School of Computing
FACULTY OF ENGINEERING & PHYSICAL SCIENCES

UNIVERSITY OF LEEDS

COMP3911 Secure Computing

9: Network Defences

Nick Efford

https://comp3911.info

Last Time

- We reviewed key characteristics of TCP/IP and saw that TCP & IP headers were vulnerable to various abuses
- We noted that standard protocols provide no guarantees of confidentiality, authenticity, integrity
- We discussed some of the ways in which denial-of-service attacks can be conducted

Objectives

UNIVERSITY OF LEEDS

- To look at some network security measures, focusing in particular on those that are deployed in the Internet and Transport layers
- To discuss some well-known recent vulnerabilities in these security measures

Network Security Measures

- Two approaches considered here:
  - Block malicious traffic from entering local network
  - Protect legitimate traffic from tampering and information disclosure threats
- Some relevant technologies
  - Firewalls
  - TLS
  - IPsec

Firewalls…

- Are supposed to provide a **single choke point** for traffic wishing to enter or leave the local network
- Pass or block traffic according to **local security policy**
    - Simplest policy: block incoming connections, except for cases where ports need to be open (e.g., port 80 if you are running a web server)
    - Or can do **packet filtering**, scanning content for things you want to block…
- Need to be immune to penetration!

In Reality…

- World has changed since firewalls first became popular
- Very difficult to maintain a single choke point
  - Massive connectivity (WiFi, BYOD, etc)
  - … therefore many ways around traditional firewalls
- Firewall technology (port blocking, packet filtering, etc) is now found in a modern OS – ‘personal firewalls’

In Reality…

UNIVERSITY OF LEEDS

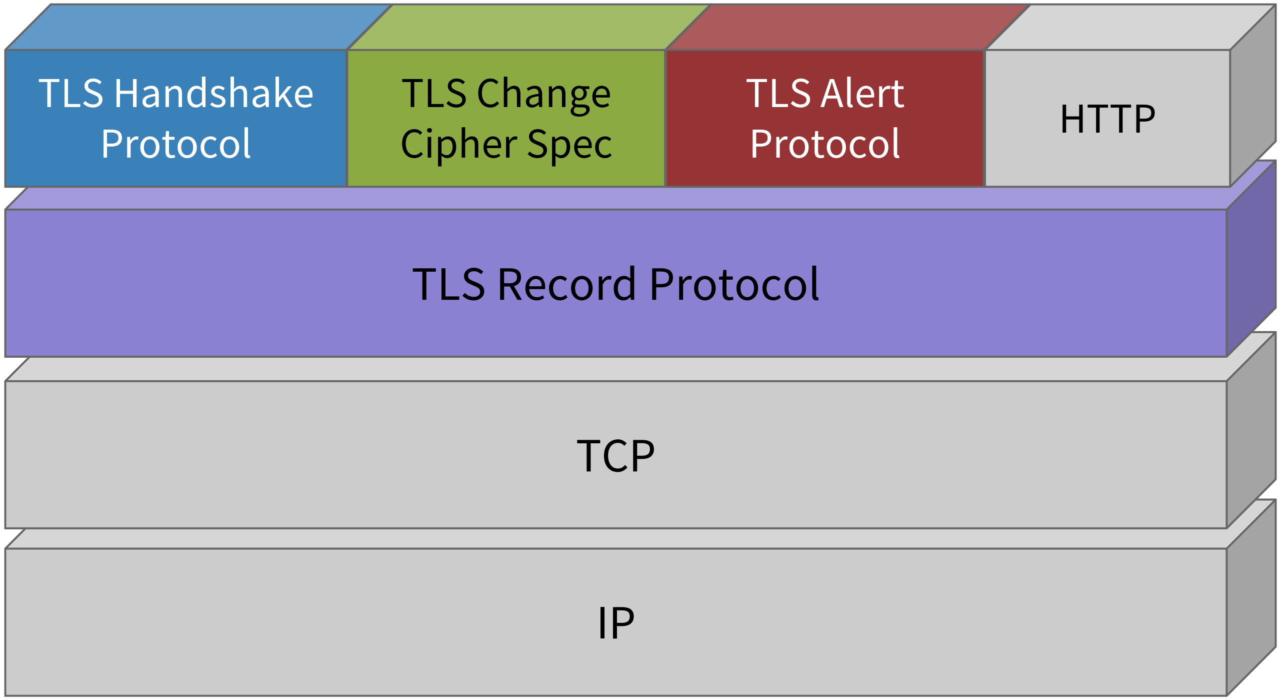
- No protection from internal threats – such as disgruntled employees, untrustworthy contractors, etc
- Can’t block all traffic; attackers can still do a lot via ports that have to be left open (e.g., port 80)
- Firewalls are computers running software, which contains vulnerabilities, which can be exploited…
    - Example: Cisco ASA vulnerability CVE-2016-1287 https://www.kb.cert.org/vuls/id/327976

See Chapter 5 of Bellovin for more on this…

Transport Layer Security

- Based on and replaces Secure Sockets Layer (SSL)
  - SSL 2.0 prohibited in 2011
  - SSL 3.0 deprecated in June 2015
- Not the same thing as SSH!
- Two implementation options
  - As part of underlying protocol suite
  - Embedded in specific packages
- Plays a key role in web security

TLS in the Protocol Stack



TLS Handshake Protocol

TLS Change Cipher Spec

TLS Alert Protocol

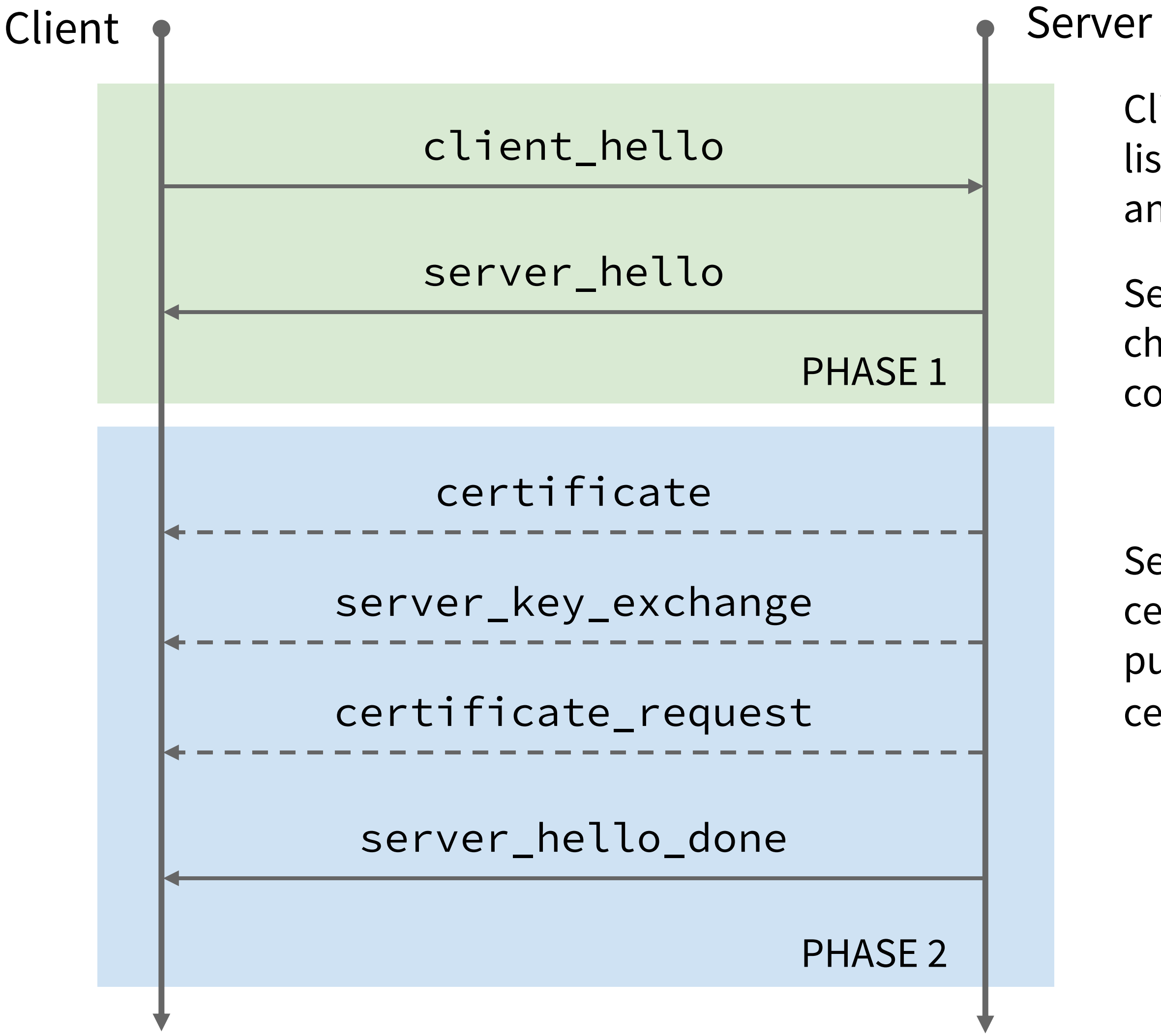
HTTP

TLS Record Protocol

TCP

IP

TLS Handshake Protocol

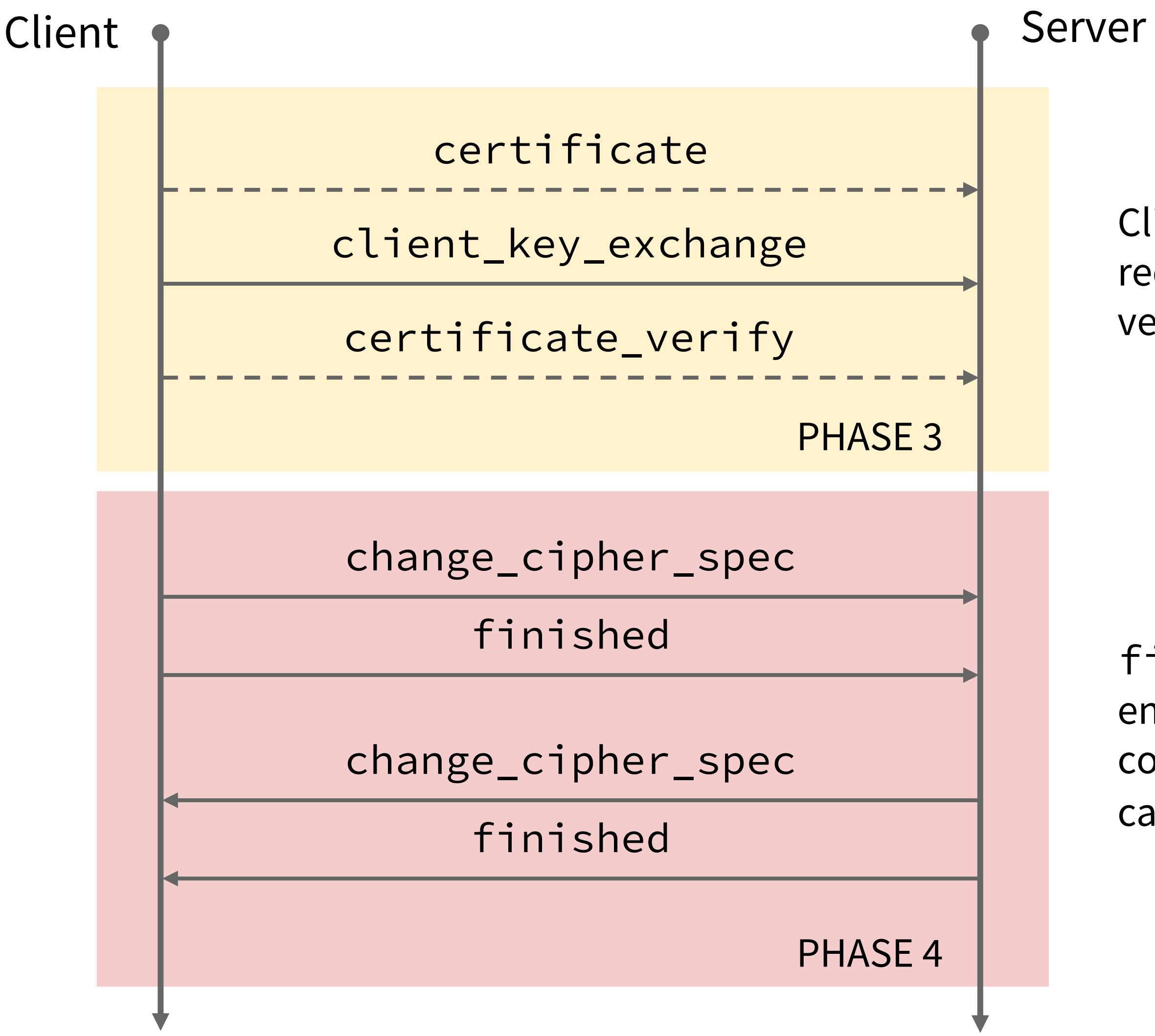


Client sends TLS version, nonce, lists of supported cipher suites and compression methods

Server replies with nonce, chosen cipher suite & chosen compression method

Server can optionally send its certificate, send a temporary public key, request client's certificate

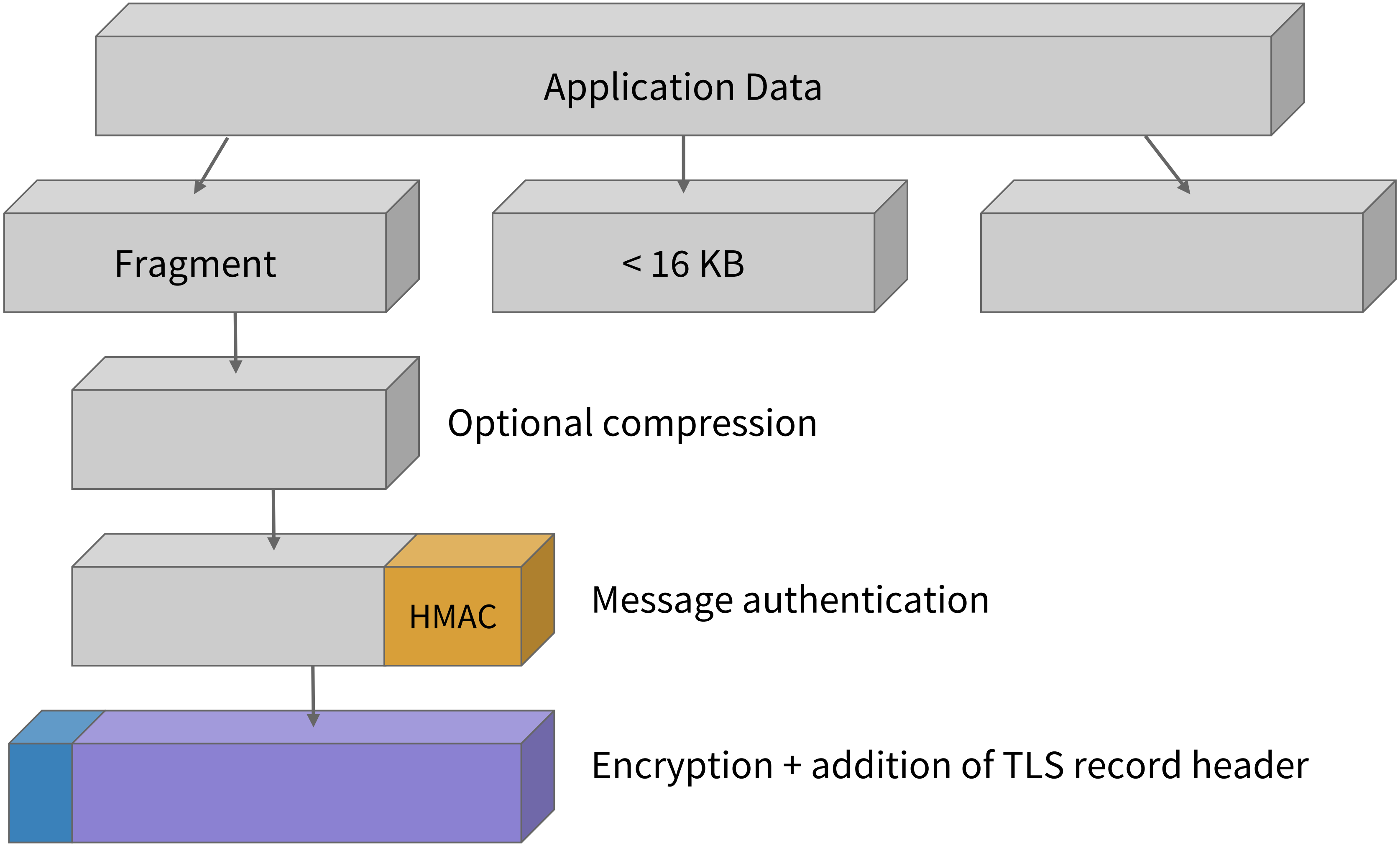
TLS Handshake Protocol



Client sends certificate if required, exchanges key, then verifies its certificate if required

`finished` messages are encrypted and have an HMAC; connection succeeds only if they can be decrypted & verified

TLS Records



Optional compression

Message authentication

Encryption + addition of TLS record header

Problems

- TLS is a complex suite of protocols
- Implementing & configuring them correctly is tricky
- Many examples of things going wrong
    - Android apps (Fahl et al, 2012)
    - Apple’s iOS bug (Feb 2014)
    - Heartbleed (April 2014)
    - DROWN (March 2016)

Android App Problems

- Over 1,000 of 13,000 apps tested in 2012 had serious flaws in their use of SSL
- Android API made it too easy to use SSL incorrectly
  - Trusting all certificates
  - Accept certificates regardless of hostname
  - etc…

Apple’s iOS Bug

```
if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
        goto fail;
if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
        goto fail;
        goto fail;
if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
        goto fail;
```

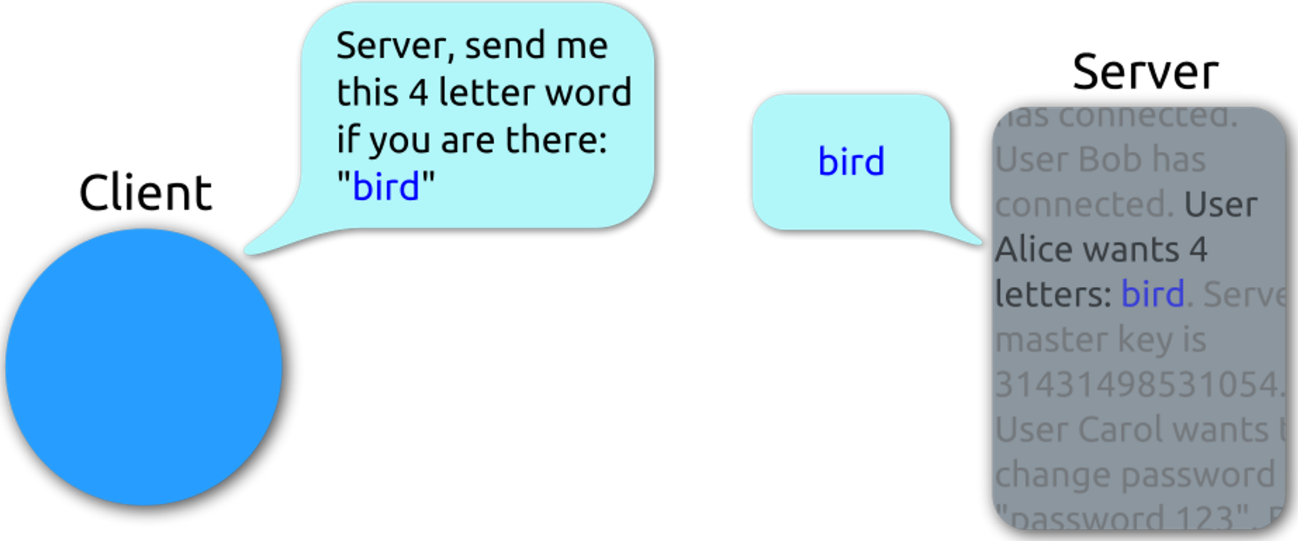
Non-zero value of `err` supposed to trigger a jump to error handler

BUT copied `goto` is not part of the `if` statement
… so will happen even when `err == 0`
… so call to `SSLHashSHA1.final` never happens!

Heartbleed (CVE-2014-0160)

- Serious implementation vulnerability in the OpenSSL library, **discovered in 2014**
- Bug first appearance in **March 2012** release
- Leaks up to 64 KB of server's memory contents
- … which can include secret keys!
- ~66% of websites vulnerable

Heartbeat – Normal usage



Client

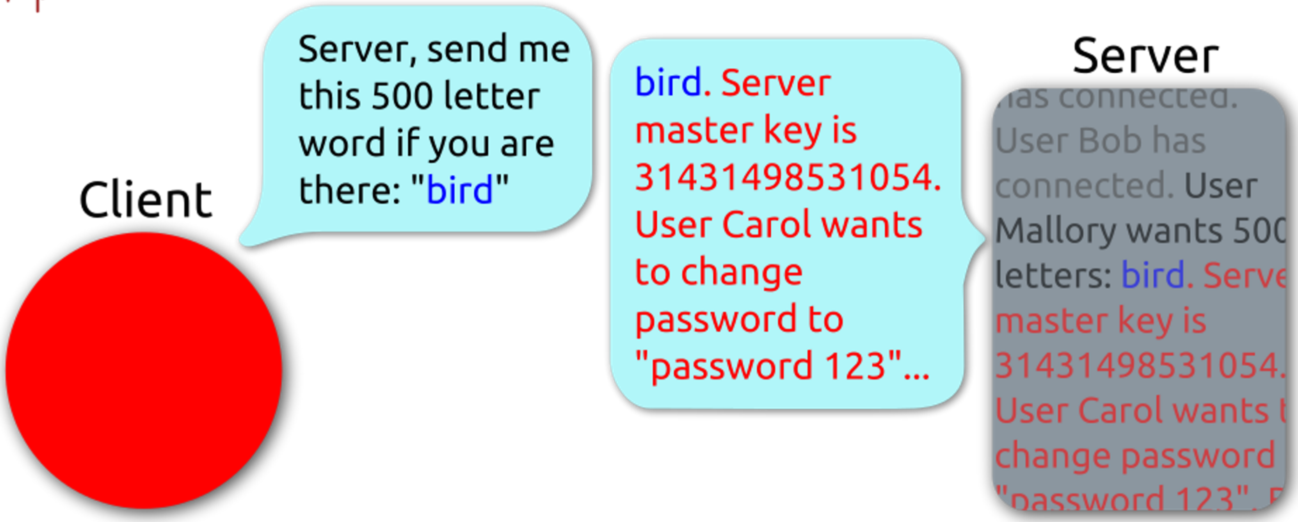
Server, send me this 4 letter word if you are there: "bird"

bird

Server

as connected. User Bob has connected. User Alice wants 4 letters: bird. Serve master key is 31431498531054. User Carol wants t change password "password 123". P

Heartbeat – Malicious usage



Client

Server, send me this 500 letter word if you are there: "bird"

bird. Server master key is 31431498531054. User Carol wants to change password to "password 123"...

Server

as connected. User Bob has connected. User Mallory wants 500 letters: bird. Serve master key is 31431498531054. User Carol wants t change password "password 123". P

(Image by FenixFeather, CC-BY-SA 3.0)

IPsec

- Similar benefits to TLS, but in network layer
- Transparent to users and to applications
    - No need to have distinct secure variants of existing application protocols (HTTPS, etc)
    - No need to change any software when IPsec is implemented in firewall or router
- Can be used as basis of virtual private networks (VPNs)

Elements of IPsec

UNIVERSITY OF LEEDS

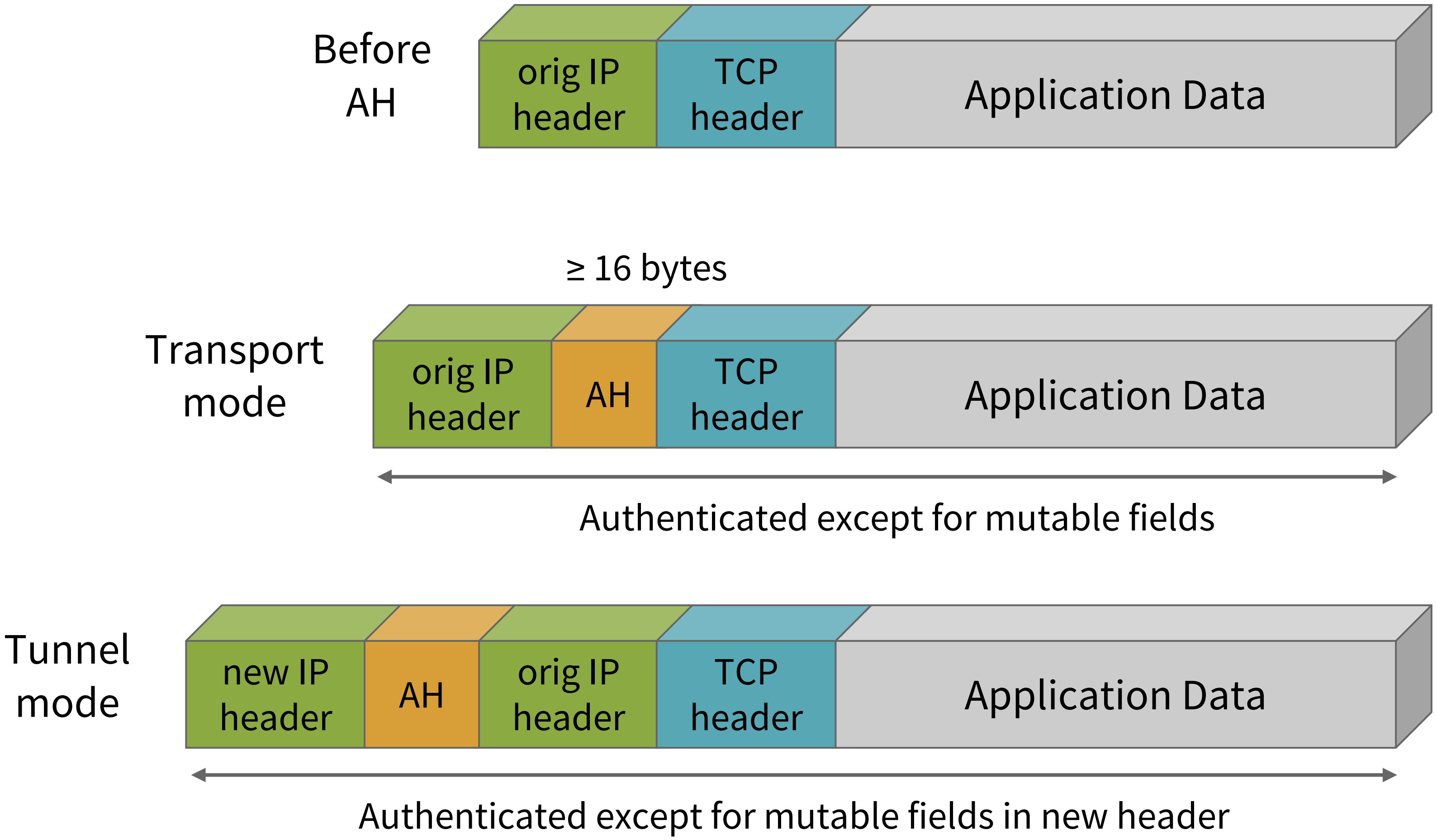
- Two main protocol extensions
  - Authentication Header (AH)
  - Encapsulating Security Payload (ESP)
    - Encryption only
    - Encryption + authentication
- Two modes of operation
  - **Transport mode**: protects upper-layer protocols encapsulated within an IP packet
  - **Tunnel mode**: protects entire packet

Transport & Tunnel Modes

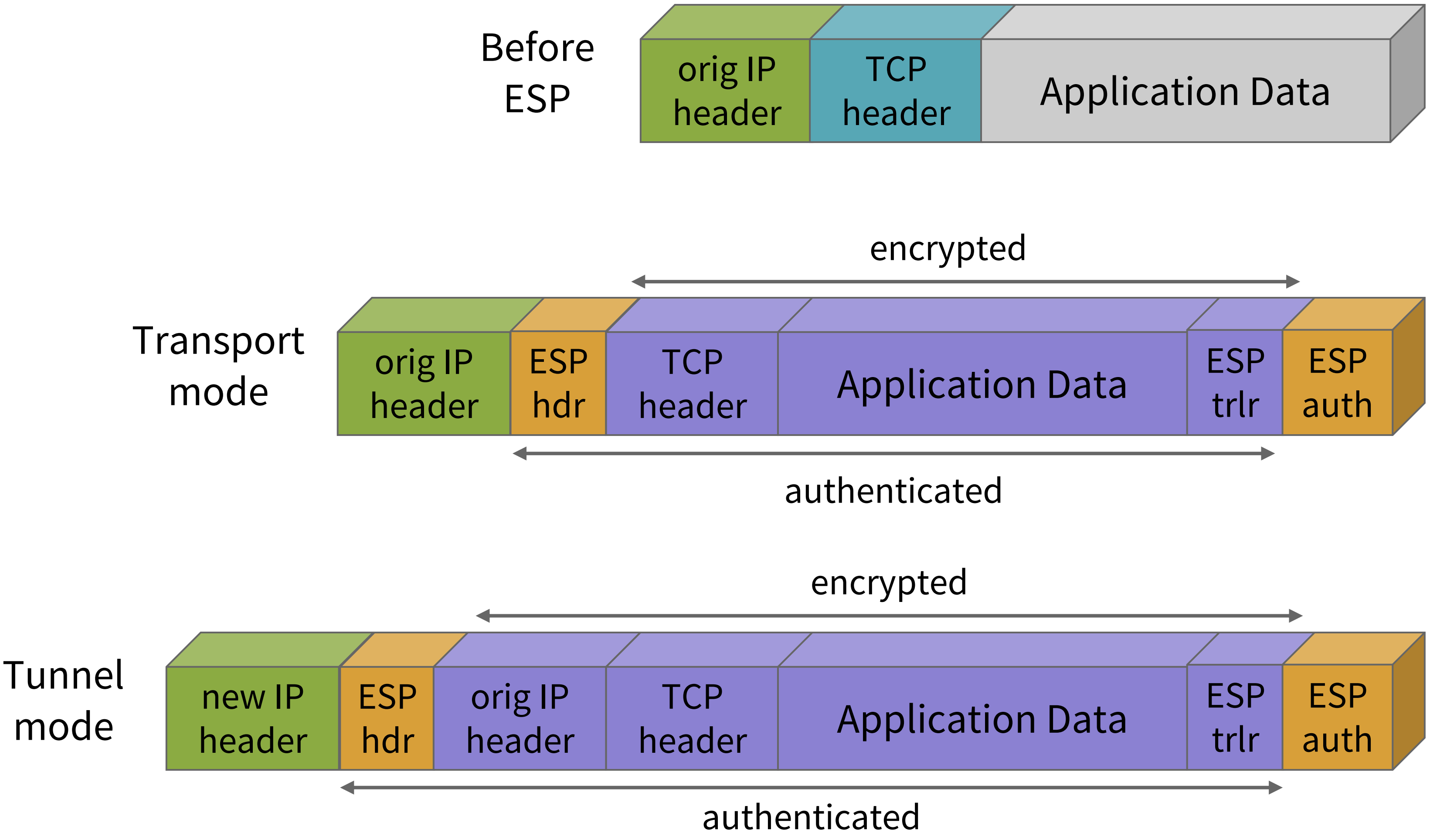
	Transport Mode	Tunnel Mode
AH	Authenticates IP payload + selected portions of header	Authenticates entire inner packet + selected parts of outer IP header
ESP	Encrypts IP payload	Encrypts inner packet
ESP + auth	Encrypts IP payload; authenticates payload, but not IP header	Encrypts & authenticates inner packet

Applying AH

UNIVERSITY OF LEEDS



Applying ESP



Summary

We have

- Discussed the diminishing value of traditional firewalls
- Explored how TLS provides guarantees of confidentiality, integrity and authenticity
- Investigated some of the recent problems discovered in TLS implementations
- Considered how IPsec provides similar benefits to TLS, but in the Internet layer

Follow-Up / Further Reading

- Chapter 5 of Bellovin for more on firewalls
- RFC 5246: the TLS 1.2 specification
- OpenSSL library
- Fahl et al, *Why Eve and Mallory love Android: An Analysis of Android SSL (In)Security* (PDF)
- Recent problems with TLS or implementations:
  - 2014: Heartbleed website and reminiscences from an Amazon engineer
  - 2016: DROWN, Sweet32