiECk
CqfMyEuCD86CTeQ9V56HkPvWDydkkaKO9vU7Altr9Z5M/O7L8hmyxYM1pwoX2Ywm
BAMDApBQgVUojNHRYK4D8w18BapW3uHBxX4Mi7mYDEx8mi6gai/JwFJHoL/92WZj
reou1ox47Z6xfkSwLyeHBVyapRYRzPrc37v8r4GV8JCsl4g7FmSiCVregfYENpdQ
ZecH/X7U29rsclD85F12K69t67sHX6+OMfjexI8SP1vYW8L7cj4mpls3SsHKr82V
jPKnMGBT6fmZYts20C2gs3u2AcTpPpvg1rbiyCvGIliHPcMsuMRL60nTudrw20Su
R1Ae27bwHTbfejgiZrN9s60f2fPfz/clLdKvWpdtzjFlAOK4k8qlsbFnQ8N5e59g
D5t5SQ1VhNbvP7MPJXrJ5walY53SW1oAYyBqECOmjasEiUn4JULWHEnsmUXCGDAN
BQJE3PvVK8MNi5AnTiEchovj4A5VX92lOZET
=MvTv
-----END PGP MESSAGE-----
```

Base 64 Alphabet: a. . . z, A. . . Z, 0. . . 9, /, + $(26 + 26 + 10 + 2 = 64)$

Caesar Shift

Plaintext

```
ATTACK AT DAWN
```

Algorithm: Shift every character forward in the alphabet

Key: Shift each character 10 characters forward

Ciphertext

```
KDDKMU KD NKGW
```

Caesar Shift

Plaintext

```
ATTACK AT DAWN
```

Algorithm: Shift every character forward in the alphabet

Key: Shift each character 10 characters forward

Ciphertext

```
KDDKMU KD NKGW
```

Analysis:

- Only 25 keys to test, one will produce English output.
- A shift by 13 is nice, since the same function decrypts (ROT13).
- Silly example, but contains fundamental ideas.

Symmetric Encryption Function

K is a piece of secret information, shared between Alice and Bob
M is a message to be sent from Alice to Bob

$E(k, x)$ is an agreed-upon, public, encryption function
$D(k, x)$ is an agreed-upon, public, decryption function

Alice computes ciphertext $C = E(K, M)$ and transmits
Bob receives, computes

$$D(K, C) = D(K, E(K, M)) = M$$

to recover plaintext

Examples: AES, IDEA, DES
Sometimes $D = E$! (e.g. German Enigma, Federal Standard DES)

Asymmetric Post Office Encryption

Alice thinks the Post Office is steaming open her mail to Bob.

1. Alice composes a message for Bob.
2. Alice places the message in strongbox.
3. At the Post Office, Alice requests a "Bob-Lock."
4. Bob has provided the Post Office with infinitely many padlocks.
5. All of these locks are identical, and Bob has the only key.

Asymmetric Post Office Encryption

Alice thinks the Post Office is steaming open her mail to Bob.

1. Alice composes a message for Bob.
2. Alice places the message in strongbox.
3. At the Post Office, Alice requests a “Bob-Lock.”
4. Bob has provided the Post Office with infinitely many padlocks.
5. All of these locks are identical, and Bob has the only key.
6. Alice attaches a Bob-Lock to the box.
7. Alice mails the box, knowing it is tamper-proof.
8. Bob receives the locked box, knowing it has not been opened.

Asymmetric Post Office Encryption

Alice thinks the Post Office is steaming open her mail to Bob.

1. Alice composes a message for Bob.
2. Alice places the message in strongbox.
3. At the Post Office, Alice requests a "Bob-Lock."
4. Bob has provided the Post Office with infinitely many padlocks.
5. All of these locks are identical, and Bob has the only key.
6. Alice attaches a Bob-Lock to the box.
7. Alice mails the box, knowing it is tamper-proof.
8. Bob receives the locked box, knowing it has not been opened.
9. Bob unlocks the box with the one key, and reads the message.

Asymmetric Encryption Function

Bob has a public key, P_B, and secret key, S_B
M is a message to be sent from Alice to Bob

$E(k, x)$ is an agreed-upon, public, encryption function
$D(k, x)$ is an agreed-upon, public, decryption function

Alice computes ciphertext with Bob's public key $C = E(P_B, M)$
Bob recovers plaintext with his secret key

$$D(S_B, C) = D(S_B, E(P_B, M)) = M$$

Examples: RSA, Diffie-Hellman Knapsack

Hash Functions

- Variable length input, possibly megabytes or gigabytes
- Fixed, small output, e.g. 128, 160, or 512 bits

Hash Functions

- Variable length input, possibly megabytes or gigabytes
- Fixed, small output, e.g. 128, 160, or 512 bits
- Easy to compute an output of the function
- Given an output, hard to find an input that yields that output
- Hard to find two inputs that yield same output

Hash Functions

- Variable length input, possibly megabytes or gigabytes
- Fixed, small output, e.g. 128, 160, or 512 bits
- Easy to compute an output of the function
- Given an output, hard to find an input that yields that output
- Hard to find two inputs that yield same output
- Public, the world only needs one good hash function
- Examples: MD5, SHA1, RIPE-MD
- Also called: one-way function, compression function, message digest

A Hash Function At Work

First paragraph from The Declaration of Independence, SHA1 Hash

```
B1AF 88EA 4F58 76D4 CD58 0EC4 5A56 E170 78CD D3AE
```

A Hash Function At Work

First paragraph from The Declaration of Independence, SHA1 Hash

```
B1AF 88EA 4F58 76D4 CD58 0EC4 5A56 E170 78CD D3AE
```

The Declaration of Independence, Revised

Vhen in the Course of human events it becomes necessary for one people to dissolve the political bands which have connected them with another and to assume among the powers of the earth, the separate and equal station to which the Laws of Nature and of Nature's God entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the separation.

A Hash Function At Work

First paragraph from The Declaration of Independence, SHA1 Hash

```
B1AF 88EA 4F58 76D4 CD58 0EC4 5A56 E170 78CD D3AE
```

The Declaration of Independence, Revised

Vhen in the Course of human events it becomes necessary for one people to dissolve the political bands which have connected them with another and to assume among the powers of the earth, the separate and equal station to which the Laws of Nature and of Nature's God entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the separation.

Revised paragraph from The Declaration of Independence, SHA1 Hash

```
C2A4 D937 D275 521E 6CBF 6225 6918 4E5F 0E5B 96C1
```

Random Number Generators

- Many protocols rely on passing random information
- A “good” random number generator is a must
- Of course, no such thing as “random” in a computer
- Example:

$$x_{i+1} = 345x_i + 879 \pmod{8756}$$

Protocols

protocol (pro´ te-kôl´ , -kol´ , -kòl´) noun

1. The forms of ceremony and etiquette observed by diplomats and heads of state.
2. A code of correct conduct. See synonyms at etiquette.
3. Computer Science. A standard procedure for regulating data transmission between computers.

Protocols

protocol (pro´ te-kôl´ , -kol´ , -kòl´) noun

1. The forms of ceremony and etiquette observed by diplomats and heads of state.
2. A code of correct conduct. See synonyms at etiquette.
3. Computer Science. A standard procedure for regulating data transmission between computers.

The Cryptographic Protocol Players

- Alice — initiates a conversation.
- Bob — the person Alice wants to talk to.
- Eve — a passive network presence, an eavesdropper.
- Mallet — an active, malicious network presence.
- Trent — a trusted person or institution, e.g. a notary, lawyer, bank.

Mallet in the Middle

1. Mallet intercepts Alice's attempt to acquire Bob's public key.
2. Mallet substitutes his own public key.
3. Alice sends Bob a message, but she is using Mallet's public key.
4. Mallet intercepts the message, and decrypts with his private key.
5. Mallet re-encrypts the message using Bob's public key.
6. Mallet sends this re-encrypted message on to Bob.

Mallet in the Middle — Analysis and Solution

- Neither Bob nor Alice knows the message has been read by Mallet.

Mallet in the Middle — Analysis and Solution

- Neither Bob nor Alice knows the message has been read by Mallet.
- Use a trusted server to distribute public keys.

Mallet in the Middle — Analysis and Solution

- Neither Bob nor Alice knows the message has been read by Mallet.
- Use a trusted server to distribute public keys.
- Bob creates a hash of his public key, a “fingerprint”

Fingerprint of a PGP 2.6 1024-bit public key

```
A4 67 92 0B 50 C9 DC E0 7E A9 FE 69 4D 9A 41 7D
```

Mallet in the Middle — Analysis and Solution

- Neither Bob nor Alice knows the message has been read by Mallet.
- Use a trusted server to distribute public keys.
- Bob creates a hash of his public key, a "fingerprint"

Fingerprint of a PGP 2.6 1024-bit public key

```
A4 67 92 0B 50 C9 DC E0 7E A9 FE 69 4D 9A 41 7D
```

- Alice creates a hash of public key she receives.
- Alice and Bob compare hashes by conventional means.

Digital Signatures

Alice has a public key, P_A, and secret key, S_A
M is a message to be sent from Alice to Bob

$E(k, x)$ is an agreed-upon, public, encryption function
$D(k, x)$ is an agreed-upon, public, decryption function

1. Alice “decrypts” original message with her secret key $C = D(S_A, M)$.
2. Bob “encrypts” received message with Alice’s public key

$$E(P_A, C) = E(P_A, D(S_A, M)) = M$$

This requires that the encryption and decryption functions commute

Digital Signatures — Analysis

1. Only Alice knows her secret key, so only she can form C.
2. Anybody can perform Bob's verification step.
3. If the message is sabotaged in transit, then Bob's verification yields garbage.
4. In practice a message is hashed, and the hash is signed and appended to the message.
5. Individuals sign each other's public keys as testimony to their genuineness.
6. Authorities (Verisign, VISA, AOL) sign public keys to form "certificates."
7. A digital signature is frequently used to show something is genuine.

Secret Sharing

- Sometimes we might want others to possess a secret of ours, but without them knowing the secret itself.
- Alice has a secret for the success of her family business, and she would like her son, Bob, to have the secret when she dies — but she does not trust him with it while she is alive.
- She enlists the aid of Trent, the executor of her estate, whom she does not entirely trust either.

Secret Sharing

- Sometimes we might want others to possess a secret of ours, but without them knowing the secret itself.
- Alice has a secret for the success of her family business, and she would like her son, Bob, to have the secret when she dies — but she does not trust him with it while she is alive.
- She enlists the aid of Trent, the executor of her estate, whom she does not entirely trust either.

1. Alice has a secret S (a string of characters in binary).
2. Alice generates a random binary string, R, of the same length.
3. Alice does an exclusive-or (XOR, $\oplus$) of R and S to form

$$T = R \oplus S$$

4. Alice gives R to Bob and T to Trent.

Secret Sharing — Analysis

Example

S = 2 cups flour, 1 egg, 1 cup water, 3 tbs butter, chocolate chips

R = k31zhbkcwp1tix3dfb14corsg1pgfd2s1fcjajgdftbnomifoakqmibsy

T = lacpszo4h2uift3h2fgagonkqplkhkfjadgwyucil1yt4re4tms34cxi4k

- The strings R and T are useless to Bob and Trent individually.
- T appears just as random as R.
- When Alice dies Trent gives Bob T, and Bob exclusive-ors R and T,

$$R \oplus T = R \oplus (R \oplus S) = (R \oplus R) \oplus S = \mathbf{0} \oplus S = S$$

- Bob and Trent could together recover the secret prior to Alice's death.

Blind Notary Protocol

Sometimes we want to have a document signed without the signer being able to read the document. This is useful for digital cash and voting protocols. We can think of this as a notarization service being provided by a blind notary.

1. Alice multiplies her document by a number, the blinding factor.
2. Bob digitally signs the unintelligible, blinded document.
3. Alice divides the signed message by the blinding factor.
4. Alice obtains a version of her document, as if simply signed by Bob.

Blind Notary Analysis

- This requires the signature function and multiplication to commute with each other.

$$D(\alpha M, S_B) = \alpha D(M, S_B)$$

- The message could say, “Bob owes Alice $1,000,000.”
- Bob is only attesting to the fact that he signed the document (perhaps at a certain time).
- This is analogous to a document in an envelope made of the paper used in multi-part forms.
- “Stamper” is a free service on the Internet, adds a signature and a timestamp.

Who Do You Trust?

- Alice says her service can predict the price of Google's stock.
- Alice would like Bob to subscribe to her service.
- Bob asks for evidence of Alice's abilities before he subscribes.
- Alice offers her dead-on "prediction" from last month as evidence.
- Bob suspects Alice has created her "prediction" after the fact.
- Alice refuses to tell Bob her current prediction, because she does not want to give it to Bob for free.
- How can Alice make a prediction that she cannot adjust later, without simultaneously letting Bob have the use of that prediction?
- Trent could be employed to solve this dilemma.
  But Trent can be a difficult person to find.

Bit Commitment

1. Alice wishes to commit to a single bit, b.
2. Alice creates two lengthy random strings, R_1, R_2.
3. Alice builds the message $R_1 R_2\, b$.
4. Alice hashes this message to form H.
5. Alice “commits” by sending Bob R_1 and H.
6. Later, Alice sends Bob the message $M = R_1 R_2\, b$, “opening” her commitment.

Bit Commitment — Analysis

- Bob cannot determine b before Alice decides to disclose it.
- There are too many possibilities for R_2 as he tries to recreate possible values of M with test values of b.

Bit Commitment — Analysis

- Bob cannot determine b before Alice decides to disclose it.
- There are too many possibilities for R_2 as he tries to recreate possible values of M with test values of b.
- Bob can check the commitment after Alice reveals M.
- Bob checks that M contains R_1 as a substring and that M hashes to H correctly.

Bit Commitment — Analysis

- Bob cannot determine b before Alice decides to disclose it.
- There are too many possibilities for R_2 as he tries to recreate possible values of M with test values of b.
- Bob can check the commitment after Alice reveals M.
- Bob checks that M contains R_1 as a substring and that M hashes to H correctly.
- Alice cannot change her bit in the interim.
- Alice would have to find R_2' so that $R_1 R_2' b'$ also hashes to H.

Flipping Coins

How do we flip a coin over a network?

1. Alice chooses a bit at random.
2. Alice sends a message to Bob commiting to this bit.
3. Bob chooses a bit at random and sends it to Alice.
4. Alice sends a message to Bob opening her commitment.
5. Alice and Bob now know each other's choices for their bits.
6. If Alice's and Bob's bits match, the coin is heads.
7. If Alice's and Bob's bits differ, the coin is tails.

Flipping Coins — Analysis

- With bit commitment, Alice can "go first" and cannot renege.
- We can do this repeatedly to generate a random sequence of bits.
- Alice and Bob can build a key for symmetric encryption in this manner, with the confidence that neither of them is proposing a key with "unusual" properties.
- Applying public-key encryption to the messages, Eve cannot listen in.
- More involved protocols can be used to deal playing cards to a group over a network.

Anonymous Digital Cash

1. Alice prepares 100 money orders, each for $50.
2. She multiplies each by a different blinding factor.
3. She presents all 100 at the bank.
4. The bank has Alice unblind 99 of her 100 money orders.
5. Bank checks that all 99 are really for $50 each.
6. Satisfied, the bank digitally signs the remaining blinded money order with bank's secret key.
7. Bank deducts $50 from Alice's account.
8. Alice unblinds the money order and takes it to a merchant.
9. Merchant uses the bank's public key to check bank's signature.
10. Merchant accepts the money order, takes it to the bank.
11. The bank verifies its signature again.
12. Bank credits $50 to the merchant's account.

Anonymous Digital Cash — Analysis

- There is a 1 in 100 chance that Alice can get the bank to sign a \$1,000,000 money order without detection. There must be a large penalty to discourage this. (Jail time?)
- Alice never needs to reveal her identity to the merchant.
- Alice must identify herself to the bank when she gets the signed money order.
- However the bank cannot trace the spent money order back to Alice since it was blinded when Alice was at the bank.
- The transaction occurs off-line, without the bank's participation.
- The money order can be backed-up, moved on networks.
- Both Alice and the merchant can copy and spend the money again. The bank will not recognize the multiple uses, nor will they know who it was that recycled the money order.
- Protect against reuse, allow for divisibility and transferability?

Better Digital Cash I

1. Alice prepares 100 identical money orders. Each contains:
    1. The amount — \$50.
    2. A random uniqueness string, different for each money order.
    3. 100 copies of information identifying Alice (SSN, Name, Account, etc.) Each of these has been split via the secret sharing protocol in left and right "halves." Alice commits to each of the 200 halves. Note: There are 100 identification pairs on each of the 100 money orders.
2. Alice blinds each of the money orders individually.
3. The bank chooses 99 of Alice's prepared money orders. Alice unblinds each and opens all the commitments. The bank checks the amount, and the uniqueness string, and combines the two halves of each of the identifications.
4. Satisfied, the bank signs the remaining blinded money order and deducts \$50 from Alice's account.
5. Alice unblinds the money order and presents it to a merchant.

Better Digital Cash II

6. Merchant uses the bank's public key to check bank's signature.
7. Satisfied, the merchant gives Alice a 100 bit selector string. According to this string, Alice opens the commitment on *one* half of each of the 100 identifications.
8. The merchant takes the money order to the bank.
9. The bank checks its signature, and checks to see if the uniqueness string is in its database of spent money orders.
10. Satisfied, the bank credits $50 to the merchant's account and records the uniqueness string and the 100 open halves of the identifications in its database.

Better Digital Cash — Analysis I

- As before, Alice cannot hope to pass a fraudulent money order by the bank.
- Alice cannot tamper with the money order after she leaves the bank, since that would invalidate the bank's signature.
- The money can be spent or deposited twice, but the bank will eventually find out due to their database of spent uniqueness strings.
- Alice cannot spend the money twice, since the two merchants would be unlikely to provide the same selector string (1 in 2^{100}). Where the selectors differ, the bank will eventually get *both* halves of Alices identification. Combining them will provide Alice's identity.
- If the money order is deposited twice by the merchant, the selector strings used will be identical. The merchant cannot fake a second, different selector string since he cannot open Alice's commitments on her identification halves.

Better Digital Cash — Analysis II

- The merchant cannot identify Alice, since he only sees one half of each of her identifications.
- The bank cannot trace Alice's legitimate use of the money order since they cannot see the uniqueness string when Alice gets the money order signed. (It was blinded at the time.) Also, like the merchant, they only see halves of her identifications.

The Perfect Crime

1. Alice kidnaps a baby.
2. Alice prepares 1,000 money orders for $1,000 each.
3. Alice blinds the money orders and sends them to the parents.
4. Alice instructs the parents to have the bank sign all the money orders, and publish the signatures in the newspaper.
5. The parents comply — they pay for and publish the signatures.
6. Alice attaches the signatures to the money orders, unblinds them (in effect making the signatures look different than when they were published) and tries spending 50 of them.
7. If any of the 50 signatures are invalid, Alice keeps the baby and appeals to the parents again, asking for even more money. If all 50 of the signatures work, the baby is released unharmed at a quiet, random location.
8. Alice continues spending more money orders with complete anonymity.

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)
- No one can view somebody else's vote. (Secrecy)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)
- No one can view somebody else's vote. (Secrecy)
- No one can duplicate a vote. (Election worker fraud)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)
- No one can view somebody else's vote. (Secrecy)
- No one can duplicate a vote. (Election worker fraud)
- No one can change a vote. (Election worker fraud)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)
- No one can view somebody else's vote. (Secrecy)
- No one can duplicate a vote. (Election worker fraud)
- No one can change a vote. (Election worker fraud)
- Every voter can be assured their vote has been counted. (Auditing)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)
- No one can view somebody else's vote. (Secrecy)
- No one can duplicate a vote. (Election worker fraud)
- No one can change a vote. (Election worker fraud)
- Every voter can be assured their vote has been counted. (Auditing)
- Everybody knows who voted and who did not. (Legal requirement)

Secure Elections

Voting electronically has many requirements:

- Only registered voters can vote. (Authentication)
- At most one vote for each registered voter. (Voter fraud)
- No one can view somebody else's vote. (Secrecy)
- No one can duplicate a vote. (Election worker fraud)
- No one can change a vote. (Election worker fraud)
- Every voter can be assured their vote has been counted. (Auditing)
- Everybody knows who voted and who did not. (Legal requirement)

Find protocols that do not use a "centralized tabulating facility"?

Best implemented with open-source software?

Conclusions

- Cryptographic building blocks (encryption, hash functions, random number generators) use basic aspects of discrete mathematics.

Conclusions

- Cryptographic building blocks (encryption, hash functions, random number generators) use basic aspects of discrete mathematics.
- Elaborate combinations of these simple functions build up sophisticated protocols.

Conclusions

- Cryptographic building blocks (encryption, hash functions, random number generators) use basic aspects of discrete mathematics.
- Elaborate combinations of these simple functions build up sophisticated protocols.
- Encryption implies secrecy, cryptography promotes trust.

Conclusions

- Cryptographic building blocks (encryption, hash functions, random number generators) use basic aspects of discrete mathematics.
- Elaborate combinations of these simple functions build up sophisticated protocols.
- Encryption implies secrecy, cryptography promotes trust.
- Analyzing protocols means thinking like a cheat, liar, poser, and thief.

Conclusions

- Cryptographic building blocks (encryption, hash functions, random number generators) use basic aspects of discrete mathematics.
- Elaborate combinations of these simple functions build up sophisticated protocols.
- Encryption implies secrecy, cryptography promotes trust.
- Analyzing protocols means thinking like a cheat, liar, poser, and thief.

Further Reading

Applied Cryptography, Second Edition, Bruce Schneier, Wiley, 1996