Epayment security ecom 6016 electronic payment systems

Law, Security

ePayment Security ECOM 6016 Electronic Payment Systems - Keep financial data secret from unauthorized parties (privacy) — CRYPTOGRAPHY Lecture 3 ePayment Security - Verify that messages have not been altered in transit (integrity) — HASH FUNCTIONS - Prove that a party engaged in a transaction ((nonrepudiation)) — DIGITAL SIGNATURES - Verify identity of users (authentication) — PASSWORDS, DIGITAL CERTIFICATES THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Cryptography and Hash Functions yp g p y - Message digest (hash) algorithms — Secure Hash Algorithm: SHA-1, SHA-2, SHA-3 competition — Securing passwords Hash Functions - A “ hash" is a short function of a message, f ti f sometimes called a “ message digest" g g - BUT: a hash is not uniquely reversible - Many messages have the same hash Hash function H produces a fixed size hash of a message M, usually 128"512 bits h = H(M) - S Symmetric encryption ti ti — DES and variations — AES: Rijndael - Public-key algorithms — RSA - Defending against attacks — Salting, nonces g - Digital signatures THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS One-Way Hash Functions - For any string s, H(s), the hash of s, is of fixed length (shorter than) (h t th s) - Hashes should be easy to compute - A “ one-way" has is computationally difficult to invert: can’t find any message corresponding to a given hash This is a message M This is a message M that we want to make unalterable so it cannot be forged or modified. One-Way Hash Functions - There are plenty of hash functions but no obvious one-way h h f hash functions ti - Good one-way hashes have the diffusion

property: Altering any bit of the message changes many bits of the hash - This prevents trying similar messages to see if they hash to the same thing We ll non reversibility - We'll see how non-reversibility provides security h = H(M) H 52f21cf7c7034a20 17a21e17e061a863 This is the hash of message M M: THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Uses of One Way Hash Functions One-Way - - - - Password verification Message authentication (message digests) Prevention of replay attack Digital signatures Key-Hashed Message Authentication Codes (HMACs) Shared Key Original Plaintext Hashing with MD5, SHA, etc. HMAC Key-Hashed Message Authentication Code (HMAC) Appended to Plaintext Before Transmission HMAC Original Plaintext Note: No encryption; only hashing THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Key-Hashed Message Authentication Codes (HMACs) Receiver Repeats the HMAC Computation On the Received Plaintext Shared Key Received Original Plaintext Nonce to Prevent Replay Attack p y - Replay attack: repeating the messages in a challenge-response protocol (lik username/ h ll t l (like / password) to gain access to a system - Defense: make the messages different EVERY TIME the protocol is used. - But how? The username and password don't change don t - Answer: use a random number, called a " nonce" each time. Require the user to include the nonce in his response - NOTE: Nonce is an obsolete word: " for the nonce" means " for the time being, " " just for now" THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Hashing with same algorithm ith Computed HMAC

ïƒœ COMPARE ïƒž Received HMAC If computed and received HMACs are the same, The sender must know the key and so is authenticated AND the message has not been altered THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Password Verification System sends nonce to user: Secure Hash Algorithm SHA-512 - US Federal Information Processing Standard, but used around the world - Uses exclusive-OR operation ïƒ… A= 0011011110001 B= 1101001101011 Aïƒ…B= 1110010011010 nonce = 992883774 System looks up password pp Password store Iam#4VKU User concatenates nonce to password: Iam#4VKU 992883774 p|| nonce p|| nonce Iam#4VKU 992883774 H H(p|| nonce) 779dsfe55d2884e0ea5 e3a011fa3211b Allow Login Yes Deny Login No Exact Match? H H(p|| nonce) 779dsfe55d2884e0ea5 e3a011fa3211b - Exclusive-OR is lossy; knowing A ïƒ… B does not reveal even one bit of either A or B - Regular OR: If a bit of A ïƒš B is zero, then both corresponding bits of both A and B were zero User sends H(p|| nonce) over network THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Information Hiding with Exclusive-OR - x ïƒ… y = 1 if either x or y is 1 but not both: y xïƒ…y 0 0 1 1 1 0 Secure Hash Algorithm SHA-512 g x 0 1 - If x ïƒ… y = 1 we can’t tell which one is a 1 - Can’t trace backwards to determine values Can t - If x ïƒ™ y = 1 then BOTH x and y are 1 THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Secure Hash Algorithm Flow LONG MESSAGE TO BE HASHED SHA-512 Block Function TAKE FIRST 32 WORDS (1024 BITS) REPEAT FOR EACH 1024-BIT BLOCK STARTING HASH EIGHT 64-BIT 64 BIT WORDS (512 BITS) EXPAND TO 80 WORDS (2560 BITS) REPEAT 79 MORE TIMES ...

FINAL HASH (512 BITS) 111011 010101 110100 010011 011101 001011 010001 001011 011001 110101 000100 110001 011101 101011 110001 111011 Ch(e, f, g) = (e AND f) XOR (NOT e AND g) Maj(a, b, c) = (a AND b) XOR (a AND c) XOR (b AND c) âˆ‘(a) = ROTR(a, 28) XOR ROTR(a, 34) XOR ROTR(a, 39) âˆ‘(e) = ROTR(e, 14) XOR ROTR(e, 18) XOR ROTR(e, 41) + = addition modulo 2^64 Kt = a 64"bit additive constant for round t Wt = a 64"bit word derived from the current 512"bit input block for round t THE UNIVERSITY OF HONG KONG FEB/MAR 2011 © 2011 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS History of SHA - We’re now on the third generation of SHA: SHA-0 (1993-1995) (weakness f SHA 0 (1993 1995) (k found early) d l) SHA-1 (1995-2005) SHA-2 (2005 - ??) - SHA-512 is part of SHA-2 - SHA 1 is weak but not yet fully cracked, still the most SHA-1 cracked widely used hash algorithm SHA-3 - RIGHT NOW there is a competition for SHA 3 — Began in 2007 — There are five finalists: BLAKE, GrÃ¸stl, JH, Keccak, Skien — Winner to be announced in 2012 Hashing V. S. Encryption Hashing V. S. Encryption Hello, world. A sample sentence to show encryption. k E NhbXBsZSBzZW50ZW5jZS B0byBzaG93IEVuY3J5cHR pb24KsZSBzZ k D Hello, world.
A sample sentence to show encryption. ï¿½ NhbXBsZSBzZW50ZW5jZS B0byBzaG93IEVuY3J5cHR p pb24KsZSBzZ Encryption is two way, and requires a key to encrypt/decrypt This is a clear text you can easily read g y without using the key. The sentence is longer than the text above. h 52f21cf7c7034a20 7a e 7e06 a863 17a21e17e061a863 — Hashing is one way There is no 'de hashing’ Hashing is one"way. There is no de"hashing THE UNIVERSITY OF HONG

KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Cryptography yp g p y MESSAGE SPACE (ALL POSSIBLE PLAINTEXT MESSAGES) “ TRANSFER $5000 TO MY SAVINGS ACCOUNT" Cryptography MESSAGE SPACE ((ALL POSSIBLE PLAINTEXT MESSAGES) “ TRANSFER TRANSFER $5000 TO MY SAVINGS ACCOUNT" ENCRYPTION IS SECURE IF ONLY AUTHORIZED PEOPLE KNOW HOW TO REVERSE IT CODE SPACE (ALL POSSIBLE ENCRYPTED MESSAGES) CODE SPACE (ALL POSSIBLE ENCRYPTED MESSAGES) - - - - - MUST BE REVERSIBLE (BUT ONLY IF YOU KNOW THE SECRET) - - - - - “ 1822UX S4HHG7 803TG 0J71D2 MK8A36 18PN1" - - - - - ENCRYPTION IS ONE-TO-ONE AND REVERSIBLE EVERY CODE CORRESPONDS TO EXACTLY ONE MESSAGE - - - - - “ 1822UX S4HHG7 803TG 0J71D2 MK8A36 18PN1" FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG The Encryption Process MATERIAL WE WANT TO KEEP SECRET Role of the Key in Cryptography - The key is a parameter to an encryption procedure - Procedure stays the same, but produces different results based on a given key S P E C I A L T Y B D F G H J K M N O Q R U V W X Z A B C D E F G H I J K L M N O P Q R S T U V W X Y Z C O N S U L T I N G EXAMPLE: OBJECT: HIDE A MESSAGE (PLAINTEXT) BY MAKING IT UNREADABLE (CIPHERTEXT) UNREADABLE VERSION OF PLAINTEXT MIGHT BE: TEXT DATA GRAPHICS AUDIO VIDEO SPREADSHEET ... MATHEMATICAL SCRAMBLING PROCEDURE DATA TO THE ENCRYPTION ALGORITHM (TELLS HOW TO SCRAMBLE THIS PARTICULAR MESSAGE) D S R A V G H E R M SOURCE: STEIN, WEB SECURITY FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS NOTE: THIS METHOD IS NOT USED IN ANY REAL CRYPTOGRAPHY SYSTEM. IT IS AN EXAMPLE INTENDED

ONLY TO ILLUSTRATE THE USE OF KEYS. THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG Symmetric Encryption SAME KEY USED FOR BOTH ENCRYPTION AND DECRYPTION Advanced Encryption Standard (AES) - Based on a method called Rijndael, invented by j , y Vincent Rijmen and Joan Daeman (both male), who won a cryptography competition - Replaced Data Encryption Standard (DES) in 2001, but DES is still widely used - Symmetric block cipher with block length 128 bits, key lengths 128/192/256 bits - V Very fast: PC implementations at 3GB per second f t i l t ti t d SENDER AND RECIPIENT MUST BOTH KNOW THE KEY THIS IS A WEAKNESS SOURCE: STEIN, WEB SECURITY THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS AES Overview Input message: 4x4 matrix Transformations in Each AES Round Symmetric key Output from Round n-1 SubByte: substitutes bytes of the 4 x 4 matrix ShiftRows: shifts rows of the 4 x 4 matrix MixColumn: replace bytes in each column by different functions of the whole column AddRoundKey: XOR round key with the 4 x 4 matrix 128-bit blocks Round n: Number of rounds based on key length 128-bit, 10 rounds 192 bit, 192-bit, 12 rounds 256-bit, 14 rounds SubByte ShiftRows MixColumn Round key Each round key is different, obtained from full symmetric key AddRoundKey Encrypted output: Input to Round n+1 R d 1 THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS SubByte Input: ea 04 65 85 83 45 5d 96 5c 33 98 b0 f0 2d as c5 16 x 16 matrix specifies byte substitutions: ShiftRows Input: Output: 87 f2 4d 97 87 f2 4d 97 Output: 87 f2 4d 97 ec 6e 4c 90 4a c3 46 e7 8c d8 95 a6 S-Box 6e 4c 90 ec 46 e7 4a

c3 a6 8c d8 95 ec 6e 4c 90 4a c3 46 e7 8c d8 95 a6 SOURCE: WILLIAM STALLINGS THE UNIVERSITY OF HONG KONG SOURCE: WILLIAM STALLINGS FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS MixColumn Add Round Key Final output for this round: Input: 87 f2 4d 97 Output: 47 40 a3 4c 37 d4 70 9f 94 e4 3a 42 ed a5 a6 bc SOURCE: WILLIAM STALLINGS SOURCE: WILLIAM STALLINGS 6e 4c 90 ec 46 e7 4a c3 a6 8c d8 95 The 4 x 4 matrix is XORed with the round key THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS AES Round Summary Input bytes: A Rijndael Animation by Enrique Zabala Transformations: Output bytes: O b ANIMATION SOURCE: WILLIAM STALLINGS 32 Cipher Block Chaining Example - - DES is an older, less secure symmetric encryption algorithm; uses 56-bit keys 56 bit In ECB mode, the same input text always produces the same output. This creates risk of partial decryption. PLAINTEXT BLOCK 1 PLAINTEXT BLOCK 2 Triple DES - - Security can be increased by encrypting multiple times with different keys Double D bl DES i not much more secure th single DES b is t h than i l because of a “ meet-in-the-middle" attack K1 K2 K3 INITIALIZATION STRING ïƒ… DES ïƒ… DES etc. PLAINTEXT BLOCK 1 DES ENCRYPT DES DECRYPT DES ENCRYPT CIPHERTEXT BLOCK 1 CIPHERTEXT BLOCK 1 CIPHERTEXT BLOCK 2 - - - This method is called 3DES-IK, for “ independent keys" q g y Equivalent to a single 112-bit key If K1 = K2 = K3 this is just single DES THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Public Key Public-Key (Asymmetric) Encryption 2. SENDERS USE SITE’S PUBLIC KEY FOR ENCRYPTION 3. SITE USES ITS PRIVATE KEY FOR DECRYPTION Public-Key

Encryption y yp 2. Bob looks up Alice’s public key 5. Alice uses her PRIVATE KEY to decrypt M 1. Bob wants to send M to Alice M 1. 1 USERS WANT TO SEND PLAINTEXT TO RECIPIENT WEBSITE 4. ONLY WEBSITE CAN DECRYPT THE CIPHERTEXT. NO ONE ELSE KNOWS HOW 4. Bob transmits the encrypted message in the clear M SOURCE: STEIN, WEB SECURITY STEIN THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS 3. Bob uses Alice’s public key to encrypt M SOURCE: CHIN-TSER HUANG 6. Alice now has M. No one else does 09/13/2011 36 Public-Key Encryption - - - - When Alice gets M no one else could have read it M, No one else has Alice’s PRIVATE key Problem: she can’t be sure Bob sent it can t Anyone with Alice’s PUBLIC key could have sent it Public Key Public-Key Authentication 2. Bob encrypts M with his PRIVATE key 4. Alice looks up B b’ Bob’s public key 1. Bob wants to send M to Alice so she is sure Bob sent it 5. Alice decrypts M with Bob’s PUBLIC key M 3. Bob sends the encrypted message to Alice M THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Public-Key Authentication - When Alice gets M she is sure it came from Bob M, - No one but Bob has Bob’s PRIVATE key - Problem: anyone can read M — all that is needed is Bob’s PRIVATE key - Is there some way to achieve security AND authentication at the same time? Secure Authenticated Messages Use two public-private key pairs — one for Bob, one for Alice M M Alice’s Public Key PUA Alice’s Private Key PRA Bob’s Private Key PRB Bob’s Public Key PUB Keys in key pairs are mathematically linked THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS One-Way Trapdoor Functions - A function that is easy to compute … - But

computationally difficult to invert without knowing the secret (the “ trapdoor") trapdoor) - Example: f (x, y) = x-y - Given f (x y), it is difficult to find either x or y (x, y) - Given f (x, y) and x (the secret), it is easy to find y - Any one way trapdoor function can be used in public one-way publickey cryptography. Rivest-Shamir-Adelman Rivest Shamir Adelman (RSA) - It is easy to multiply two numbers but apparently hard y py pp y to factor a number into a product of two others. y - Given p, q, it is easy to compute n = p - q - Example: p = 5453089; q = 3918067 - Easy to find n = 21365568058963 y - Given n, it is hard to find two numbers p, q with p - q = n - Now suppose n = 7859112349338149 What are p and q such that p - q = n ? - Multiplication is a one-way function - RSA exploits this fact in public-key encryption THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Rivest-Shamir-Adelman (RSA) - Each user generates a public/private key pair: - Select two large primes at random: p q (1024 bits) p, - Compute their product n = p - q — note: Ï†(n) = number of divisors of n = (p-1)(q-1) - Select a small odd number e that does not divide Ï†(n) - Find the multiplicative inverse of e, that is, a number (Ï†( )) such that e - d = 1 (mod Ï†(n)) - Public encryption key is the pair (e. n) - Private decryption key is the pair (d, n) - Knowing (e, n) is of no help in finding d. Still need p q g and q, which involves factoring n, which is difficult THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS RSA Encryption - The message M is an integer - To encrypt message M using key (e, n): - Compute C(M) = M e (mod n) p ( ) ( ) - To decrypt message C using key (d, n): - Compute P(C) = C d (mod n) - N t th t P(C(M)) = C(P(M)) = (M e)d (d n) Note

that (mod) e-d = M (mod n) = M Because e - d = 1 ((mod n)) - DEMO THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS RSA Example p = 61; q = 53 n = pq = 3233 (modulus, can be given to others) e = 17 (public exponent, can be given to others) d = 2753 (private exponent kept secret!) exponent, PUBLIC KEY = (3233, 17) PRIVATE KEY = (3233, 2753) To encrypt 123, compute 12317 (mod 3233) = 337587917446653715596592958817679803 mod 3233 = 855 37 digits INVERSE OF 5 IS 3 MULTIPLICATION MOD 7 Multiplicative Inverses p Over Finite Fields - - - 1 1 The i Th inverse e-1 of a number e satisfies e-1 - e = 1 f b ti fi The inverse of 5 is 1/5 If we only allow numbers from 0 to n-1 (mod n), then for special n1 n) values of n, each e has a unique inverse 0 0 1 2 3 4 5 6 0 0 0 0 0 0 0 1 0 1 2 3 4 5 6 2 0 2 4 6 1 3 5 3 0 3 6 2 5 1 4 4 0 4 1 5 2 6 3 5 0 5 3 1 6 4 2 6 0 6 5 4 3 2 1 6 - 2 = 12 WHEN DIVIDED BY 7 GIVES REMAINDER 5 To decrypt 855 compute 8552753 (mod 3233) = 123 855, (intermediate value has 8072 digits) SOURCE: FRANCIS LITTERIO THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS EACH ROW EXCEPT THE ZERO ROW HAS EXACTLY ONE 1 EACH ELEMENT HAS A UNIQUE INVERSE THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Trapdoor Functions for Cryptography - ANY one-way trapdoor function f(x) can be used for y p ( ) public-key cryptography - Alice wants to send message m to Bob - Bob’s public key e is a parameter to the trapdoor function fe(x) (the inverse fe -1(x) is easy to compute knowing B b’ private k d b t diffi lt without d) k i Bob’s i t key but difficult ith t - Alice computes fe(m), sends it to Bob 1 - Bob computes fe -1(fe(m)) = m (easy if d is known) - Eavesdropper Eve can’t compute m = fe -1(fe(m)) 1 without th t d ith t the

trapdoor d t find th i to fi d the inverse fe -1 Discrete Logarithms - If ab = c, we say that logac = b y g - Example: 232 = 4294927296 so log2(4294927296) = 32 p g g y - Computing ab and logac are both easy for real numbers - In a finite field, it is easy to calculate c = ab mod p but given c, a and p it i very diffi lt t find b i d is difficult to fi d - This is the “ discrete logarithm" problem - Analogy: Given x it is easy to find two real numbers y, z such that x = y - z - Given an integer n it is hard to find two integers p, q such that n = p - q THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Diffie-Hellman Key Exchange y g - Object: allow Alice and Bob to exchange a secret key - Protocol has two public parameters: a prime p and a number g < p such that given 0 < n < p there is some k such that gk = n (g is called a generator) g) - Alice and Bob generate random private values a, b between 1 and p-2 - Alice’s public value is ga (mod p); Bob’s is gb (mod p) - Alice and Bob share their public values - Alice computes (gb)a (mod p) = gba (mod p) - Bob computes (ga)b (mod p) = gab = gba (mod p) - Let key = gab. Now both Alice and Bob have it. - No one else can compute it — they don’t know a or b THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Security Attacks y - A LOT of money is protected by cryptography - H k Hackers are constantly t i to defeat it t tl trying t d f t — — — — — Brute force (try all keys) Mathematical attack (find weaknesses in the algorithm) Social engineering (get people to reveal their key) Man-in-the-middle (intercept communications) Side channel attacks THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Side Channel Attacks - “ Side channel": any observable information

emitted by the physical implementation of the cryptosystem - Timing (see when certain operations performed) - C h contents (Cache t t (see which memory l hi h locations are ti accessed) - Electromagnetic radiation (monitor RF emissions) - Power consumption (trace the power used by a chip) - Physical chip structure (for hard wired keys) hard-wired Cache Observation - AES uses large tables (4 x 1024 bytes) for efficiency - One encryption accesses only a small portion of the tables, which is a function of the data and the encryption key THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS Power Consumption p - Some bit operations consume more electric power than others Major Ideas j - Secure hash algorithms create message digests - E Encryption algorithms are complex ti l ith l — must be studied carefully (by cryptographers) — subject to sophisticated attacks bj t t hi ti t d tt k - Symmetric encryption is fast - AES is the new standard symmetric encryption algorithm - Nonces defend against replay attacks - RSA is the principal public-key encryption algorithm Public key - Public-key encryption is slow because of the need to work with huge numbers (~2000 bits) THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS SOURCE: BERTONI ET AL. THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS El Gamal Encryption - Based on the discrete logarithm g - Bob’s public key is (p, q, r) - Bob’s private key is s such that r = qs mod p & THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS - Alice sends Bob the message m by picking a random secret number k and sending (a, (a b) = (qk mod p mrk mod p) p, - Bob computes b (as)-1 mod p = mrk (qks)-1 = mqks

(qks)-1 = m - (Bob knows s; nobody else can do this) THE UNIVERSITY OF HONG KONG FEB/MAR 2012 © 2012 MICHAEL I. SHAMOS