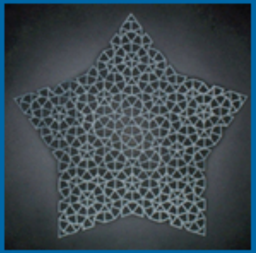
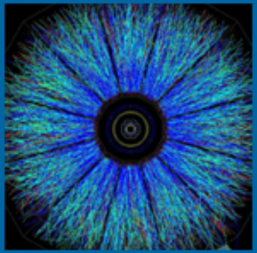
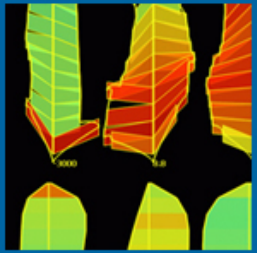
Swansea University
Prifysgol Abertawe

CS130:Professional Issues Cryptography and Data Security: Solving the Key Exchange Problem



College of Science
Coleg Gwyddoniaeth

www.swansea.ac.uk/science

Topic Learning Goals

How does encryption secure a message?

What is Diffie-Hellman key exchange?

What is RSA Public Key/Private Key encryption?

What maths underpin modern encryption?

Encryption as a Locked Box

Picture encryption as a box that you secure a message in by locking it with a key (shared secret)

- But everything is digital so it can be copied infinitely…

The sender locks up the box and hands it along in pass-the-parcel style

When it reaches the intended recipient, they then open the box with their key

But how does the recipient get their key?

Diffie-Hellman Principles

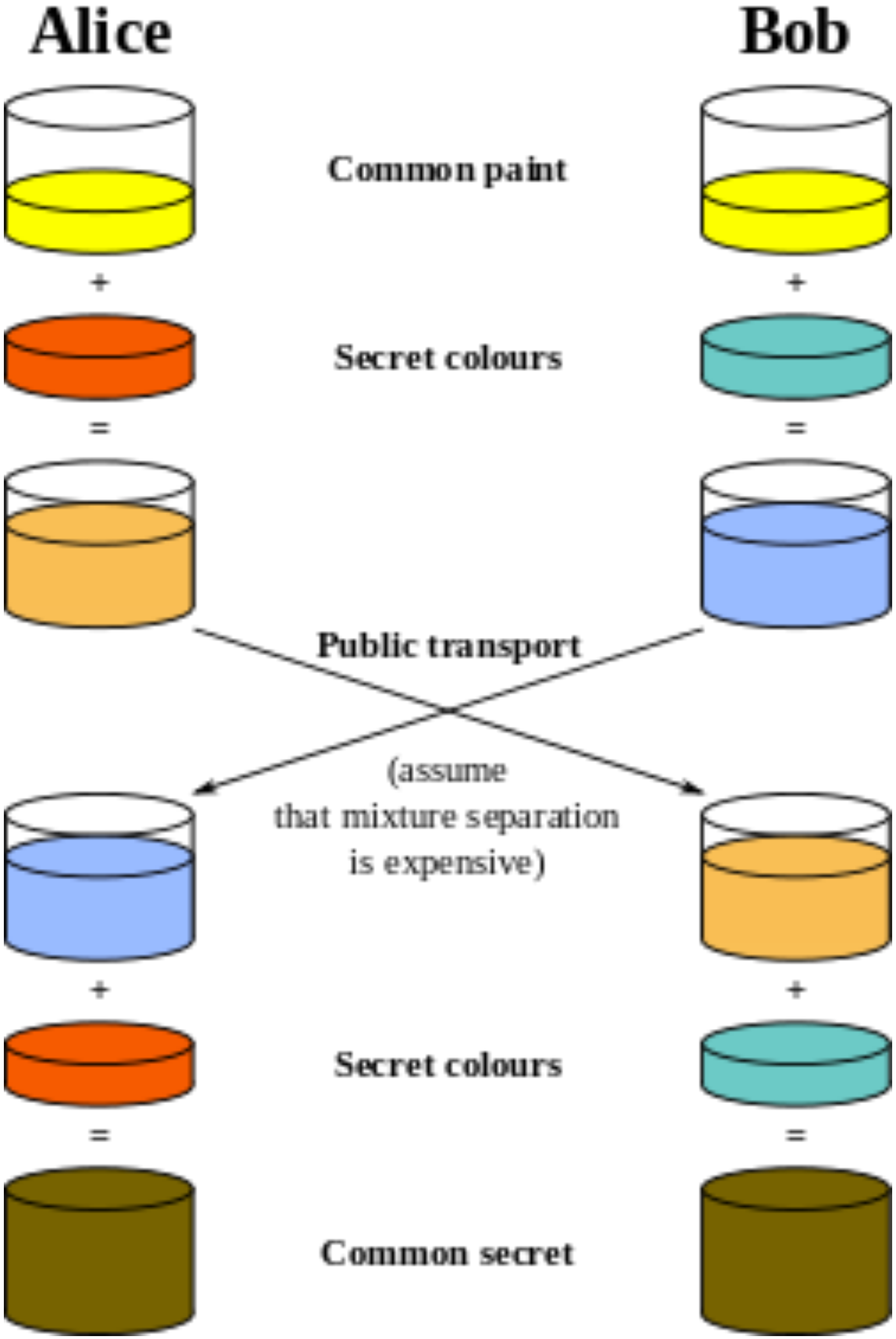
A way to establish a shared secret between two parties where communication can be seen by Eve

Relies on maths that works like paint works

- Specifically factorisation

What is the weakness of this approach?

- ???



Diffie-Hellman Principles

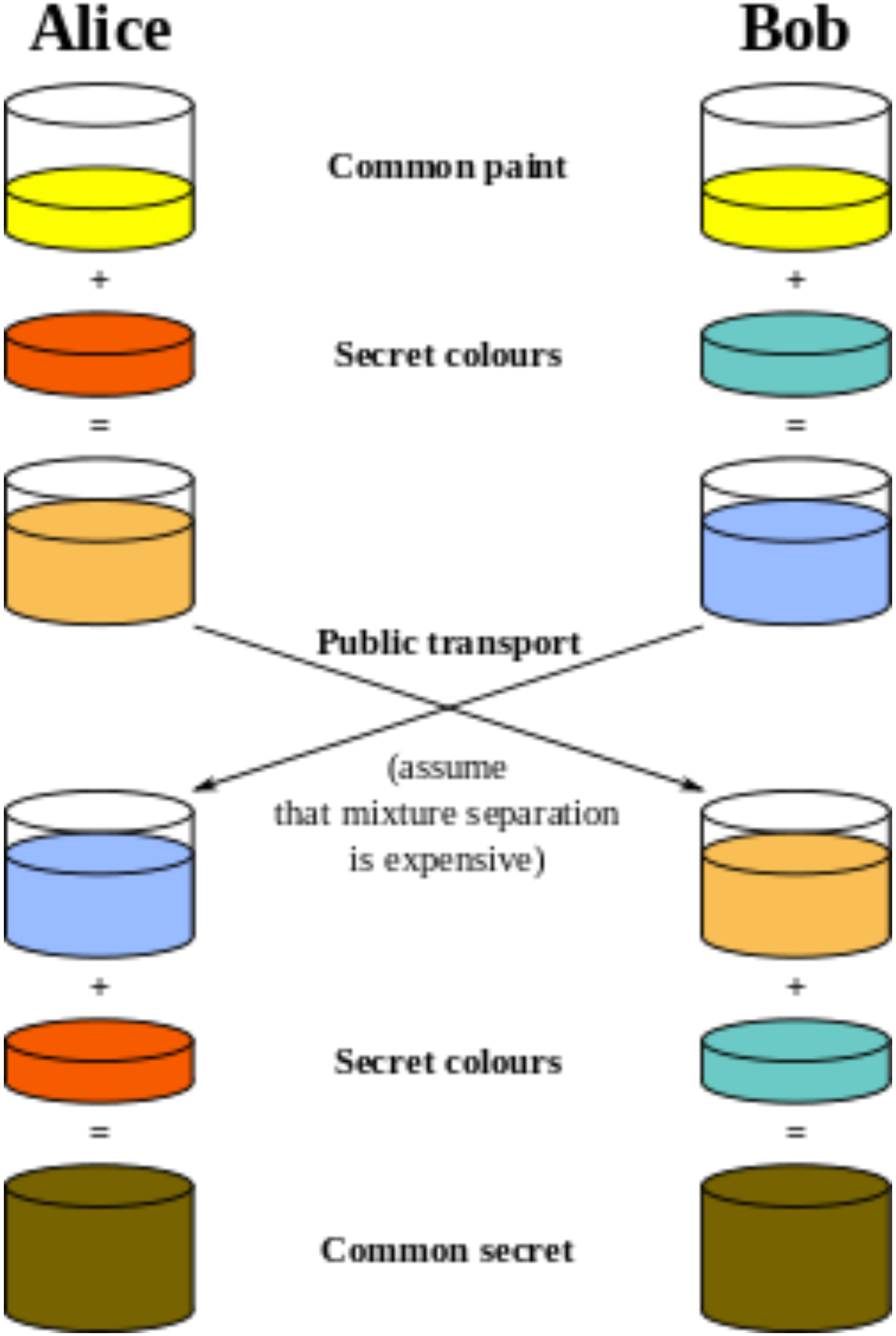
A way to establish a shared secret between two parties where communication can be seen by Eve

Relies on maths that works like paint works

- Specifically factorisation

What is the weakness of this approach?

- The Man in the Middle Attack where Eve intercepts and pretends to be Alice and Bob



Public Key/Private Key Pairs - RSA

A “lock” anyone can close (encrypt) with your public key but only your private key can open

RSA generates a ***public key*** and a ***private key***

- You publish the public key online and let anyone use it to lock up their message to you in a box
- You keep the private key a secret that only you know and only it can open the messages encrypted with the public key

Anybody can encode a message to send to you using the public key but only the private key can decrypt it

Public Key, Private Key Encryption

```
// Code to run on Sender’s machine
private Message pkEncrypt(Message originalMessage, PublicKey pk)
{
        new Message em = SomeFunkyMaths(originalMessage,pk);
        return em;
}

// em is sent over the internet to the recipient

// code to run on Recipients machine
private Message pkDecrypt(Message em, PrivateKey k)
{
        new Message decryptedMessage = reverseFunkyMaths(em,k);
        return decrypted;
}

// now decrypted should equal the original message
```

Key Idea: This Doesn’t Work

```
// A malicious user intercepts your message and tries to
// decrypt it using your Public Key

private Message pkDecrypt(Message em, PublicKey k)
{
       new Message decryptedMessage = reverseFunkyMaths(em,k);
       return decrypted;
}

// decrypted will still be unreadable in this case
```

Encryption foundation: One way functions

Trapdoor or one way functions, easy to calculate the result with inputs but hard to reverse

Easy: 2 x 2 x 2 x 3 = ?

Solving the Problem: Public Key Encryption

Trapdoor or one way functions, easy to calculate the result with inputs but hard to reverse

Easy: 2 x 2 x 2 x 3 = 24

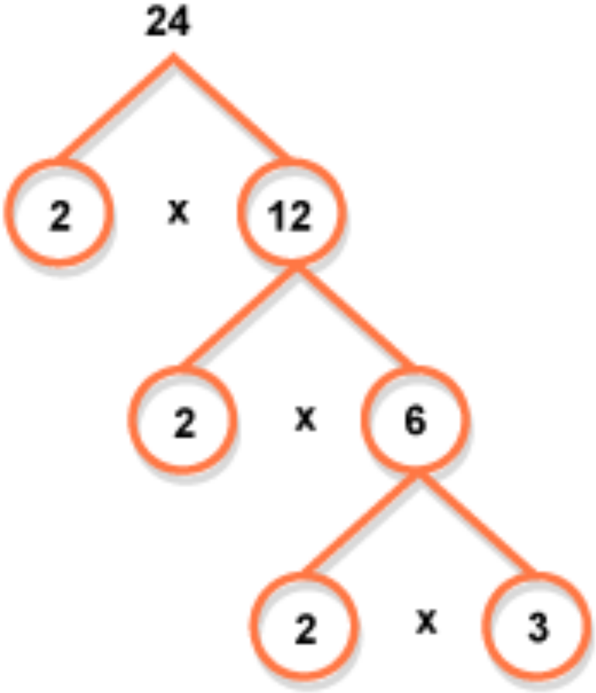
Hard: Prime factorisation of 24

Solving the Problem: Public Key Encryption

Trapdoor or one way functions, easy to calculate the result with inputs but hard to reverse

Easy: 2 x 2 x 2 x 3 = 24

Hard: Prime factorisation of 24



Solving the Problem: Public Key Encryption

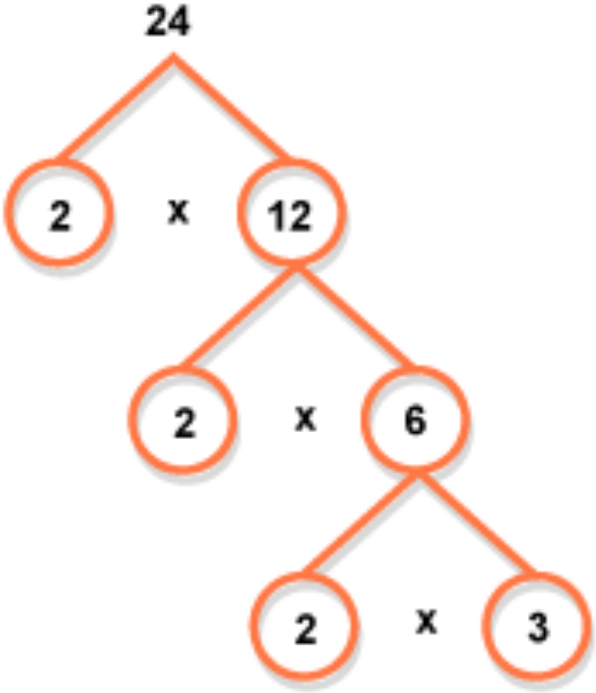
Trapdoor or one way functions, easy to calculate the result with inputs but hard to reverse

Easy: 2 x 2 x 2 x 3 = 24 (PrivKey)

Hard: Prime factorisation of 24 (PubKey)

So, we encrypt using functions that are easily decrypted (reversed) if you know their prime factors like Eulers totient function

- Still more to this like block size we will not get into



1. Select 4 prime numbers - private key
2. Multiply them to get your public key
3. Swap with a friend and figure out their private key

2, 3, 5, 7, 11, 13, 17, 19, 23, 29

Public Key Encryption – How Big?

2039568783564019774057658669290345772801939933

Public Key Encryption – How Big?

2039568783564019774057658669290345772801939933143482630947726464532830627227012776329366160631440881733123728826771238795387094001583065673383282791544996983660719067664400370742171178056908727928481491120222863321448761833763265120835748216479339929612497319836219304274280243803104015000563790123

Public Key Encryption – UK Population

2039568783564019774057658669290345772801939933143482630947726464532830627227012776329366160631440881733123728826771238795387094001583065673383282791544996983660719067664400370742171117805690872792848149112022863321448761833763265120835748216479339929612497319836219304274280243803104015000563790123

Public Key Encryption – World Population

2039568783564019774057658669290345772801939933143482630947726464532830627227012776329366160631440881733123728826771238795387094001583065673383282791544996983660719067664400370742171117805690872792848149112022863321448761833763265120835748216479339929612497319836219304274280243803104015000563790123

People who’ve ever lived

2039568783564019774057658669290345772801939933143482630947726464532830627227012776329366160631440881733123728826771238795387094001583065673383282791544996983660719067664400370742171117805690872792848149112022286332144876183376326512083574821647933992961249731983621930427428024380310401500056379 0123

Grains of sand on earth

2039568783564019774057658669290345772801939933143482630947726464532830627227012776329366160631440881733123728826771238795387094001583065673383282791544996983660719067664400370742171117805690872792848149112022286332144876183376326512083574821647933992961249731983621930427428024380310401500056379
0123

Age of the Universe (in seconds)

203956878356401977405765866929034577280193
993314348263094772646453283062722701277632
936616063144088173312372882677123879538709
400158306567338328279154499698366071906766
440037074217117805690872792848149112022286
332144876183376326512083574821647933992961
249731983621930427428024380310401500056379
0123

Atoms in the Universe

20395687835640197740576586692903457728019399331434826309477264645328306272270127763293661606314408817331237288267712387953870940015830656733832827915449969836607190676644003707421711780569087279284814911202228633214487618337632651208357482164793399296124973198362193042742802438031040150005637901230123

We still have a major problem

Public Key Encryption is the foundation of the modern Internet but how do you know the owner of a Public Key is who they claim to be?

Topic Learning Goals

How does encryption secure a message?

What is Diffie-Hellman key exchange?

What is RSA Public Key/Private Key encryption?

What maths underpin modern encryption?

Topic Learning Goals

How does encryption secure a message?

It locks it in a “box” that you need a key to open

What is Diffie-Hellman key exchange?

A system that mixes numbers together to allow public sharing of keys albeit with a vulnerability

What is RSA Public Key/Private Key encryption?

Locked boxes using two different keys, one which locks and one which unlocks

What maths underpin modern encryption?

Trapdoor functions like prime factorisation