Crypto and the Internet: Real-World Usage

Josh Benaloh
Senior Cryptographer
Microsoft Research

Internet Security?

- The Internet was *not* designed for security.
- Sending data via the Internet is like sending post cards through the mail …

  …when you don't trust the Post Office.

A Typical Internet Session

You (client) → Merchant (server)

⋮

I want to make a purchase. →

← What is your Credit Card Number?

My Credit Card is 6543 2345 6789 8765. →

Basic Encryption

Can we at least protect the credit card number so that it won't be revealed to anyone except the intended merchant?

Kerckhoffs's Principle (1883)

The security of a cryptosystem should depend only on the key.

You should assume that attackers know everything about your system *except* the key.

PINs, Passwords, & Keys

Informally …

- A *PIN* is a 4-6 digit speed bump.
- A *password* is a short, user-chosen, usually guessable selection from a small dictionary.
- A *key* is an unguessable, randomly chosen string – usually at least 128 bits.

Off-Line Attacks

- Don't even think about using user-chosen passwords as encryption keys.
- Don't even think about using keys derived deterministically from user-chosen passwords.
- Given the ciphertext, an attacker can do a (guided) exhaustive search through the space to find the password.

Symmetric Encryption

- If the client has a pre-existing relationship with the merchant, the two parties may have a shared secret key K – known only to these two.
- User encrypts private data with key K.
- Merchant decrypts data with key K.

Symmetric Ciphers

Private-key (symmetric) ciphers are usually divided into two classes.

- Stream ciphers
- Block ciphers

Symmetric Ciphers

Private-key (symmetric) ciphers are usually divided into two classes.

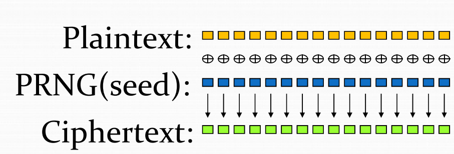
- Stream ciphers
- Block ciphers

Stream Ciphers

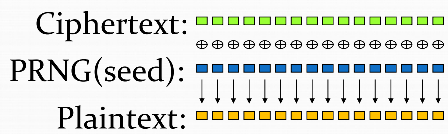
RC4, *A5/1*, *SEAL*, etc.

- Use the key as a seed to a pseudo-random number-generator.
- Take the stream of output bits from the PRNG and XOR it with the plaintext to form the ciphertext.

Stream Cipher Encryption



Stream Cipher Decryption



A PRNG: Alleged RC4

Initialization

S[0..255] = 0,1,…,255; j=0

K[0..255] = Key,Key,Key,…

for i = 0 to 255

  j = (j + S[i] + K[i]) mod 256

  swap S[i] and S[j]

A PRNG: Alleged RC4

Iteration

i = (i + 1) mod 256

j = (j + S[i]) mod 256

swap S[i] and S[j]

t = (S[i] + S[j]) mod 256

Output S[t]

Some Good Properties

- Stream ciphers are typically very fast.
- Stream ciphers can be very simple.
- The same function is used for encryption and decryption.

Stream Cipher Security

If two plaintexts are *ever* encrypted with the same stream cipher and key

$C_1 = K \oplus P_1$

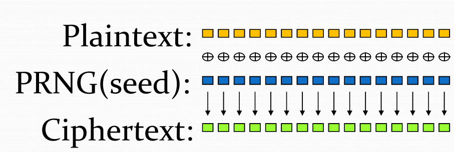
$C_2 = K \oplus P_2$

an attacker can easily compute

$C_1 \oplus C_2 = P_1 \oplus P_2$

from which P_1 and P_2 can usually be teased apart easily.

Stream Cipher Encryption



Stream Cipher Integrity

- It is easy for an adversary (even one who can't decrypt the ciphertext) to alter the plaintext in a known way.

Stream Cipher Integrity

- It is easy for an adversary (even one who can't decrypt the ciphertext) to alter the plaintext in a known way.

Bob to Bob's Bank:
Please transfer $0,000,002.00 to the account of my good friend Alice.

Stream Cipher Integrity

- It is easy for an adversary (even one who can't decrypt the ciphertext) to alter the plaintext in a known way.

Bob to Bob's Bank:
Please transfer $1,000,002.00 to the account of my good friend Alice.

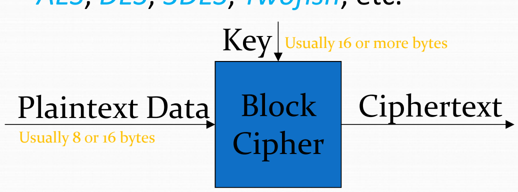
Symmetric Ciphers

Private-key (symmetric) ciphers are usually divided into two classes.

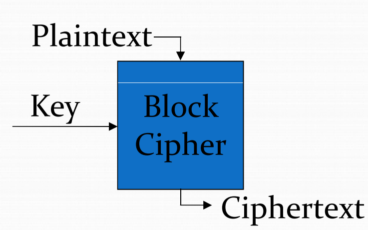
- Stream ciphers
- Block ciphers

Block Ciphers

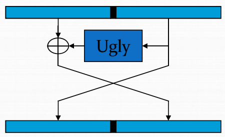
AES, *DES*, *3DES*, *Twofish*, etc.



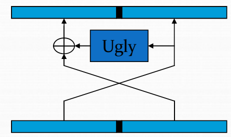
How to Build a Block Cipher



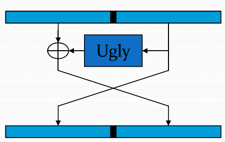
Feistel Ciphers



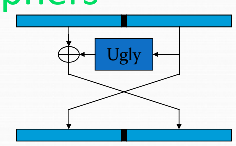
Feistel Ciphers



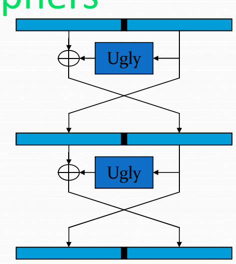
Feistel Ciphers



Feistel Ciphers



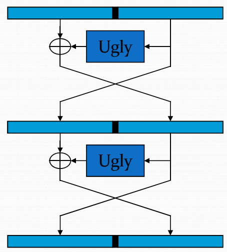
Feistel Ciphers



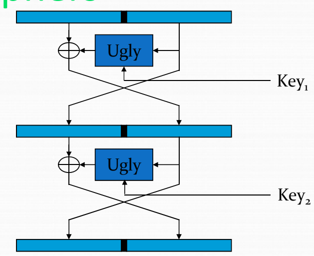
Feistel Ciphers

- Typically, Feistel ciphers are iterated for about 10-16 rounds.
- Different “sub-keys” are used for each round.
- Even a weak round function can yield a strong Feistel cipher if iterated sufficiently.

Feistel Ciphers



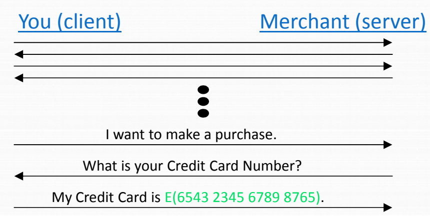
Feistel Ciphers



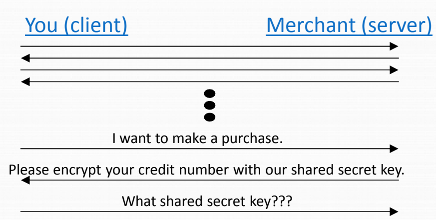
Feistel Ciphers

- Typically, Feistel ciphers are iterated for about 10-16 rounds.
- Different "sub-keys" are used for each round.
- Even a weak round function can yield a strong Feistel cipher if iterated sufficiently.

Transfer of Confidential Data



Transfer of Confidential Data



Asymmetric Encryption

- What if the user and merchant have no prior relationship?
- Asymmetric encryption allows someone to encrypt a message for a recipient without knowledge of the recipient's decryption key.

The Fundamental Equation

- $$Z = Y^X \bmod N$$

The Fundamental Equation

- $$Z = Y^X \bmod N$$

When Z is unknown, it can be efficiently computed.

The Fundamental Equation

- $$Z = Y^X \bmod N$$

When X is unknown, the problem is known as the *discrete logarithm* and is generally believed to be hard to solve.

The Fundamental Equation

- $$Z = Y^X \bmod N$$

When Y is unknown, the problem is known as *discrete root finding* and is generally believed to be hard to solve … without the factorization of N.

The Fundamental Equation

- $$Z = Y^X \bmod N$$

The problem is not well-studied for the case when N is unknown.

How to compute $Y^X \bmod N$

Compute Y^X and then reduce mod N.

- If X, Y, and N each are 1,000-bit integers, Y^X consists of ~2^{1010} bits.
- Since there are roughly 2^{250} particles in the universe, storage is a problem.

How to compute Y^X mod N

- Repeatedly multiplying by Y by itself X times (with a modulo N reduction after each multiplication) solves the storage problem.
- However, we would need to perform ~2^{900} 64-bit multiplications per second to complete the computation before the sun burns out.

March 5, 2013

How to compute Y^X mod N

Multiplication by Repeated Doubling

To compute $X \cdot Y$,

compute $Y, 2Y, 4Y, 8Y, 16Y, \ldots$

and sum up those values dictated by the binary representation of X.

Example: $26Y = 2Y + 8Y + 16Y$.

March 5, 2013

How to compute Y^X mod N

Exponentiation by Repeated Squaring

To compute Y^X,

compute $Y, Y^2, Y^4, Y^8, Y^{16}, \ldots$

and multiply those values dictated by the binary representation of X.

Example: $Y^{26} = Y^2 \cdot Y^8 \cdot Y^{16}$.

March 5, 2013

How to compute Y^X mod N

- We can now perform a 1,000-bit modular exponentiation using ~1,500 1,000-bit modular multiplications.
- 1,000 squarings: $Y, Y^2, Y^4, \ldots, Y^{2^{1000}}$
- ~500 "ordinary" multiplications

March 5, 2013

The Fundamental Equation

$$Z = Y^X \bmod N$$

When Y is unknown, the problem is known as *discrete root finding* and is generally believed to be hard to solve … without the factorization of N.

March 5, 2013

RSA Encryption/Decryption

- Select two large primes p and q.
- Publish the product $N = pq$.
- The exponent X is typically fixed at 65537.
- Encrypt message Y as $E(Y) = Y^X \bmod N$.
- Decrypt ciphertext Z as $D(Z) = Z^{1/X} \bmod N$.
- Note $D(E(Y)) = (Y^X)^{1/X} \bmod N = Y$.

March 5, 2013

RSA Signatures and Verification

- Not only is $D(E(Y)) = (Y^X)^{1/X} \bmod N = Y$, but also $E(D(Y)) = (Y^{1/X})^X \bmod N = Y$.
- To form a signature of message Y, create $S = D(Y) = Y^{1/X} \bmod N$.
- To verify the signature, check that $E(S) = S^X \bmod N$ matches Y.

March 5, 2013

Transfer of Confidential Data

You (client) — Merchant (server)

I want to make a purchase. →

← What is your Credit Card Number?

My Credit Card is 6543 2345 6789 8765. →

March 5, 2013

Transfer of Confidential Data

You (client) — Merchant (server)

I want to make a purchase. →

← Here is my RSA public key E.

My Credit Card is E(6543 2345 6789 8765). →

March 5, 2013

Intermediary Attack

You (client) — Intermediary — Merchant (server)

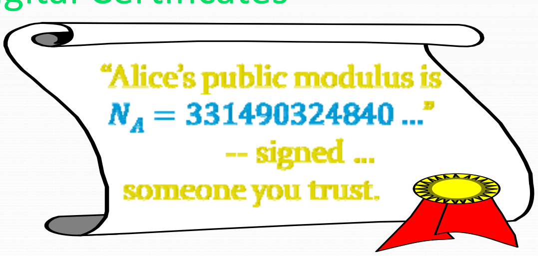
I want to make a purchase. → (You to Intermediary); I want to make a purchase. → (Intermediary to Merchant)

← My public key is Ẽ. (Intermediary to You); ← My public key is E. (Merchant to Intermediary)

Ẽ(CC#) → (You to Intermediary); E(CC#) → (Intermediary to Merchant)

March 5, 2013

Digital Certificates



March 5, 2013

Transfer of Confidential Data

You (client) — Merchant (server)

I want to make a purchase. →

← Here is my RSA public key E *and* a cert.

My Credit Card is E(6543 2345 6789 8765). →

March 5, 2013

Replay Attack

You (client) Merchant (server)

I want to make a purchase.

Here is my RSA public key E *and* a cert.

My Credit Card is E(6543 2345 6789 8765).

Later …

Eavesdropper Merchant (server)

I want to make a different purchase.

Here is my RSA public key E *and* a cert.

My Credit Card is E(6543 2345 6789 8765).

March 5, 2013

Transfer of Confidential Data

You (client) Merchant (server)

I want to make a purchase.

Here is my RSA public key E *and* a cert *and* a "nonce".

My Credit Card and your nonce are E("6543 2345 6789 8765", nonce).

March 5, 2013

SSL/PCT/TLS History

- 1994: Secure Sockets Layer (SSL) V2.0
- 1995: Private Communication Technology (PCT) V1.0
- 1996: Secure Sockets Layer (SSL) V3.0
- 1997: Private Communication Technology (PCT) V4.0
- 1999: Transport Layer Security (TLS) V1.0

March 5, 2013

SSL/PCT/TLS Handshake

You (client) Merchant (server)

Let's talk securely.
Here are the protocols and ciphers I understand.

I choose this protocol and these ciphers.
Here is my public key, a cert, a nonce, etc.

Using your public key, I've encrypted a random symmetric key (and your nonce).

March 5, 2013

SSL/PCT/TLS Agility

A principal reason for the success of SSL/TLS is its agility.

- The handshake negotiates symmetric and asymmetric ciphers, the hash function, and even the protocol's own version.
- This has allowed the protocol to survive and expand while many underlying primitives have been discredited or lost favor.

March 5, 2013

SSL/PCT/TLS Secure Channel

Once the negotiation is complete, *all* subsequent secure messages are sent

- encrypted – using the negotiated session key, and
- integrity checked with a keyed hash.

March 5, 2013

Hybrid Cryptography

- Asymmetric cryptography has many useful features not available in traditional symmetric cryptography.
- Symmetric cryptography is *much* more efficient than asymmetric.
- The practical hybrid is formed by using asymmetric cryptography to establish a secure channel and symmetric cryptography within the secure channel.

March 5, 2013

Application: Verifiable Elections

- Current election technology requires trust in
  - the officials who manage elections,
  - the equipment and its manufacturers, and
  - the processes used in the election.
- Cryptography allows us to eliminate this trust.

March 5, 2013

A Verifiable Election

Voter Name	Vote
Alice Smith	Jefferson
Bob Williams	Adams
Carol James	Adams
David Fuentes	Jefferson
Ellen Chu	Jefferson

Totals	
Jefferson	3
Adams	2

A Verifiable Election

Voter Name	Vote	
Alice Smith	[redacted]	X37BM6YPM
Bob Williams	[redacted]	2J8CNF2KQ
Carol James	[redacted]	VRSF5JQWZ
David Fuentes	[redacted]	MW5B2VA7Y
Ellen Chu	[redacted]	8VPPS2L39

Totals	
Jefferson	3
Adams	2

A Verifiable Election

Voter Name	Vote	
Alice Smith	[redacted]	X37BM6YPM
Bob Williams	[redacted]	2J8CNF2KQ
Carol James	Adams	VRSF5JQWZ
David Fuentes	[redacted]	MW5B2VA7Y
Ellen Chu	[redacted]	8VPPS2L39

Totals	
Jefferson	3
Adams	2

A Verifiable Election

Voter Name	Vote	
Alice Smith	[redacted]	X37BM6YPM
Bob Williams	[redacted]	2J8CNF2KQ
Carol James	[redacted]	VRSF5JQWZ
David Fuentes	[redacted]	MW5B2VA7Y
Ellen Chu	[redacted]	8VPPS2L39

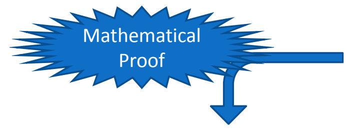
Totals	
Jefferson	3
Adams	2

A Verifiable Election

X37BM6YPM
2J8CNF2KQ
VRSF5JQWZ
MW5B2VA7Y
8VPPS2L39

Totals	
Jefferson	3
Adams	2

A Verifiable Election



X37BM6YPM
2J8CNF2KQ
VRSF5JQWZ
MW5B2VA7Y
8VPPS2L39

Totals	
Jefferson	3
Adams	2

The Voter's Perspective

Systems that produce verifiable elections can be built to look exactly like current systems …

- paper-based
- fully-electronic
- in-person
- remote

… with one addition …

A Verifiable Receipt

The Voter's Perspective

Voters can …

- Use receipts to check their results are properly recorded on a public web site.
- Throw their receipts in the trash.
- Write and use their own election verifiers
- Download applications from sources of their choice to verify the mathematical proof of the tally.
- Believe verifications done by their political parties, LWV, ACLU, etc.
- Accept the results without question.

Some systems producing verifiable elections …

Prêt à Voter Ballot

Bob	
Eve	
Carol	
Alice	
David	
	17320508

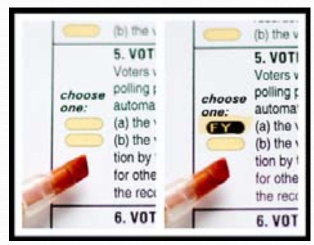
Prêt à Voter Ballot

Bob	
Eve	
Carol	
Alice	X
David	
	17320508

Prêt à Voter Ballot

X
17320508

Scantegrity



VeriScan

Helios

Helios

STAR-Vote

- Voters use electronic ballot marking devices to indicate their preferences.
- When a voter's selections are completed, the device provides the voter with a paper ballot summary and an encrypted receipt. It also records the encrypted ballot.
- The voter can review the paper ballot summary, and optionally deposit it in a ballot box.
- All encrypted ballots are posted, but the only votes counted are those for which a corresponding paper ballot has been deposited. The remaining ballots are decrypted.

Benefits of E2E-Verifiability

- Strong public assurance of election integrity
- Elimination of trust requirements
- Certification relief

Questions???

March 5, 2013