

Defending Computer Networks

Lecture 22: More Encryption

Stuart Staniford

Adjunct Professor of Computer Science


Logistics

- HW3 regrade complete
- HW4 should be out tomorrow
- Guest lecture Tuesday

Chinese hack U.S. weather systems, satellite network

By **Mary Pat Flaherty**, **Jason Samenow** and **Lisa Rein** November 12 at 12:20 PM 

 Follow @marypatflaherty

 Follow @capitalweather

 Follow @Reinlwapo

Hackers from China breached the federal weather network recently, forcing cybersecurity teams to [seal off data](#) vital to disaster planning, aviation, shipping and scores of other crucial uses, officials said.

The intrusion occurred in late September but officials gave no indication that they had a problem until Oct. 20, said three people familiar with the hack and the subsequent reaction by the National Oceanic and Atmospheric Administration, which includes the National Weather Service. Even then, NOAA did not say its systems were compromised.

Officials also said that the agency did not notify the proper authorities when it learned of the attack.

http://www.washingtonpost.com/local/chinese-hack-us-weather-systems-satellite-network/2014/11/12/bef1206a-68e9-11e4-b053-65cea7903f2e_story.html

Banks take on retailers over who foots cyber attacks bill

Gina Chon and Barney Jopson in Washington



Banks are gearing up for a big fight with retailers over who covers the cost of [cyber attacks](#), after they paid most of the bill for breaches that they blamed on retailers' own security deficiencies.

In a rare show of unity, industry bodies that represent banks are banding together to urge lawmakers to introduce legislation that would force retailers to pay for the clean-up themselves during the new session of Congress next year.



More

IN BANKS

[Virgin Money float values bank at £1.25bn](#)

[John Plender No end in sight to a rotten culture](#)

[Rigging scandal costs six banks \\$4.3bn](#)

[Monte Paschi 'ploughs ahead' despite losses](#)

[Sign up now](#)

The tussle between the two sectors comes as cyber attacks become an increasingly common problem for companies and highlights the difficulties in deciding who is responsible for the costs. Cyber attacks on retailers [Home Depot](#) and [Target](#), for example, affected almost 100m credit cards in the past year.

“This is an equity argument,” said [Cam Fine](#), head of the Independent Community Bankers of America, which has about 5,000 members. “If it was Home Depot’s data security system that was breached, shouldn’t they have to reimburse banks for all of the costs since it wasn’t the banks’ fault? That’s just common sense.”

Main Focus of Today

- AES (Advanced Encryption Standard)
- Public/Private Key Cryptography

AES

- AES = Advanced Encryption Standard
- Published by US NIST (2001)
 - Following an open competition
 - Including public crypto-analysis attempts
- Replaced earlier (1977) DES standard
- Widely used in practice
 - Fairly fast
- Currently believed secure
 - Some slight uncertainty following Snowden
 - https://www.schneier.com/blog/archives/2012/03/can_the_nsa_bre.html

AES Design

- Block oriented cipher
 - Takes blocks of plaintext 128 bits at a time
 - 16 bytes
 - Other sizes must be padded
- Key sizes of 128, 192, and 256 bits
- What's known as a substitution-permutation network
- Repeated numerous rounds
- Will sketch how it works while skipping worst of the math

AES Rounds

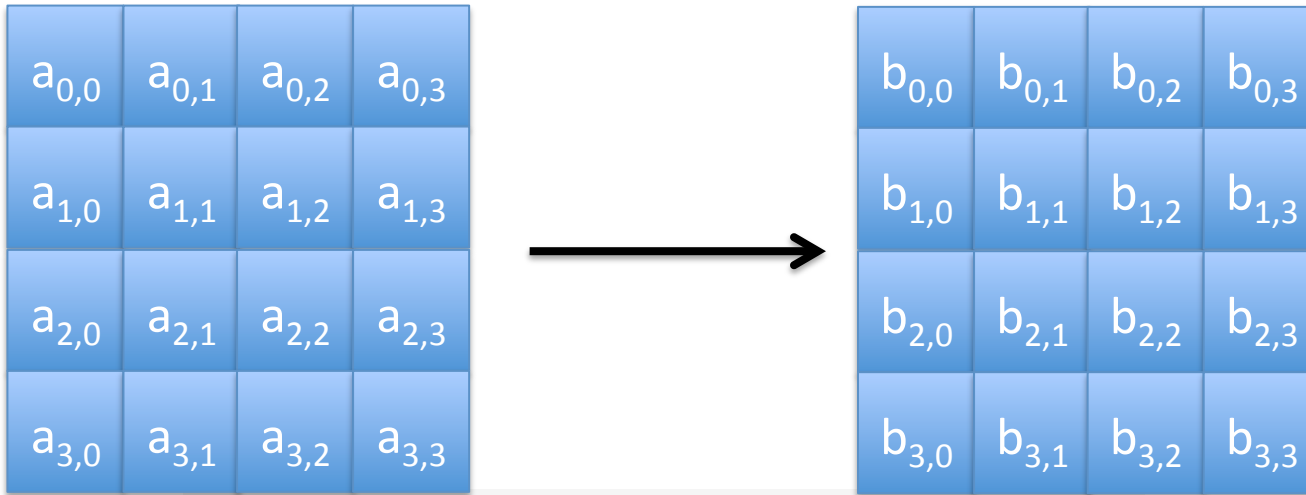
- The basic cipher operations are performed repeatedly on each block
 - 128 bit keys, 10 rounds
 - 192 bit keys, 12 rounds
 - 256 bit keys, 14 rounds
- Each round uses a different key
 - Derived from the master key using a set of complex algebraic operations
 - Depending on the algebra of finite (Galois) fields.

AES State Matrix

- Each 16 byte block is arranged in a 4x4 state matrix
- Used to keep track of what block will turn into

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

First Step: S-box (substitution)



	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Shift Rows Step

No change

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

One left

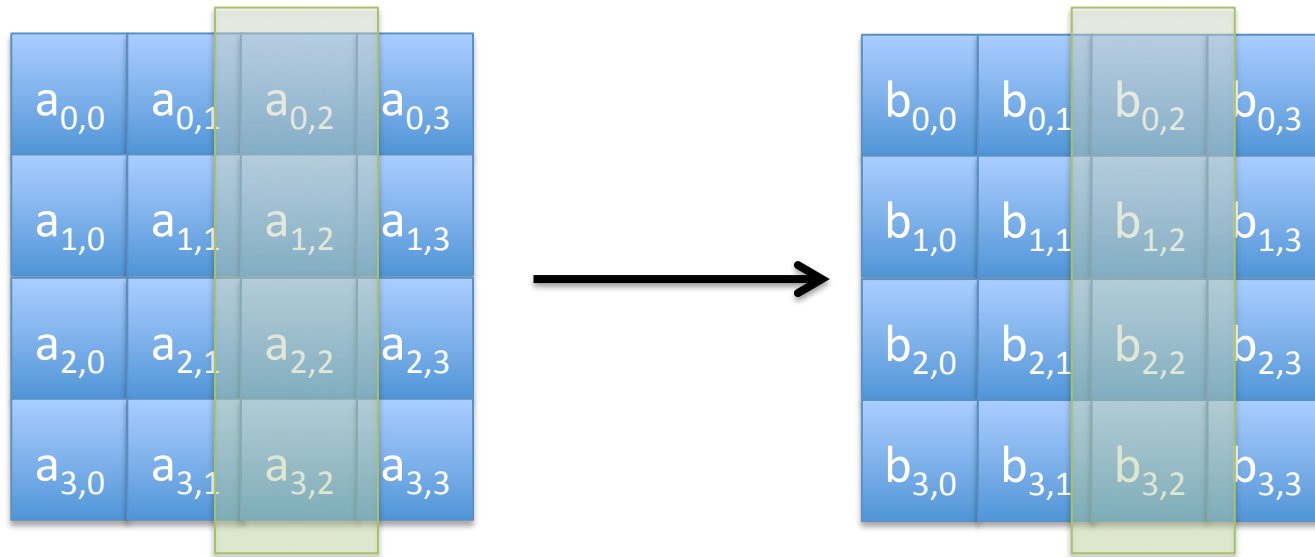
Two left

Three left



$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,0}$
$a_{2,2}$	$a_{2,3}$	$a_{2,0}$	$a_{2,1}$
$a_{3,3}$	$a_{3,0}$	$a_{3,1}$	$a_{3,2}$

Mix Columns Step



Each column is transformed by a complex (but fixed) algebraic equation that takes input from all four bytes and incorporates them into each output byte

Mix Columns Detail

Multiply block by this matrix:

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$

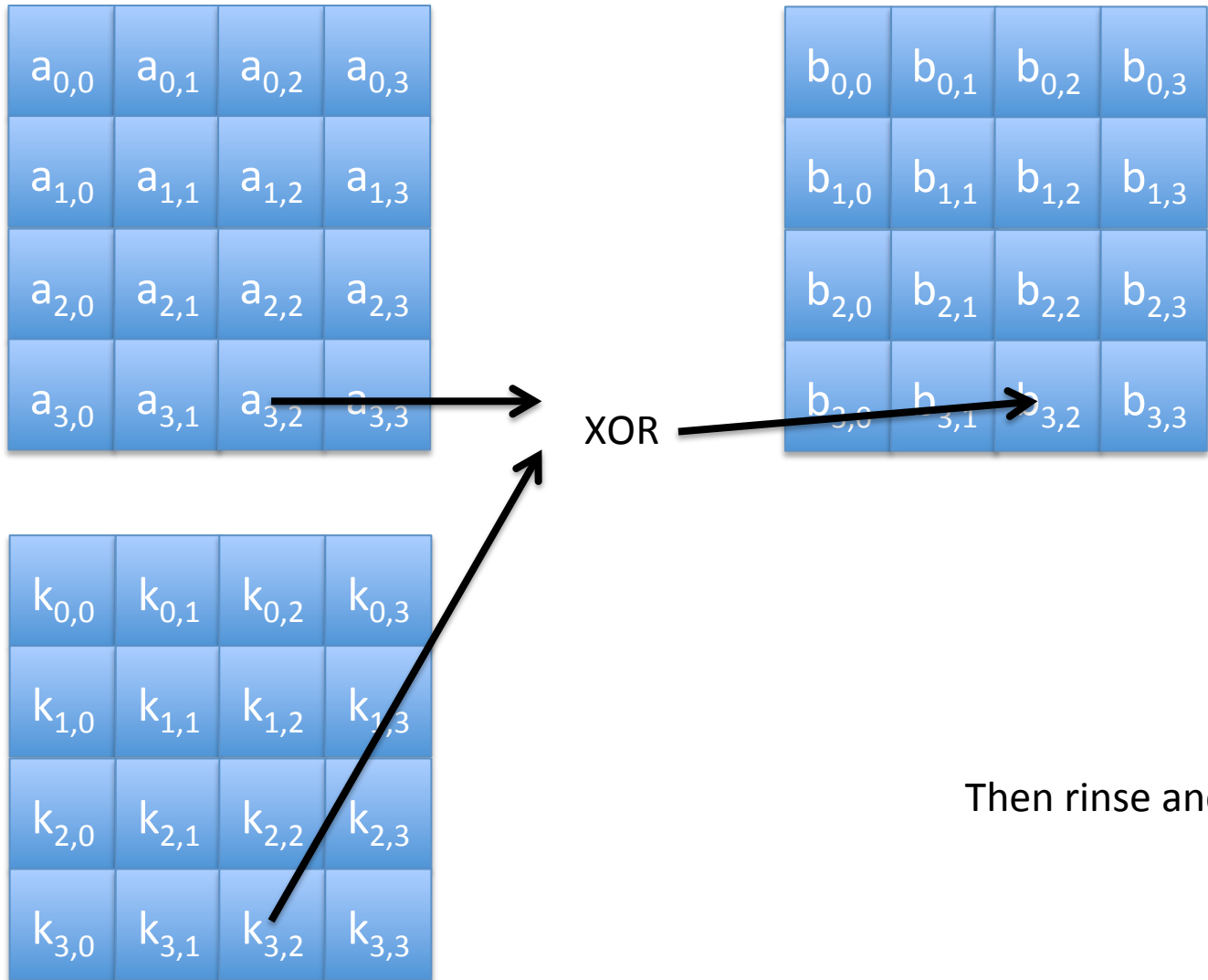
$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$

where

multiplication by 1 means no change, multiplication by 2 means shifting to the left, and multiplication by 3 means shifting to the left and then performing XOR with the initial unshifted value. After shifting, a conditional XOR with 0x1B should be performed if the shifted value is larger than 0xFF. Addition is simply XOR.

http://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Add Round Key



Then rinse and repeat...

Public Key Cryptography

- Also known as Asymmetric Cryptography
 - To distinguish from symmetric cryptography
 - shared secret
- Instead of shared key, keys come in pairs
 - Public key used to encrypt data
 - Private key used to decrypt data
 - Not feasible to infer private key from public key

Main Advantage

- Enormously simplifies key management
 - Imagine large group that need to communicate by shared key schemes
 - Share one key amongst many and risk losing everything
 - Or keep track of n^2 keys
- With public key crypto, I give out my public key, everyone can know it.
 - Anyone can send me a message
 - Only I can read those messages
 - I only have to worry about one secret key (mine)
 - Don't have to share my secret (private) key with anyone

History

- Invented in the seventies
- 1973, secretly, by GCHQ (British intel)
- 1976 Diffie Hellmann Key exchange
- 1977 RSA public/private key exchange
 - Rivest, Shamir, Adleman
- Many more schemes since then
- RSA still in wide practical use
 - 1024 bit keys considered weak, but longer ok

RSA

- Underlying base is difficulty of factoring very large numbers
- Will sketch algorithm while again skipping worst of math details
- We choose two large primes p, q
 - hundreds of digits each
 - Modulus, $n = pq$
 - Size of n in bits is the key length
 - Then choose an exponent, e that
 - Has no common factors with $(p-1)(q-1)$
- Public key is n and e
- Private key can be computed from p & q

Encryption

- Take the text and turn blocks of text into numbers. Say M is number for some block
- Then take $M^e \bmod n$
- Toy example
 - <http://www.woodmann.com/crackz/Tutorials/Rsa.htm>
 - $p = 43$, $q = 59$ (both prime)
 - $n = 43 * 59 = 2537$
 - $e = 13$ (no common factor with 42 or 58)

Encryption Example

- Message is ST OP
- Encode as 1819 1415
 - $1819^{13} \bmod 2537 = 2081$
 - $1415^{13} \bmod 2537 = 2182$
 - Ciphertext is 2081 2182

Decryption

- Find decryption exponent d such that
- $de \bmod (p-1)(q-1) = 1$
- d is the multiplicative inverse of e ,
 - modulo $(p-1)(q-1)$
 - Tractable algorithms for this are known
- Then plaintext P can be computed from C
 - $P = C^d \bmod n$

Decryption Example

- Suppose ciphertext is 0981 0461
- $d = 937$ for our previous example
- $937 * 13 \bmod 2436 = 12181 \bmod 2436 = 1$
- $0981^{937} \bmod 2537 = 0704$
- $0461^{937} \bmod 2537 = 1115$
- Message was HE LP

Applicability of Public Key

- Typically public key algorithms are computationally expensive.
- Not practical to apply to long messages
- Therefore generally used in the process of establishing a temporary symmetric key
 - Session key
 - Encrypted by public key crypto for transfer
 - Then used to encrypt lengthy communication session
 - Then thrown away

Let's Try It

- `openssl genrsa -out private_key.pem 1024`
- `more private_key.pem`
- `openssl rsa -in private_key.pem -text -noout`
- `openssl rsa -pubout -in private_key.pem -out public_key.pem`
- `more public_key.pem`