บทที่ 16

ความปลอดภัยในระบบคอมพิวเตอร์

Computer Security

วัตถุประสงค์

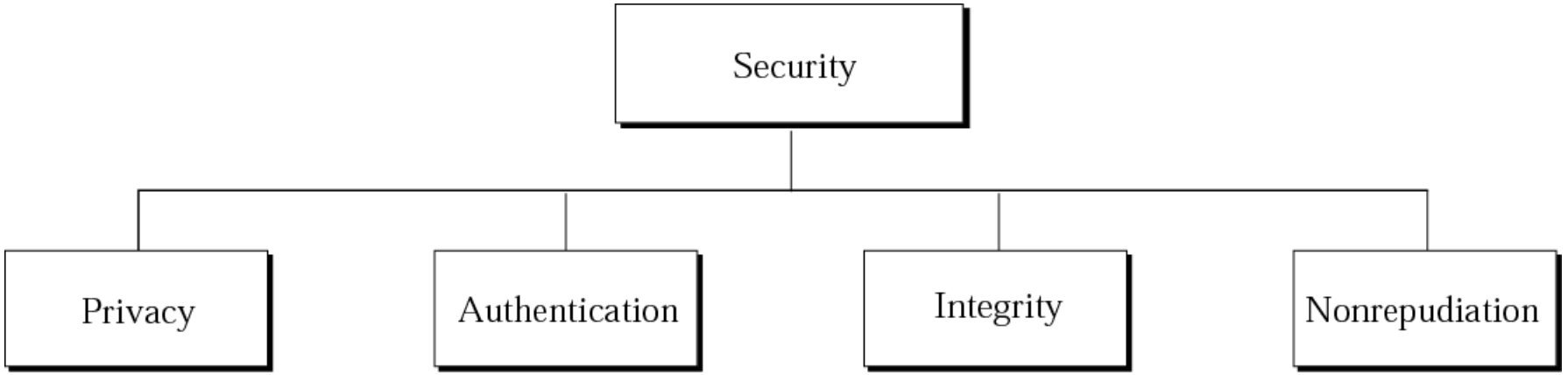
หลังจากเรียนจบบทที่ 16 แล้ว นักศึกษาต้องสามารถ:

- อธิบาย 4 รูปแบบของความปลอดภัยในระบบเครือข่ายคอมพิวเตอร์ได้: privacy, authentication, integrity, และ nonrepudiation
- อธิบายวิธีการที่จะทำให้ระบบเครือข่ายบรรลุความปลอดภัยทั้ง 4 รูปแบบโดยใช้กระบวนการเข้ารหัส (encryption) และ การถอดรหัส (decryption)
- อธิบายความแตกต่างระหว่างการเข้ารหัสโดยใช้ secret-key และ public-key
- เข้าใจวิธีการใช้ digital signature เพื่อก่อให้เกิด privacy, integrity, และ nonrepudiation

ความปลอดภัยมีความสำคัญอย่างยิ่งในระบบคอมพิวเตอร์ในปัจจุบัน ยิ่งเทคโนโลยีอินเทอร์เน็ตแพร่หลายมากขึ้นเท่าไร การแลกเปลี่ยนข้อมูลข่าวสารก็ยิ่งมีมากขึ้น ข้อมูลข่าวสารเหล่านี้แหละที่ต้องการความปลอดภัย ตัวอย่างเช่นถ้าท่านต้องการสั่งซื้อสินค้าผ่านอินเทอร์เน็ต ท่านก็ต้องการให้ข้อมูลที่ท่านส่งให้กับผู้ขาย เช่นเลขที่บัตรเครดิต เป็นความลับและใช้โดยผู้ขายสินค้าเท่านั้น ในขณะเดียวกัน เมื่อท่านรับข้อมูลข่าวสาร ท่านก็ต้องการทราบและตรวจสอบว่าใครเป็นคนส่ง เป็นต้น

ในบทนี้จะศึกษาความรู้เบื้องต้นของวิธีการรักษาความปลอดภัยในระบบคอมพิวเตอร์ เรื่องนี้เป็นเรื่องใหญ่และสำคัญ มีตำราหลายเล่มที่อธิบายเกี่ยวกับเรื่องนี้โดยเฉพาะ สิ่งที่เรียนจะเป็นเครื่องกระตุ้นให้เห็นความสำคัญที่จะศึกษาต่อไป ลักษณะของความปลอดภัยมี 4 ชนิดคือ:

1. Privacy 2. authentication 3. integrity 4. nonrepudiation ดังรูป 16.1



รูปที่ 16-1 ประเภทของความปลอดภัย

1. Privacy (Confidentiality):ในระบบการสื่อสารข้อมูลที่มีความปลอดภัยนั้น ทั้งผู้ส่ง (sender) และผู้รับ (receiver) ต่างก็คาดหวังความเป็นส่วนตัว (privacy) หรือความลับ (confidentiality) เป็นอย่างยิ่ง นั่นคือเฉพาะผู้ส่งและผู้รับเท่านั้นที่เข้าใจสาระของข้อมูลข่าวสาร (message) ที่ส่ง

2. Authentication: ความลับของข้อมูลที่ส่งไม่เพียงพอสำหรับระบบการสื่อสารข้อมูล ระบบจะต้องระบุเอกลักษณ์ของผู้ส่งเพื่อให้ผู้รับสามารถตรวจสอบและรับรู้ด้วย

3. Integrity: ความปลอดภัยและการระบุตัวตนเป็นเพียงปัจจัย 2 ประการที่จำเป็นในระบบการสื่อสาร ความกลมกลืนกัน (integrity) ของข้อมูลก็จะต้องรักษาไว้ ทั้งผู้ส่งและผู้รับคงไม่พอใจที่ข้อมูลข่าวสารที่ส่งมีการเปลี่ยนแปลงระหว่างการส่ง เช่น ถ้าท่านต้องการโอนเงินจำนวน10,000 บาทจากบัญชีของท่านไปให้เพื่อนยืม แต่จำนวนเงินเมื่อถึงปลายทางเปลี่ยนเป็น 100 บาท อย่างนี้คงเสียหายทั้งผู้ส่งและผู้รับ ความผิดพลาดอาจเกิดขึ้นจากการพิมพ์ที่ผิดพลาด เกิดจากผู้บุกรุก (intruder) ผู้ได้รับประโยชน์จากการเปลี่ยนแปลง หรือไม่ก็เกิดจากอุบัติเหตุจากความผิดพลาดของฮาร์ดแวร์หรือซอฟท์แวร์ก็เป็นได้

4. **Nonrepudiation**: เป็นการป้องกันไม่ให้มีการปฏิเสธจากผู้ส่งในข้อมูลข่าวสารที่เขาเป็นผู้ส่ง เช่น ถ้าท่านส่งภาพที่ผิดกฎหมายผ่านเครือข่ายไปให้ผู้อื่นอันก่อให้เกิดความเสียหาย ระบบจะต้องมีหลักฐานผูกมัดว่าท่านเป็นผู้ส่งที่จะปฏิเสธไม่ได้ นั่นคือระบบจะต้องมีหลักฐานพิสูจน์ให้ชัดเจนว่าท่านเป็นผู้ส่ง

อีกตัวอย่างหนึ่ง สมมติว่าท่านส่งข้อความไปยังธนาคารเพื่อที่จะโอนเงินจากบัญชีของท่านไปยังบัญชีอื่น ทางธนาคารจะต้องพิสูจน์ทราบให้ได้ว่าผู้ที่ส่งคำขอมานั้นเป็นเจ้าของบัญชีจริง ในการโอนเงินที่ท่านต้องไปธนาคารเอง **ลายเซ็น**ของท่านคือ**เครื่องพิสูจน์ตน**ว่าเป็นเจ้าของบัญชีตัวจริง ส่วน**การโอนเงินทางอินเทอร์เน็ต** ระบบอาจใช้ **user name** กับ **password** เป็นเครื่องมือพิสูจน์ทราบ

รายละเอียดของแต่ละลักษณะมีดังนี้

16.1

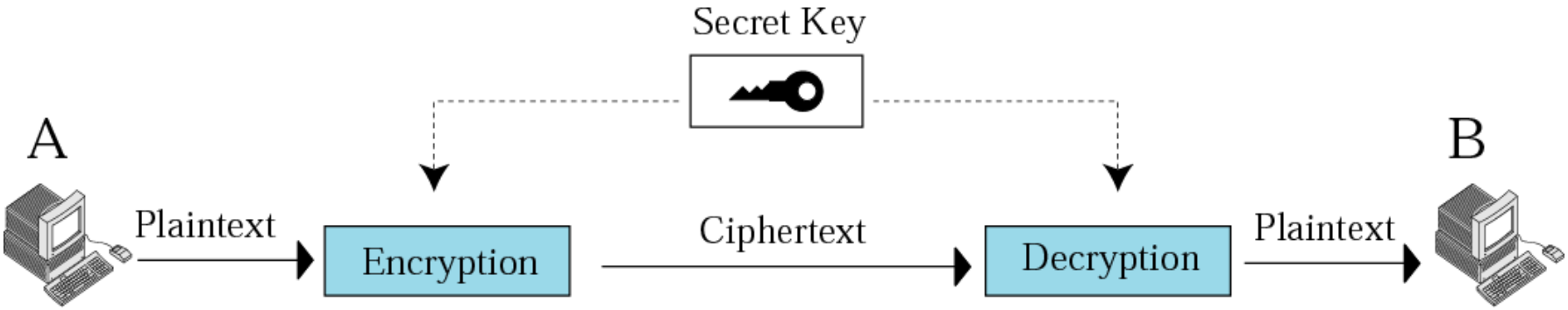
ความเป็นส่วนตัว

PRIVACY

16.1.1 Privacy: (ความเป็นส่วนตัว) กำหนดว่าข้อมูลที่ส่งจะต้องถูกเข้ารหัส (encrypt) ณ จุดที่ผู้ส่งทำการส่งข้อมูล และถูกถอดรหัส (decrypt) ณ จุดที่ผู้รับปลายทาง ซึ่งการทำเช่นนี้จะทำให้ผู้บุกรุกที่แอบลักข้อมูลระหว่างทางการส่ง ไม่อาจเข้าใจได้โดยง่าย

(1) Encrypt/Decrypt: ปัจจุบันการดำเนินการเกี่ยวกับ privacy กระทำด้วยวิธีการเข้ารหัสและการถอดรหัส ข้อมูลจะถูกเข้ารหัสที่ผู้ส่งและถูกถอดรหัสที่ผู้รับ วิธีการเข้ารหัสและถอดรหัสมี 2 แบบคือแบบที่ใช้กุญแจลับ (secret key) กับ แบบที่ใช้กุญแจสาธารณะ (public key)

(1.1) Secret key: เป็นวิธีที่ง่ายที่สุด โดยผู้ส่งใช้กุญแจลับกับอัลกอริธึมการเข้ารหัสทำการเข้ารหัสข้อมูลที่จะส่ง และเมื่อถึงปลายทาง ผู้รับจะใช้กุญแจลับตัวเดิมกับอัลกอริธึมการถอดรหัสทำการถอดรหัสข้อมูลเพื่อให้ข้อมูลกลับมาอยู่ในรูปเดิม ดังรูปที่ 16.2 ข้อมูลเดิมก่อนที่จะส่งเรียกว่า “plaintext” เมื่อเข้ารหัสแล้วจะเรียกว่า “ciphertext” ในที่นี้ ทั้ง A และ B ใช้ secret key ตัวเดียวกัน และ อัลกอริธึมเข้ารหัสกับอัลกอริธึมถอดรหัสจะเป็น inverse ซึ่งกันและกันเช่น ถ้าอัลกอริธึมเข้ารหัสทำการบวกเลข 5 เข้ากับข้อมูล อัลกอริธึมถอดรหัสจะเอา 5 ไปลบออกจากข้อมูลเป็นต้น



รูปที่ 16-2 การเข้ารหัสโดยใช้ Secret key

In secret key encryption, the same key is used in encryption and decryption. However, the encryption and decryption algorithms are the inverse of each other.

Data Encryption Standard (DES): การเข้ารหัสโดยใช้กุญแจลับมีใช้กันมากว่า 2,000 ปี ในระยะแรก อัลกอริธึมไม่ยุ่งยาก กุญแจลับก็ไม่ยากที่จะเดา แต่ปัจจุบันอัลกอริธึมมีความยุ่งยากและสลับซับซ้อนมาก อัลกอริธึมที่นิยมใช้กันทั่วไปในปัจจุบันมีชื่อว่า "data encryption standard" (DES)

DES ทำการเข้ารหัสและถอดรหัสในระดับบิต มีขั้นตอนการทำงานดังนี้

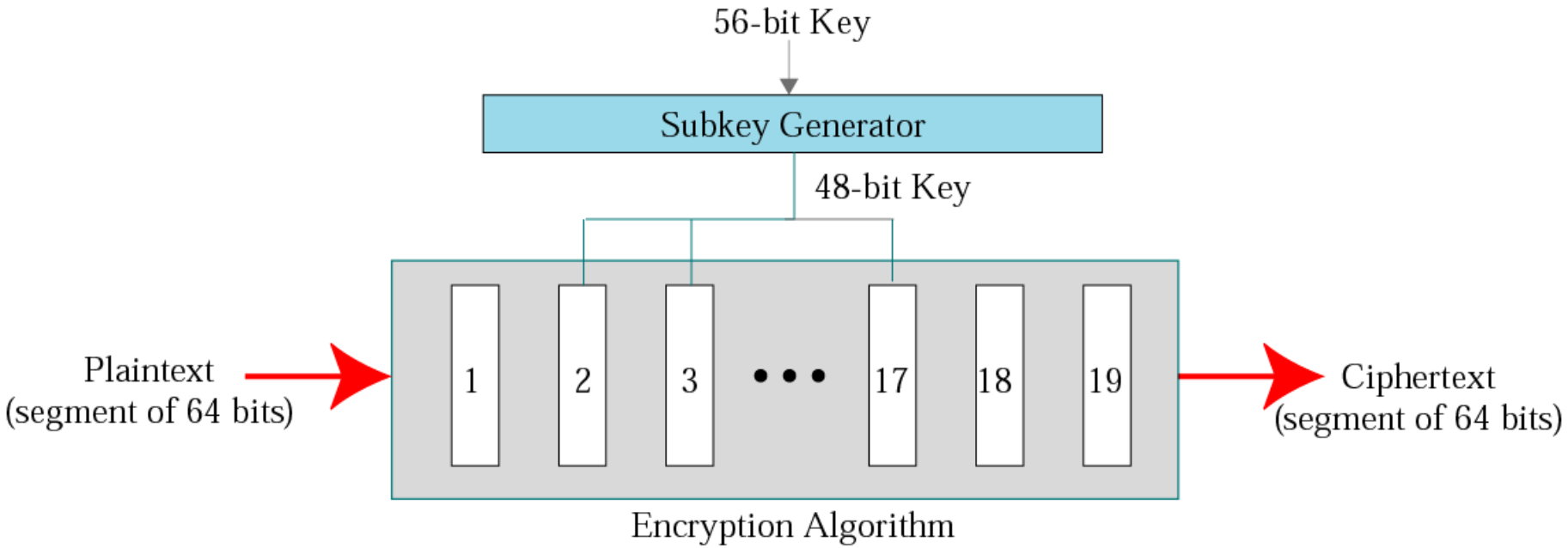
ขั้นที่ 1: แปลงข้อมูลที่จะส่งเป็นสตริงของบิต

ขั้นที่ 2: แบ่งสตริงของบิตออกเป็นส่วนๆ แต่ละส่วนมีขนาด 64 บิต ถ้าส่วนใดไม่เต็ม 64 บิตก็ให้เติม 0 เข้าไปข้างหน้าให้ครบ 64 บิต

ขั้นที่ 3: เข้ารหัสแต่ละส่วนโดยใช้กุญแจที่มีขนาด 56-บิต (ที่จริง กุญแจมีขนาด 64 บิต อีก 8 บิตเป็นส่วนควบคุมความผิดพลาด)

ดังรูปที่ 16.3

แนวคิดคือทำการ scramble ข้อมูลและกุญแจลับเข้าด้วยกันโดยให้ทุกๆบิตของ ciphertext ขึ้นอยู่กับแต่ละบิตของ plaintext และกุญแจลับ การทำดังกล่าวจะทำให้ผู้ไม่ประสงค์ดีคาดเดาบิตของ plaintext จากบิตของ ciphertext ได้ยากยิ่งขึ้น



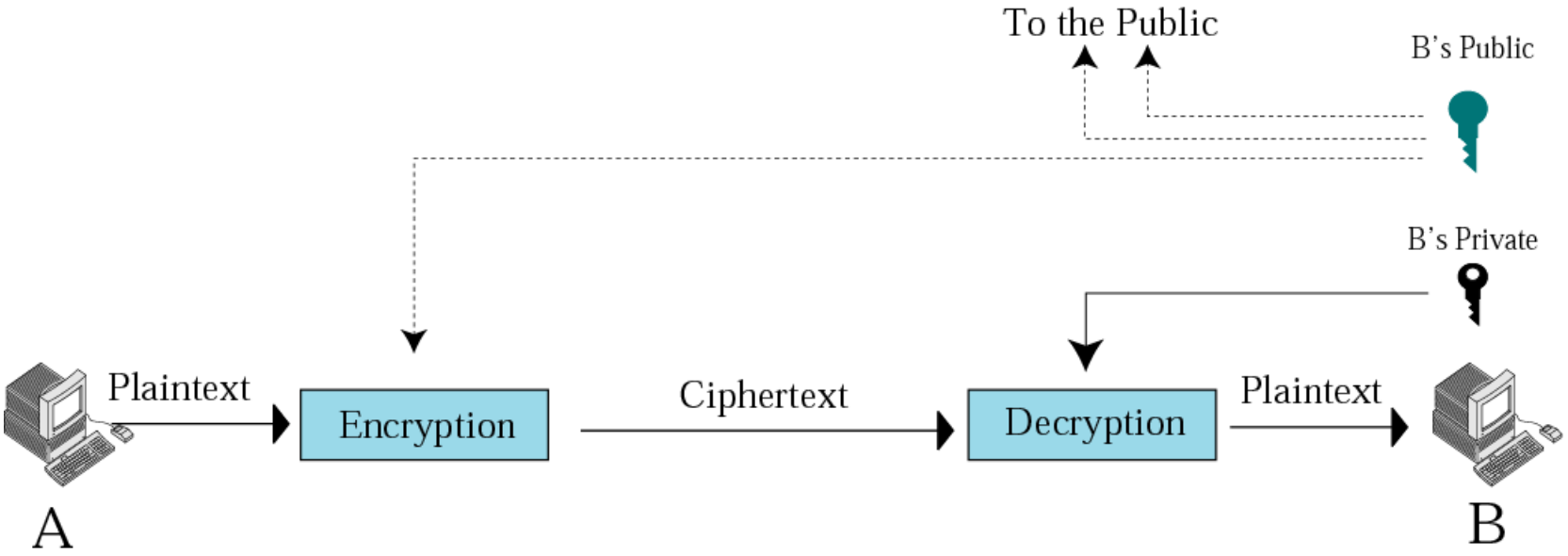
รูปที่ 16-3 DES

จากรูปที่ 16.3 อัลกอริธึมแบ่งการเข้ารหัสออกเป็น 19 ระยะ คือระยะที่ 1, 18, และ 19 ทั้ง 3 ระยะนี้อัลกอริธึมทำการสลับที่ (permutation) บิตต่างๆในส่วนย่อยโดยไม่ได้ใช้กุญแจ ส่วนระยะที่ 2-17 ทำเหมือนกันทั้งหมดคือข้อมูล 32 บิตทางขวาของระยะที่ 2 สลับไปเป็นข้อมูล 32 บิตทางซ้ายของระยะที่ 3 ข้อมูลส่วนนี้จะถูก scramble กับกุญแจ ผลลัพธ์ที่ได้จะนำไปเป็น 32 บิตทางขวาของระยะต่อไป รายละเอียดเกินกว่าที่จะอธิบาย ณ ที่นี้ได้

ข้อดีและจุดอ่อน: การเข้ารหัสโดยใช้ secret key มีข้อดีด้านเวลาคือมีความเร็วสูงในการเข้ารหัสและถอดรหัสเมื่อเปรียบเทียบกับวิธี public key ที่จะอธิบายต่อไป อัลกอริธึมนี้เหมาะกับข้อมูลที่มีขนาดใหญ่ๆ อย่างไรก็ดี การเข้ารหัสโดยใช้ secret key ก็มีจุดอ่อน 2 ประเด็นหลักคือการที่แต่ละคู่ของผู้รับ-ผู้ส่งต้องมี secret key ที่ไม่ซ้ำกัน นั้น ถ้ามีผู้คน N คนสื่อสารกัน จะต้องใช้ secret key ที่ไม่ซ้ำกันมีจำนวนถึง N(N-1)/2 ตัว ลองคิดดูว่าถ้า N = 1,000,000 คน เราจะต้องใช้ secret key จำนวนเท่าใด? นอกจากนี้ การกระจาย (distribution) ของ keys ระหว่างผู้รับกับผู้ส่งยังเป็นเรื่องที่ยากที่จะไม่ให้เหมือนกัน

(1.2) Public key: วิธีที่ 2 ของ privacy คือ การเข้ารหัสโดยใช้ public key วิธีนี้ใช้ key 2 ตัวคือ private key กับ public key โดยผู้รับข้อมูลจะเก็บ private key ไว้ ส่วน public key จะประกาศให้สาธารณะทราบ (อาจผ่านทางอินเทอร์เน็ต)

จากรูปที่ 16.4 ถ้าผู้ส่ง A ต้องการส่งข้อมูล M ไปยังผู้รับ B
ขั้นที่ 1: A ใช้ public key เพื่อทำการเข้ารหัสข้อมูล M
ขั้นที่ 2: A ทำการส่งข้อมูล M ที่ scramble แล้วผ่านเครือข่าย
ขั้นที่ 3: เมื่อ B ได้รับข้อมูลที่ scramble เขาจะใช้ private key ที่เขาเก็บไว้ทำการถอดรหัส
ขั้นที่ 4: B ได้ข้อมูล M ที่ต้องการ



รูปที่ 16-4 การเข้ารหัสโดยใช้ Public key

แนวคิดเบื้องหลังการใช้ Public key ในการเข้ารหัสและถอดรหัสข้อมูลคือการกระทำทั้งสองไม่เป็น inverse ของกันและกัน แม้ว่าผู้บุรุกจะทราบทั้ง public key อัลกอริธึมในการเข้ารหัสและถอดรหัส แต่เขาก็จะไม่สามารถถอดรหัสได้เพราะเขาไม่ทราบ private key

การเข้ารหัสโดยใช้อัลกอริธึม Rivest-Shamir-Adleman (RSA): อัลกอริธึมที่นิยมใช้มากที่สุดสำหรับวิธีที่ใช้ public key คือ RSA ในอัลกอริธึมนี้ private key ประกอบด้วยคู่ลำดับ (N,d) ส่วน public key ประกอบด้วยคู่ลำดับ (N,e) เมื่อ N เป็นตัวร่วมระหว่าง key ทั้งสอง เมื่อผู้ส่งต้องการส่งข้อมูลเขาจะใช้อัลกอริธึมต่อไปนี้เพื่อเข้ารหัส:

$$C = P^e \bmod N$$

เมื่อ P คือ plaintext ที่แทนด้วยตัวเลข และ C คือตัวเลขที่แทน Ciphertext ส่วนผู้รับจะใช้อัลกอริธึมต่อไปนี้ทำการถอดรหัส:

$$P = C^d \bmod N$$

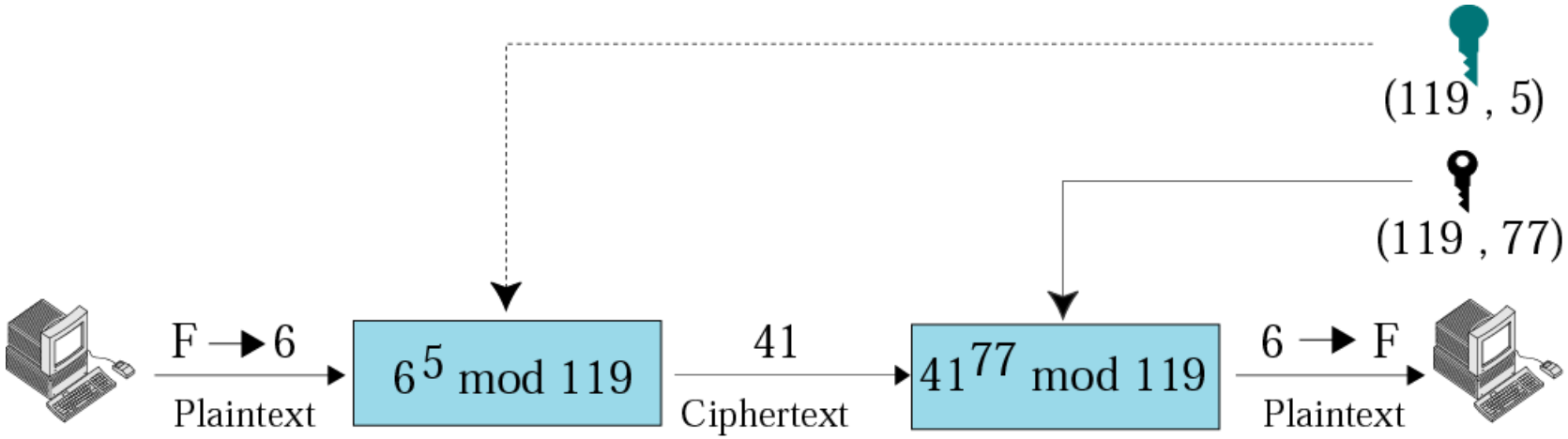
ตัวอย่าง: สมมติว่า private key = (119,77) และ public key = (119,5) ผู้ส่งต้องการส่งข้อมูล 'F' สมมติว่าเราแทน 'F' ด้วยเลข 6 (เพราะ 'F' เป็นอักษรตัวที่ 6) อัลกอริธึมเข้ารหัสจะคำนวณ $C = 6^5 \bmod 119 = 41$ ตัวเลข 41 จะถูกส่งไปยังผู้รับเป็น ciphertext เมื่อผู้รับได้รับเลข 41 เขาจะทำการถอดรหัสโดยคำนวณ $P = 41^{77} \bmod 119 = 6$ แล้วจึงตีความว่าเป็นข้อมูล 'F' ขั้นตอนการทำงานโดยรวมแสดงในรูปที่ 16.5

ข้อสังเกต: ในทางปฏิบัติ ตัวเลข d และ e จะมีขนาดใหญ่มาก อาจเป็นเลข 10-20 หลัก ทำให้การเดาหรือพยายามที่จะหาเลขเหล่านี้ต้องใช้เวลานานเป็นเดือนหรือไม่ก็เป็นปี แม้ว่าจะต้องใช้เครื่องคอมพิวเตอร์ที่มีประสิทธิภาพมากๆก็ตาม

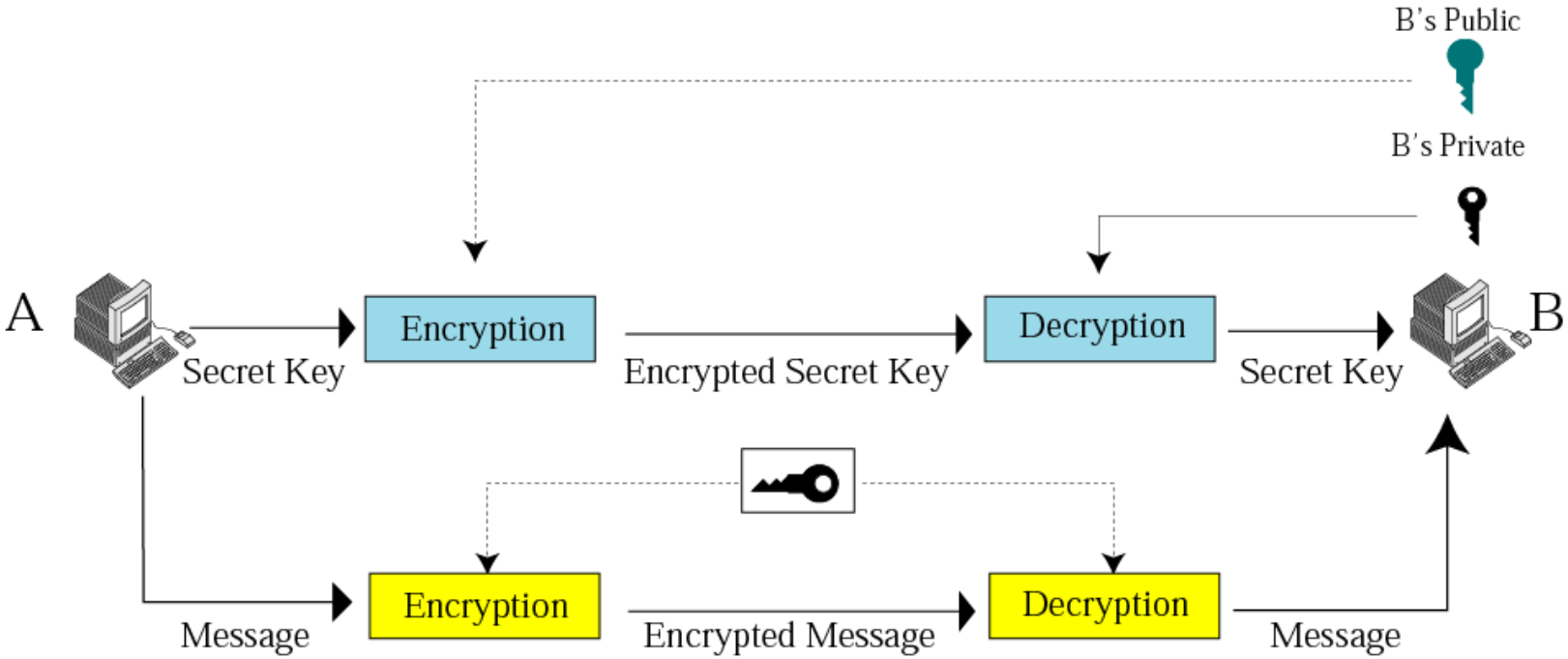
การเลือก public key และ private key: คำถามที่เราสนใจคือเราจะเลือกตัว เลข N, d, และ e อย่างไร ผู้ที่คิดอัลกอริธึมนี้ได้พิสูจน์ทางคณิตศาสตร์ว่าถ้าใช้วิธีการต่อไปนี้แล้ว อัลกอริธึม RSA จะสามารถทำงานได้แน่นอน:

1. **เลือกจำนวนเฉพาะที่มีค่ามากๆ 2 จำนวนสมมติว่าเป็น p กับ q**
2. **คำนวณหาค่า N = p x q**
3. **เลือกเลข e (ต้องน้อยกว่า N) ที่ e และ (p-1)(q-1) เป็น relatively prime**
4. **เลือก d ที่ (e x d) mod [(p-1)(q-1)] = 1**

ข้อดีและจุดอ่อนของวิธี public key: ข้อดีของการมี 2 keys ทำให้ลดจำนวน key โดยรวมลงได้มาเช่นถ้ามีคน 1 ล้านคนสื่อสารกัน จะใช้ key เพียง 2 ล้าน key เท่านั้น จุดอ่อนของวิธีนี้คือความซับซ้อนของอัลกอริธึม ถ้าข้อมูลมีขนาดใหญ่ การเข้ารหัสและถอดรหัสจะใช้เวลานาน



รูปที่ 16-5 RSA



รูปที่ 16-6 การผสมกัน (Combination)

16.2

ลายเซ็นอิเล็กทรอนิกส์

DIGITAL SIGNATURE

คงจำได้ว่าความปลอดภัยมีลักษณะที่สำคัญ 4 ลักษณะ ลักษณะแรกคือ privacy ซึ่งสามารถทำได้โดยใช้ secret key และ public key ดังที่ได้อธิบายไปแล้ว ส่วนอีก 3 ลักษณะที่เหลือคือ integrity, authentication, และ nonrepudiation สามารถดำเนินการภายใต้แนวคิดเดียวคือการเซ็นชื่อ (authenticating) ในเอกสาร โดยเจ้าของเอกสารหรือผู้ส่ง เมื่อผู้ส่งเซ็นชื่อในเอกสารแล้วจะเปลี่ยนแปลงลายเซ็นไม่ได้ เหมือนกับข้อสอบทุกฉบับ ผู้ออกข้อสอบจะต้องเซ็นชื่อกำกับทุกหน้า ในกรณีนี้ผู้เซ็นไม่สามารถปฏิเสธความรับผิดชอบได้หากมีอะไรเกิดขึ้น ในทำนองเดียวกันการส่งเอกสารอิเล็กทรอนิกส์ก็ต้องมีการเซ็นเอกสารเช่นเดียวกัน วิธีการนี้เรียกว่าลายเซ็นดิจิตอล หรือ digital signature

ลายเซ็นอิเล็กทรอนิกส์สามารถทำได้ 2 แบบคือ การเซ็นทั้งเอกสาร (sign the whole document) กับ การเซ็นส่วนย่อยของเอกสาร (sign a digest of the document) ซึ่งมีรายละเอียดดังต่อไปนี้

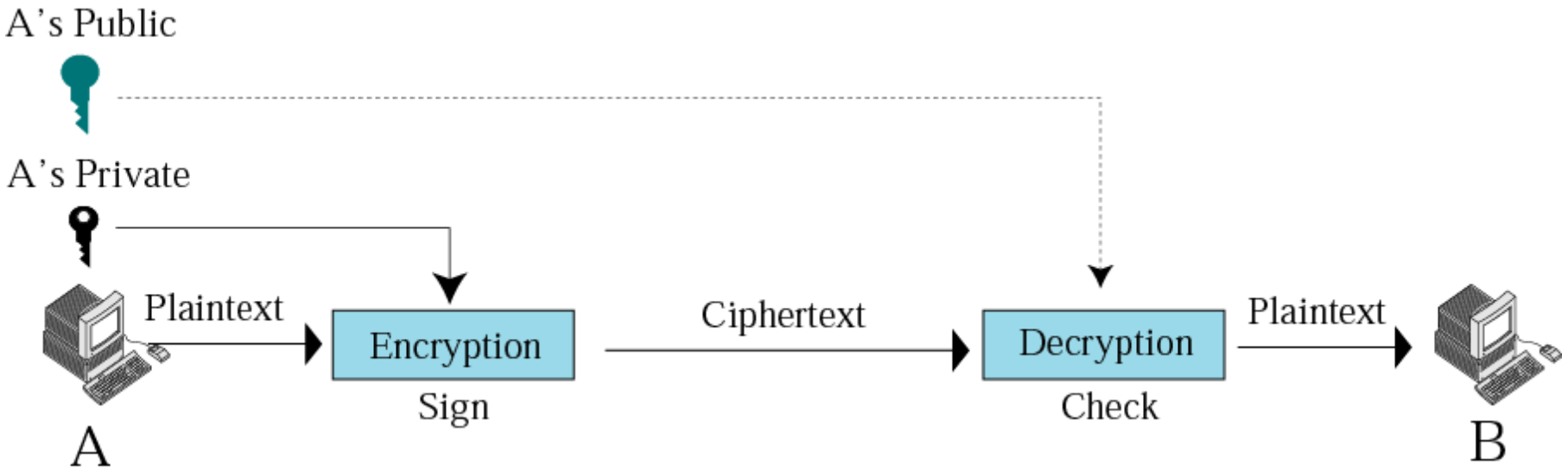
(1) การเซ็นเอกสารทั้งหมด: เราสามารถใช้การเข้ารหัสแบบ private key ทำการเซ็นเอกสารทั้งหมดได้ ในกรณีนี้ผู้ส่งใช้ private key ทำการเข้ารหัสข้อมูล ส่วนผู้รับจะใช้ public key ของผู้ส่งทำการถอดรหัส นั่นคือเราจะใช้ private key สำหรับการเข้ารหัส และใช้ public key สำหรับการถอดรหัส ดังรูปที่ 16.7

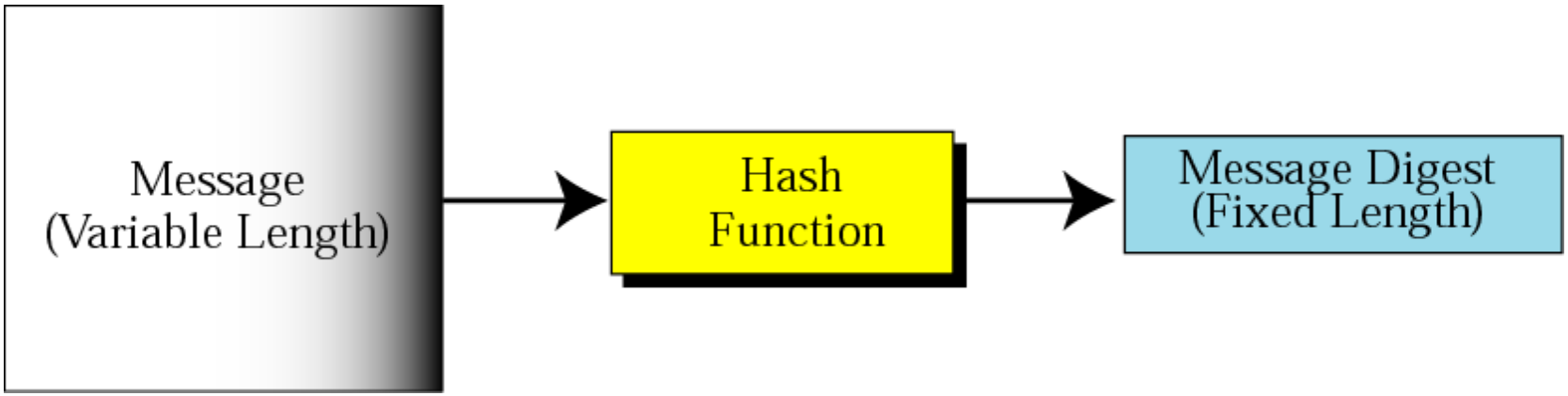
(2) การเซ็นส่วนย่อยของเอกสาร: เพื่อให้เกิดความรวดเร็ว การเซ็นเอกสารจะกระทำเป็นเพียงบางส่วนเท่านั้น ในกรณีนี้ผู้ส่งเอกสารจะถูกแบ่งเป็นส่วนย่อย แล้วทำการเซ็นแต่ละส่วน (เข้ารหัสด้วย private key ของตนเอง) ทางด้านผู้รับจะทำการตรวจสอบลายเซ็นในเอกสารแต่ละส่วน (ถอดรหัสด้วย public key ของผู้ส่ง)

วิธีการนี้ใช้เทคนิคที่เรียกว่า “hash function” เพื่อสร้างส่วนย่อยๆของเอกสาร ไม่ว่าเอกสารหรือข้อมูลจะยาวเท้าใดก็ตาม แต่ส่วนย่อยที่ถูกแบ่งจะต้องมีขนาดคงที่ (โดยปกติจะมีขนาด 128 บิต) ดังแสดงในรูปที่ 16.8

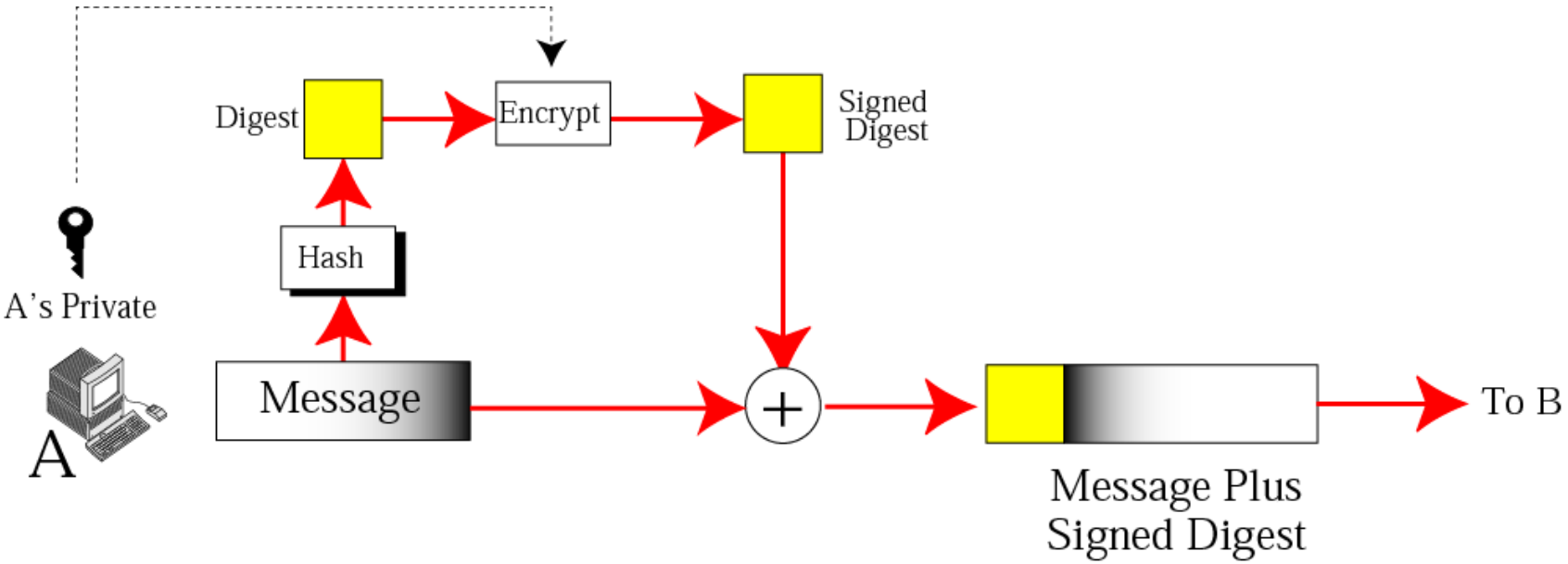
Figure 16-7

Signing the whole document

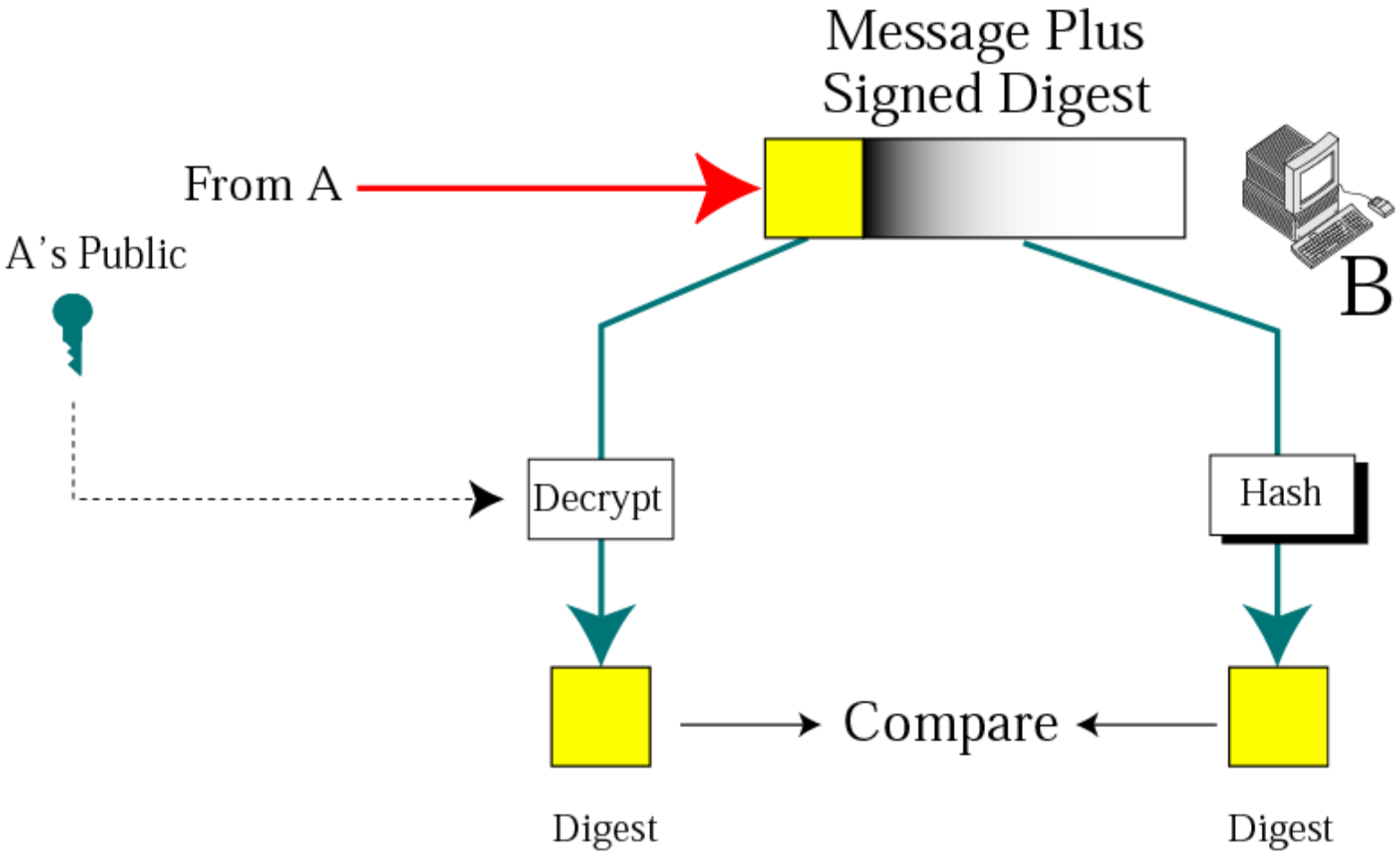




รูปที่ 16-8 Signing the digest



รูปที่ 16-9 Sender site



รูปที่ 16-10 Receiver site

คำสำคัญ

Authentication, Ciphertext, DES, decryption, digital signature, encryption, nonrepudiation, permutation, plaintext, private key, public key, RSA, secret key, security