CMPE 220

Class 18 – System Security - Continued

Types of Threats

- Breach of Confidentiality
- Breach of Integrity
- Breach of Availability
- Theft of Service
- Denial of Service
- Ransom!

Types of Attacks

- Virus
- Trojan Horse
- Trap Door / Back Door
- Logic Bomb

- Port Scanning
- Masquerading
- Man-in-the-Middle

- Human Attacks

San José State University

Mitigating Threats

- Principle of Least Authority (POLA)
  - Permissions policies
- Password Policies
- 2-Factor Authentication
- Active Filtering
  - Firewalls
  - Security Software

Mitigating Threats

- Software Updates
- Encryption
- Physical Security
- Staff Education
- Backups

San José State University

Threat Detection

- Access Logging
- Look for known virus “signatures”
- Checksums on system files
- Monitor system resource usage (profiling)

The Biggest Risk to Computer Security

- PEOPLE!
  - Nefarious
  - Dumb
- People violate security protocols
  - Especially if protocols are too onerous
- People install back doors
- People use weak passwords

San José State University

Encryption: One Way Algorithms

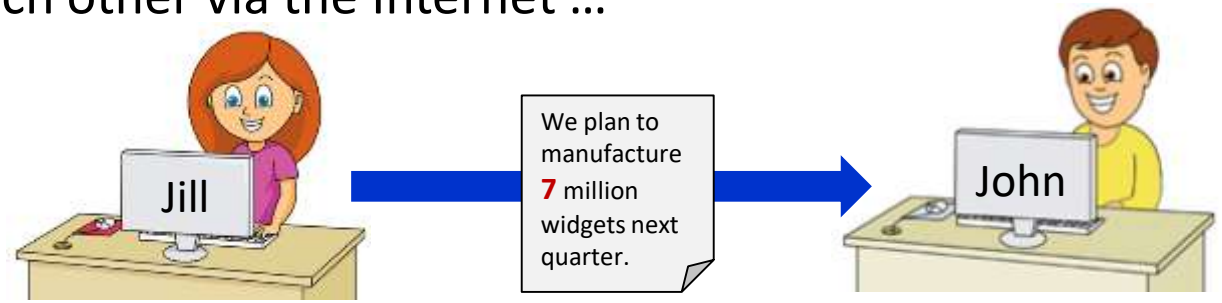
- A one-way hash function is a cryptographic algorithm that turns an arbitrary-length input into a fixed-length binary value, and this transformation is one-way, that is, given a hash value it is *statistically infeasible* to re-create a document that would produce this value.
- There are three widely used hash algorithms: MD4, MD5, and SHA. MD4 and MD5 produce 128-bit hashes, and SHA a 160-bit hash.

Encryption: Linux Passwords

- Plaintext password is encrypted and stored
- When user logs in, password is encrypted and compared to the stored password
- The system does not store the plaintext password, so it can’t be stolen

Public Key Encryption

- Sending data such as email messages to each other via the Internet …

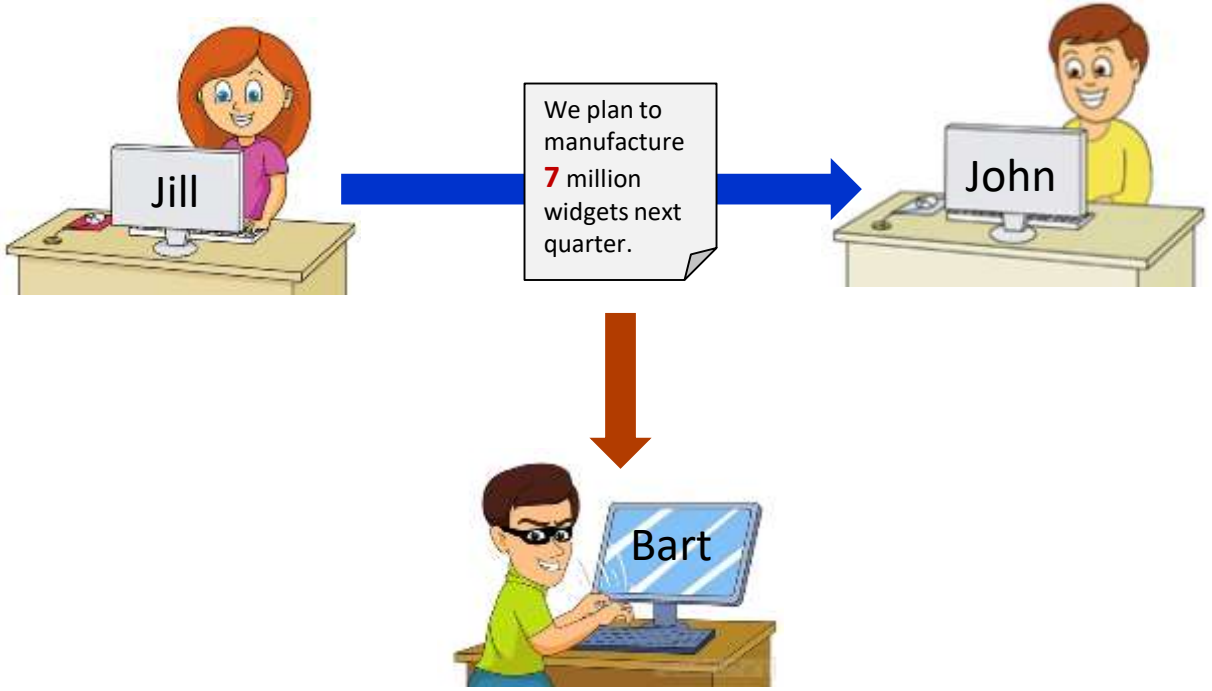


- … is like sending <u>postcards</u> via the U.S. mail system.

- Anyone can read the message along the way!

San José State University

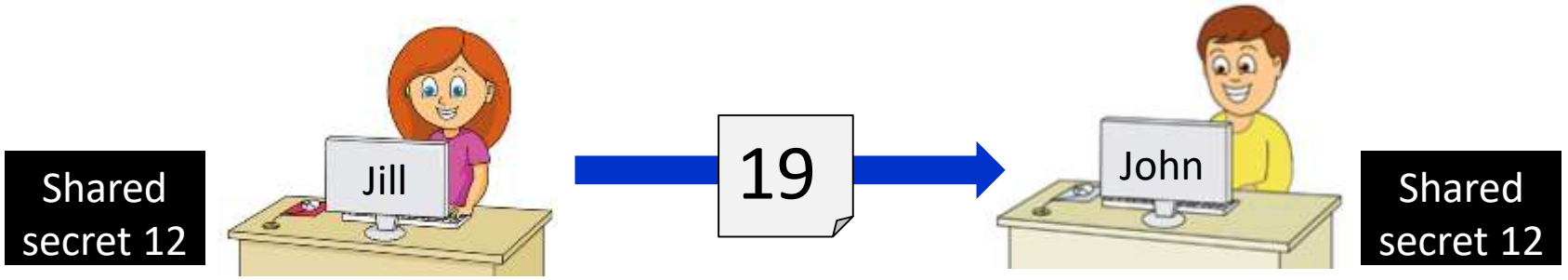
Security, *cont'd*



- How can we keep the nefarious Bart from reading confidential messages that Jill and John are sending each other?

The Shared Secret

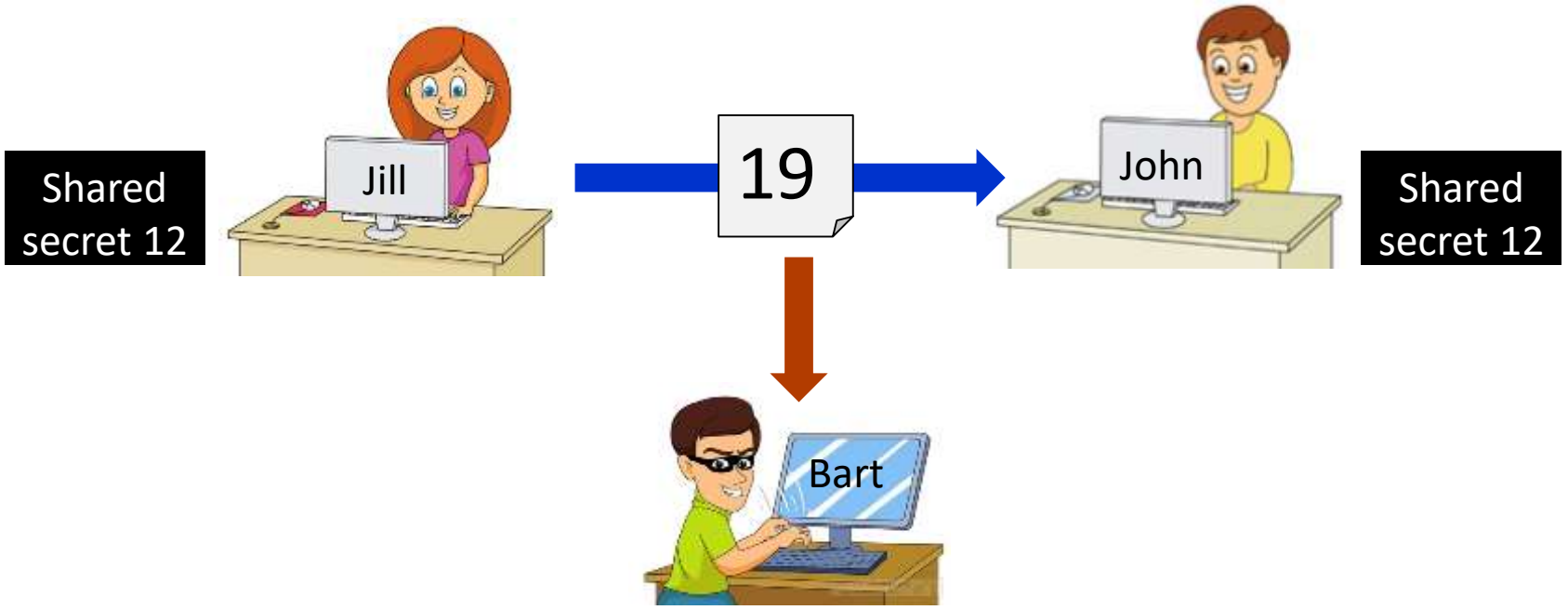
- Jill needs to send a message containing the confidential data 7 to John.



- John and Jill can agree ahead of time to a shared secret – the number 12.
- Then Jill can encrypt the data by adding 12 to the confidential data 7.
- John decrypts the data by subtracting 12.

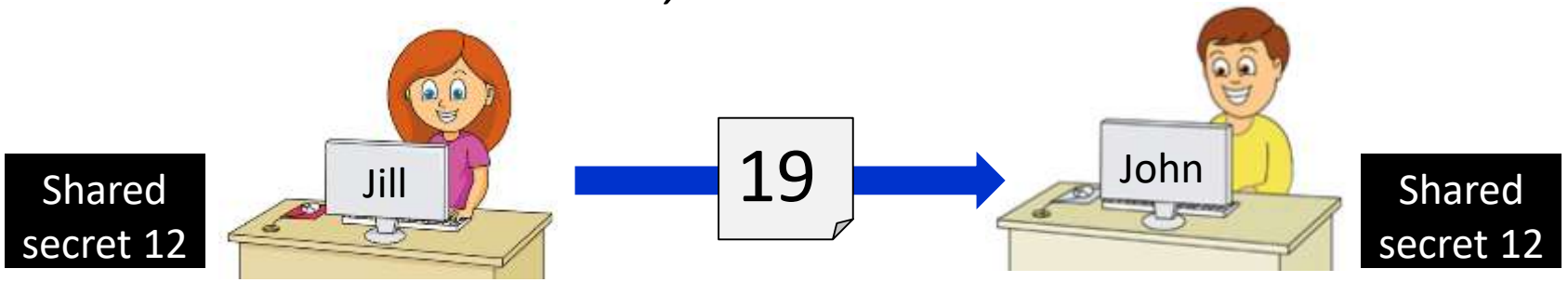
San José State University

The Shared Secret, *cont'd*



- Because Bart doesn't know the shared secret 12, he won't be able to decrypt the message and obtain the confidential data 7.

The Shared Secret, *cont'd*



- But this shared secret solution has problems.
  - Jill and John must arrange beforehand to share the secret 12.
  - What if Jill doesn't already know John?
  - What if Jill wants to send the confidential data to all her vice presidents at the same time?

How can Jill and her recipients share a secret?

San José State University

Public Key Cryptography, *cont'd*

- How can Jill and her recipients share a secret number in order to encrypt the confidential data?

- A security scheme called public key cryptography was invented just for this purpose.

- In this simplified introduction, let's pretend that multiplication is a one-way operation.
  - Once you've multiplied two numbers, say 4x5=20, you can't recover the original numbers by dividing.
  - In other words, you can't do 20÷4=5 or 20÷5=4

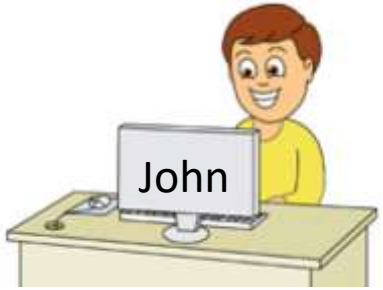
San José State University

Public Key Cryptography, *cont'd*

- Jill chooses a private key.
  - Let's suppose Jill chooses 10.

- Each person to whom Jill wants to send confidential data also chooses a private key.
  - Let's suppose John chooses 8.

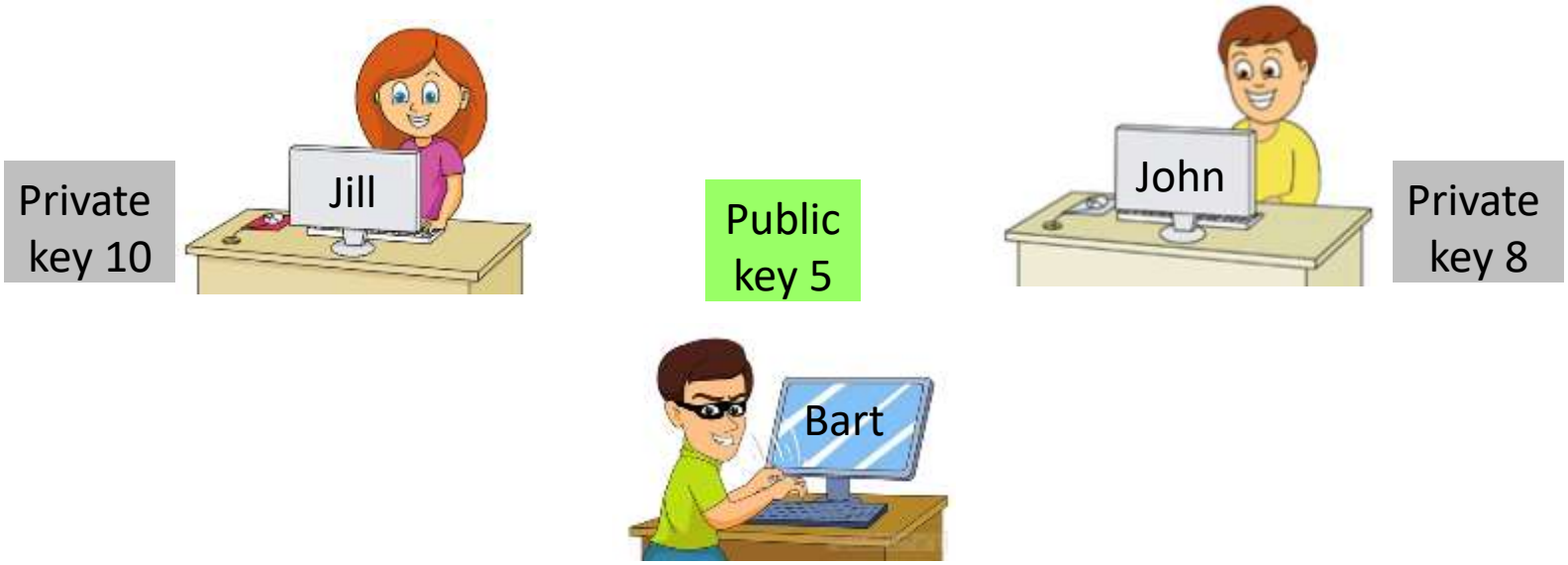
Private key 10



Private key 8

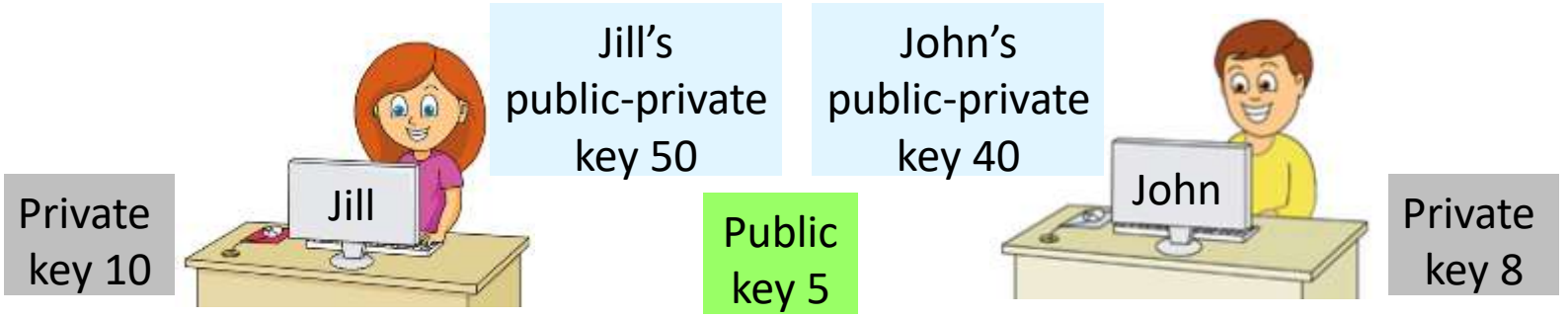
San José State University

Public Key Cryptography, *cont'd*



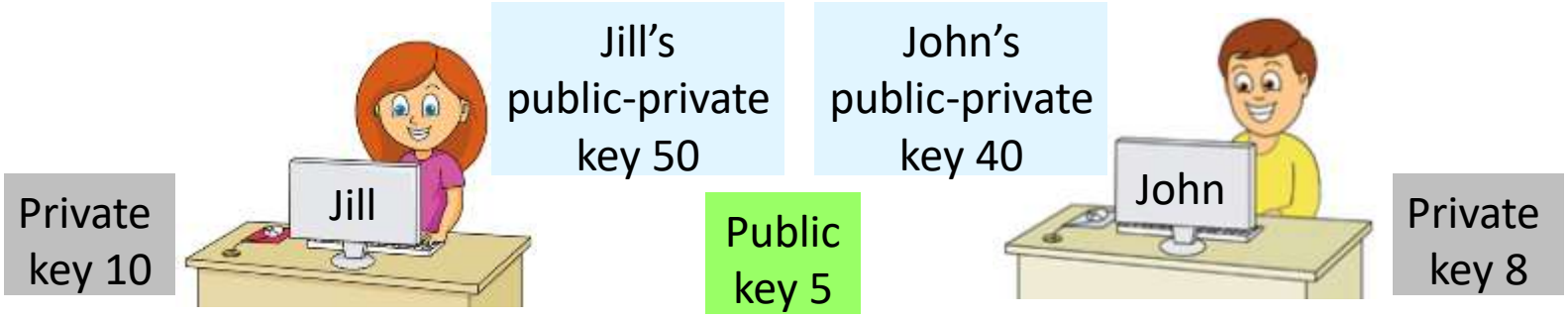
- Now Jill announces a public key.
  - Let's suppose the public key is 5.
- Everyone can see the public key.
  - Including the nefarious Bart.

Public Key Cryptography, *cont'd*



- Now Jill can create her public-private key.
  - She multiplies her private key by the public key: 10x5=50.
- John creates his public-private key.
  - He multiplies his private key by the public key: 8x5=40.

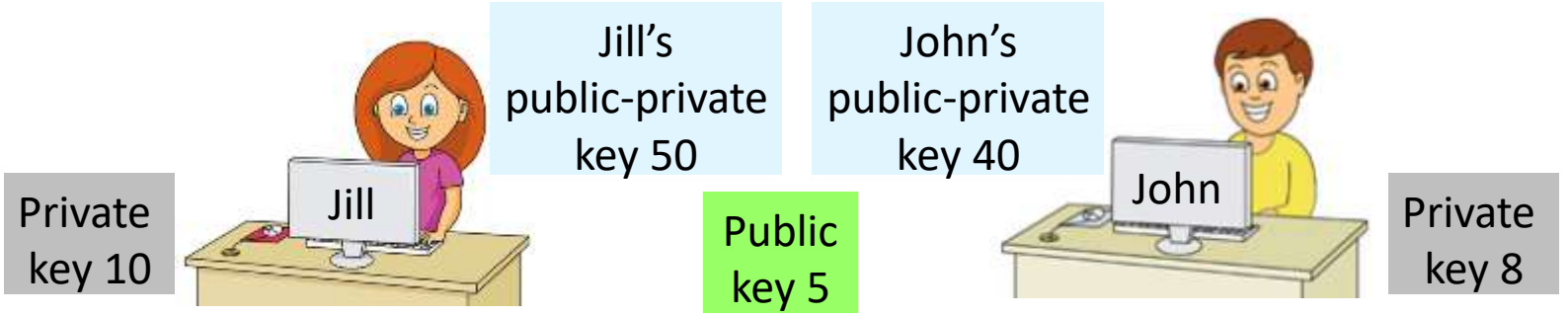
Public Key Cryptography, *cont'd*



Remember that we're pretending that multiplication is a one-way operation.

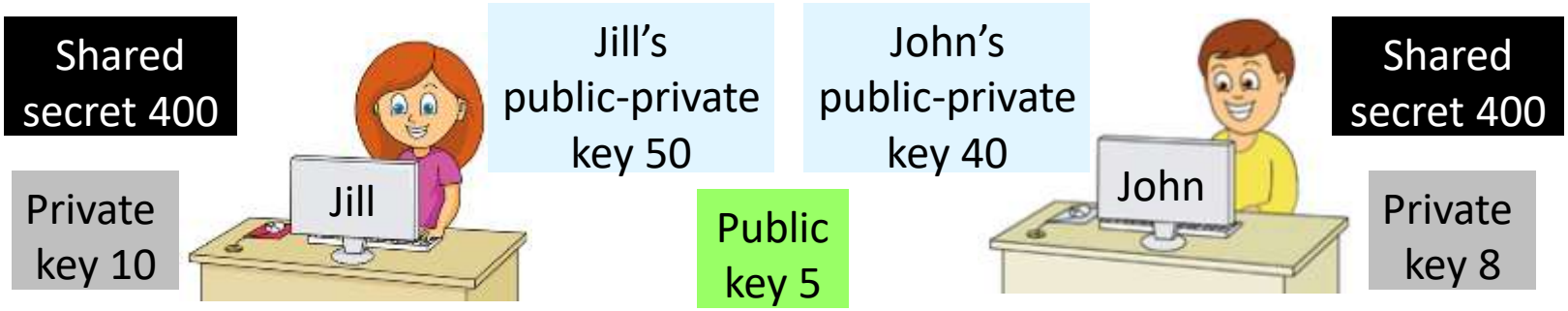
- We cannot discover Jill's private key 10 by dividing her public-private key 50 by the public key 5.
- We cannot discover John's private key 8 by dividing his public-private key 40 by the public key 5.

Public Key Cryptography, *cont'd*



- What is the goal of all this?
  - To create a shared secret between Jill and John.
- Jill multiplies John's public-private key by her private key: 40x10=400
- John multiplies Jill's public-private key by his private key: 50x8=400

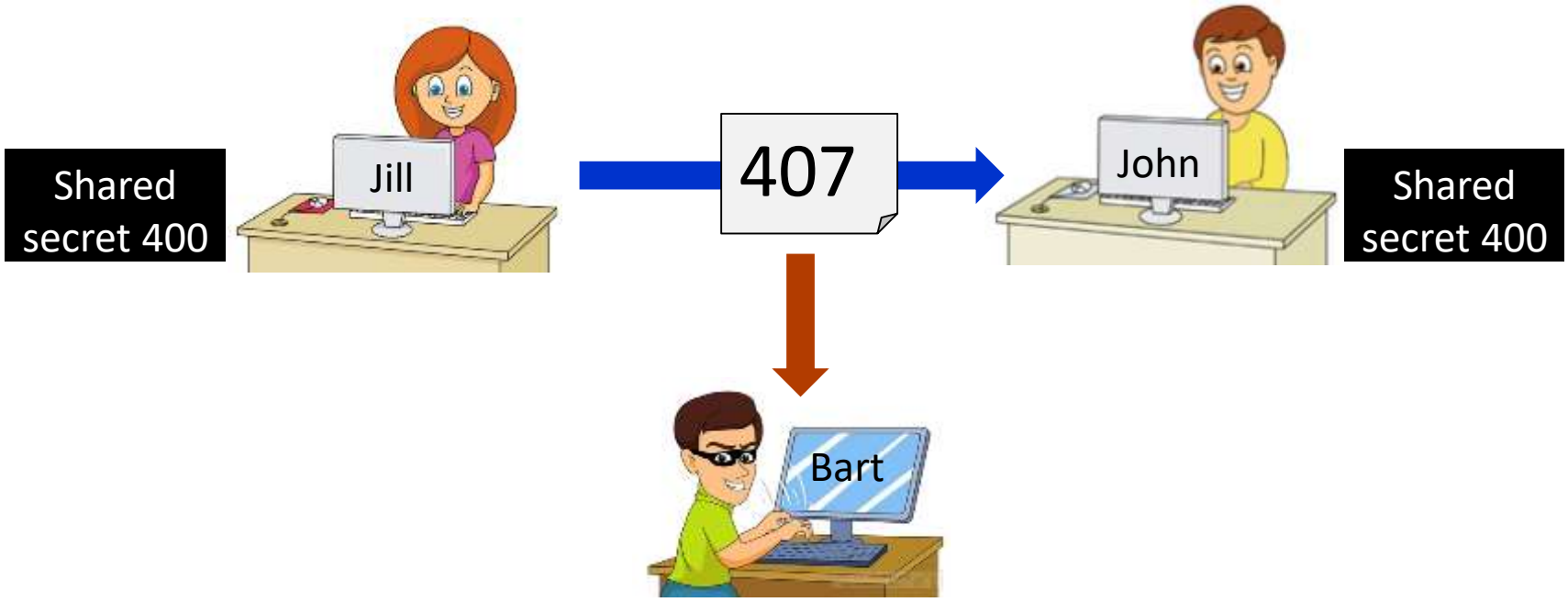
Public Key Cryptography, *cont'd*



- Now Jill and John have a shared secret 400.
- Jill can encrypt the confidential data 7 by adding the shared secret 400.
- John can decrypt the confidential data 7 by subtracting the shared secret 400.

San José State University

Public Key Cryptography, *cont'd*



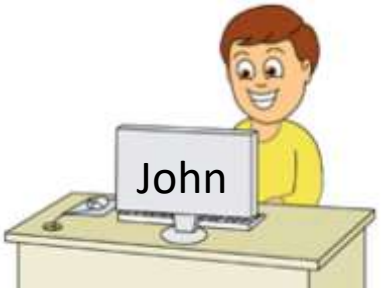
- Bart can't decrypt the 407 because he doesn't know the shared secret 400.

Public Key Cryptography, *cont'd*

- Public key encryption works with multiple recipients.
- Jill needs to send confidential data to both John and his twin brother Mark.
- Each picks a private key.

Private key 2

Private key 10

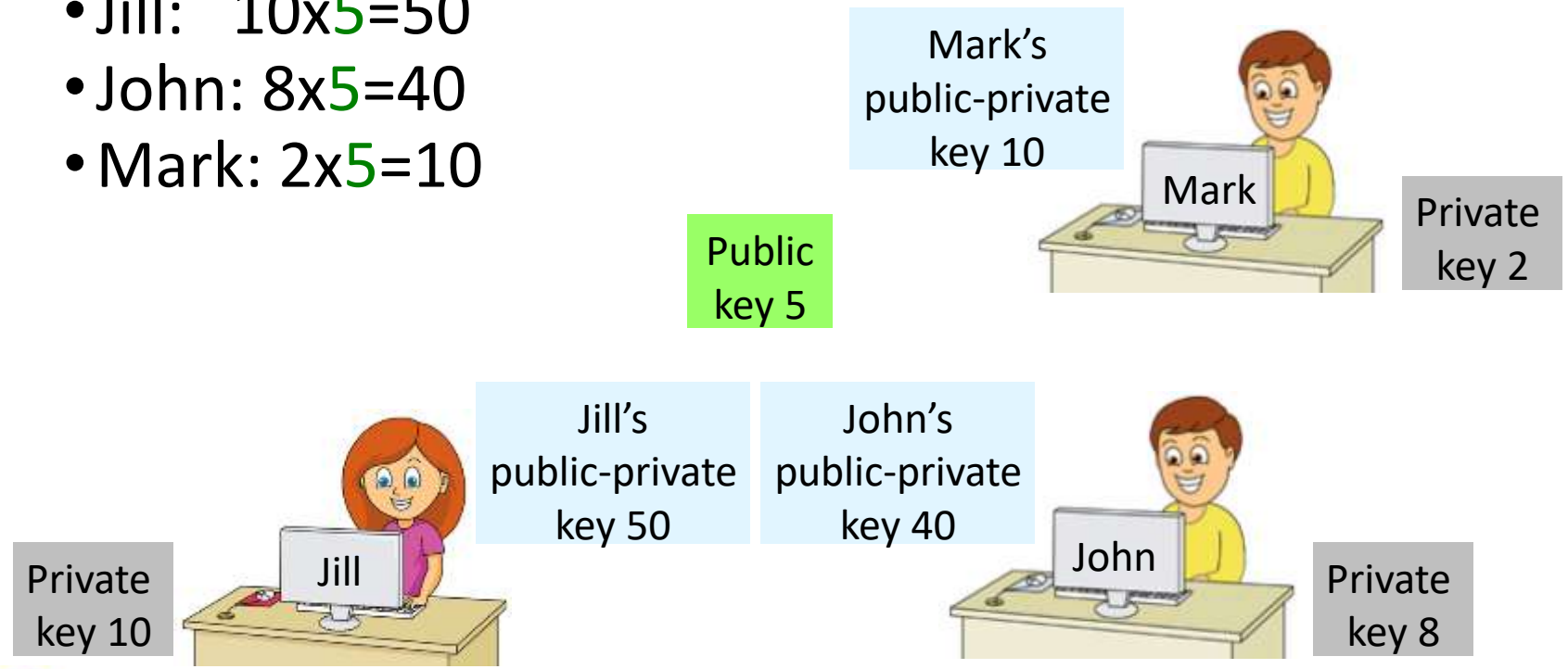


Private key 8

San José State University

Public Key Cryptography, *cont'd*

- Jill announces the public key 5, and everyone generates his or her public-private key.
  - Jill: 10x5=50
  - John: 8x5=40
  - Mark: 2x5=10

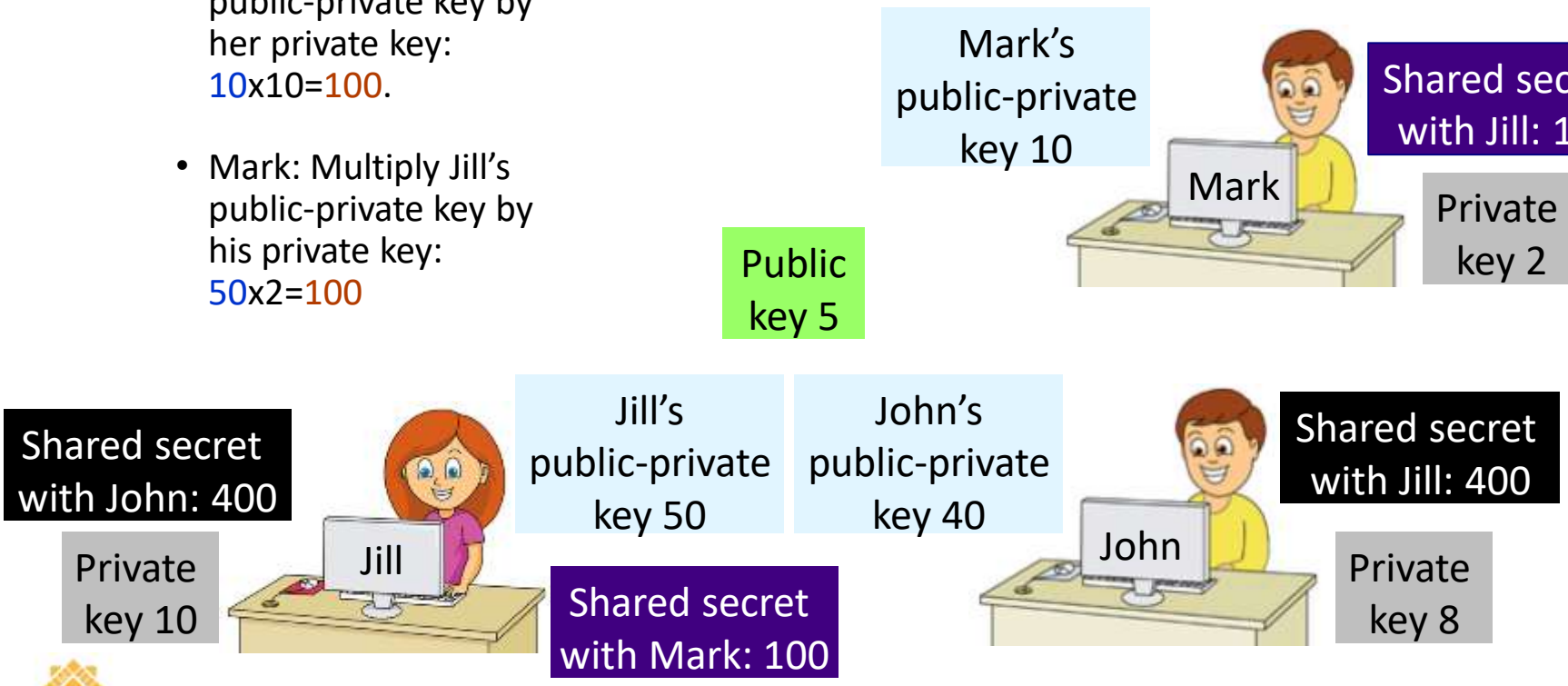


Public Key Cryptography, *cont'd*

- Jill will have a shared secret with each recipient.
  - Jill and John will share 400 between them, as before.
  - Jill and Mark will have a different shared secret.

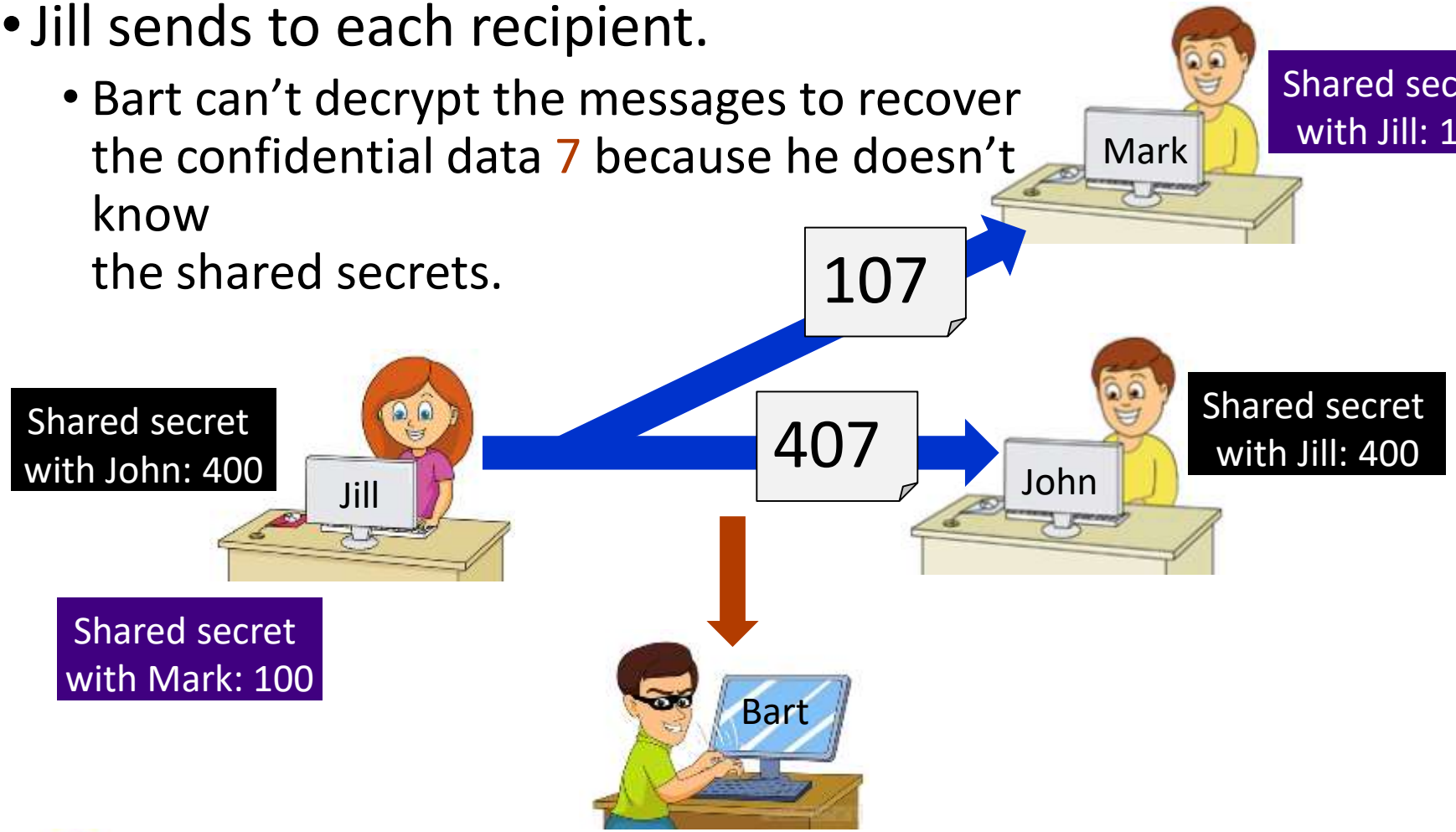
    - Jill: Multiply Mark's public-private key by her private key: 10x10=100.

    - Mark: Multiply Jill's public-private key by his private key: 50x2=100



Public Key Cryptography, *cont'd*

- Jill sends to each recipient.
  - Bart can't decrypt the messages to recover the confidential data 7 because he doesn't know the shared secrets.



Cryptography in the Real World

- Of course, in the real world, we can't use simple operations like multiplication and addition to generate keys and to encrypt data.
  - Multiplication and addition are not one-way operations.

- Real-world encryption uses very large prime numbers and modulo arithmetic.
  - Not even today's most powerful supercomputer can undo such operations.
  - Worry: Can quantum computers in the future?

When is Cryptography Used?

- Public key cryptography is a key exchange protocol first published by Whitfield Diffie and Martin Hellman in 1976.
  - It was actually invented earlier in 1970 by the British government, but it was classified.

- Whenever you visit a secure website, you are using the Diffie-Hellman protocol or a variant.
  - A secure website has a URL that starts with **`https:`** instead of **`http:`**

Computer Security as a Career

- Cybersecurity is a hot field.
  - Computers are used everywhere.
  - Big data.
  - Privacy issues.

San José State University

Protecting Yourself

- How to Safe on the Internet: Tips for Computers, Phones, and More
  - https://turbofuture.com/internet/Staying-Safe-on-the-Internet