HOCHSCHULE
DER MEDIEN

New Security Features in TLS V1.3

Prof. Dr. Roland Schmitz
Stuttgart Media University
Computer Science and Media

Agenda

HOCHSCHULE DER MEDIEN

- TLS Overview
  - TLS Record Protocol
  - TLS Handshake Protocol
- History of TLS and SSL
- Some Recent Attacks on TLS
  - Heartbleed
  - POODLE
  - FREAK
- New Security Features in TLS v1.3
  - ECC-DHE
  - Galois-Counter Mode (GCM)

What is TLS?

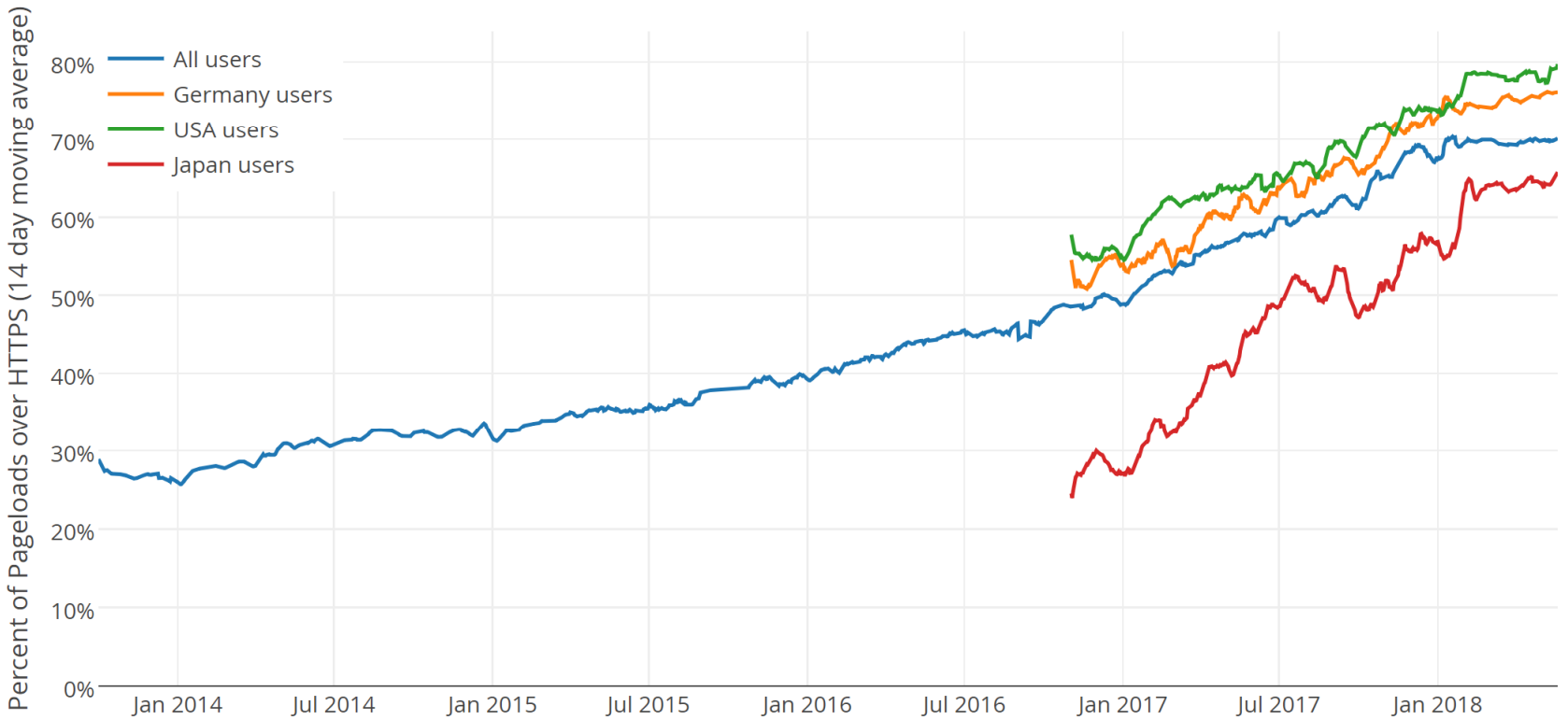
- TLS = Transport Layer Security
- TLS provides a secure (encrypted, authenticated) tunnel on top of TCP between a client and a server
- Mostly used to secure http, transforming it to https
- Major enabler of electronic commerce
  - But can also be used to fulfill DSGVO requirements, e.g. when transferring sensitive data to third parties
- Trust-on-first-use (TOFU) protocol

TLS is growing in popularity

HOCHSCHULE DER MEDIEN

Percentage of Web Pages Loaded by Firefox Using HTTPS

(14-day moving average, source: Firefox Telemetry)

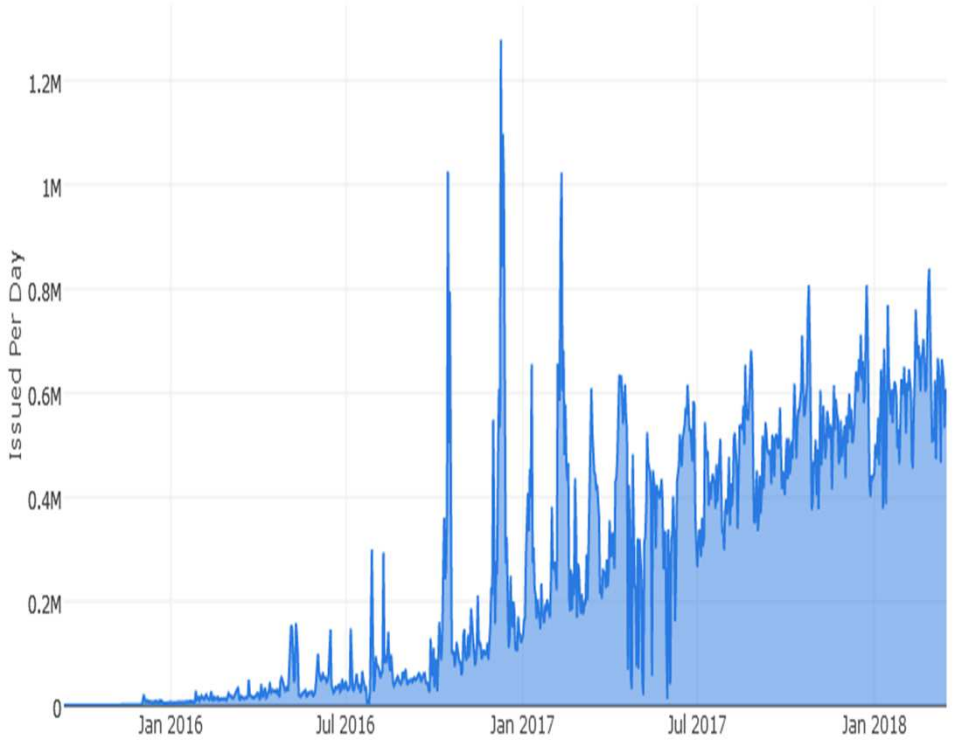


(source: letsencrypt.org)

Reasons for Growing TLS Popularity

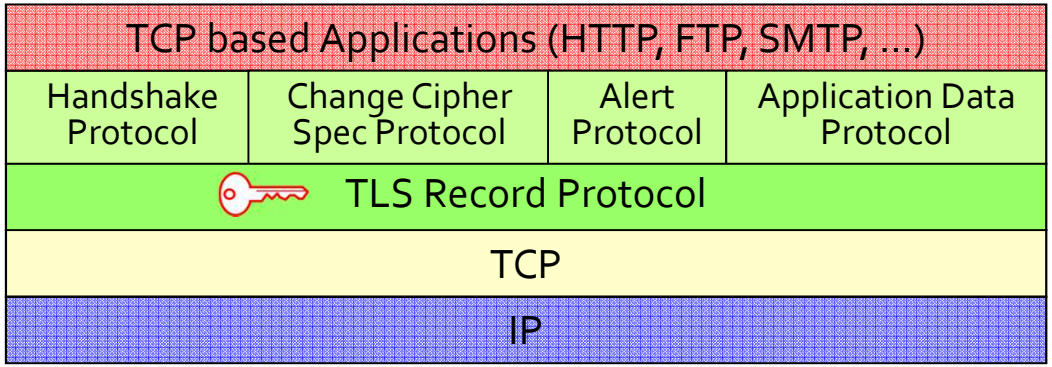
HOCHSCHULE DER MEDIEN

- Edward Snowden
- Google uses https support as ranking factor
- Easier access to certificates:
  - Let's Encrypt has issued more than 46 million active certificates by end of 2017



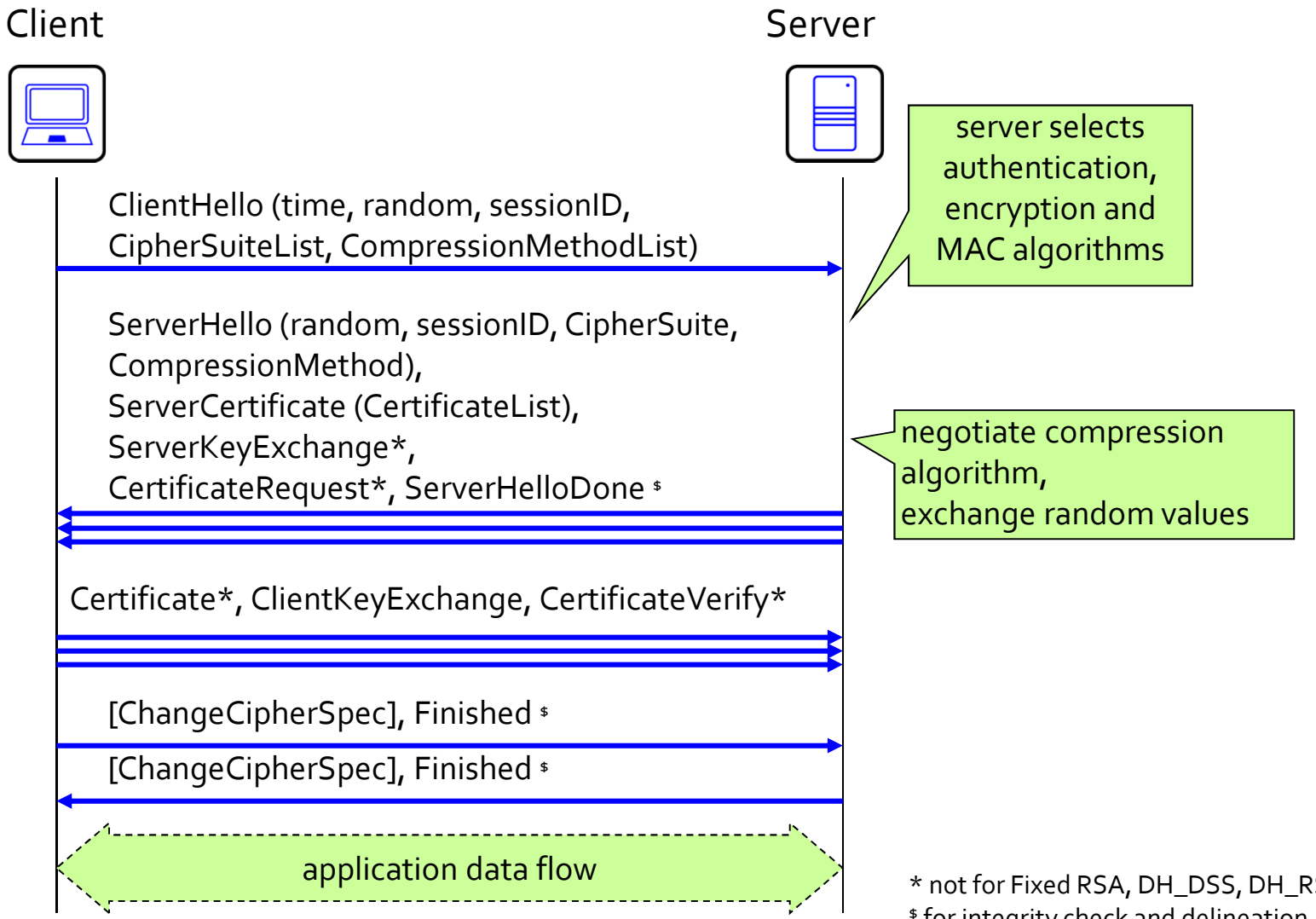
Number of Certificates Issued by Let's Encrypt Per Day (source: letsencrypt.org)

TLS Protocol Family Overview (Pre V1.3)



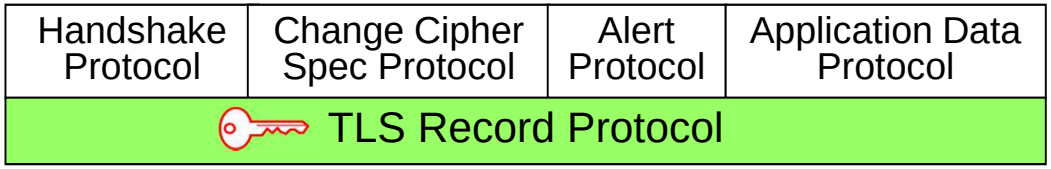
- Handshake Protocol:
  - Client and Server agree on algorithms (Cipher Suites) and keys
  - Server and optional Client authentication based on digital certificates
- Record Protocol
  - Symmetric End-to-End encryption
  - Integrity Protection via symmetric MACs

TLS Handshake Overview (Pre V1.3)

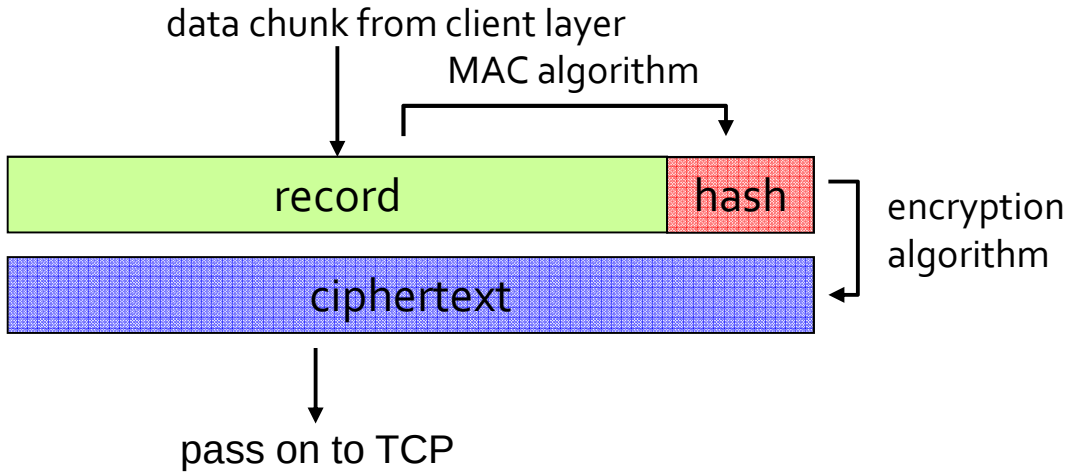


* not for Fixed RSA, DH_DSS, DH_RSA
$ for integrity check and delineation

TLS Record Protocol Overview (Pre V1.2)



- tasks
  - › fragmentation
  - › compression (optional)
  - › Integrity protection and confidentiality
  - › MAC-then-Encrypt:
    - › First, key dependent MAC computation
    - › Then encryption



History of SSL and TLS Overview

~~1994: SSL v1.0~~

March 1995: SSL v2.0

Nov 1995: SSL v3.0

→

1999: TLS v1.0

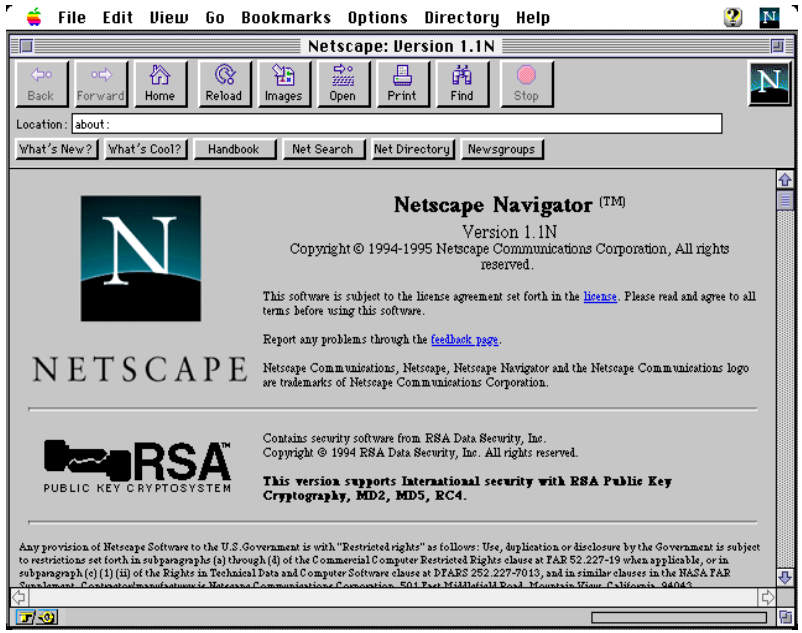
2006: TLS v1.1

2008: TLS v1.2

March 2018: TLS v1.3 Draft submitted to IESG for publication

Detailed History of SSL and TLS

- Until 1994: Netscapes develops SSL. Version 1.0 has serious security issues and is never published
- March 1995: SSL V2.0 shipped in Netscape Navigator V1.1
  - Has unprotected Handshake Messages
- November 1995: SSL V3.0
- January 1999: IETF publishes TLS V1.0 (almost identical to SSL V3.0)
- April 2006: TLS V1.1 addresses the BEAST Attack (but nobody notices)
- August 2008: TLS V1.2 introduces authenticated encryption (AEAD)
- February 2014: Firefox 27 supports TLS V1.2
- October 2014: POODLE Attack on TLS V1.2
- February 2016: Work on TLS V1.3 starts
- The Payment Card Industry Security Standards Council sets 30th June 2018 as deadline for disabling SSLv3.0 / TLS v1.0

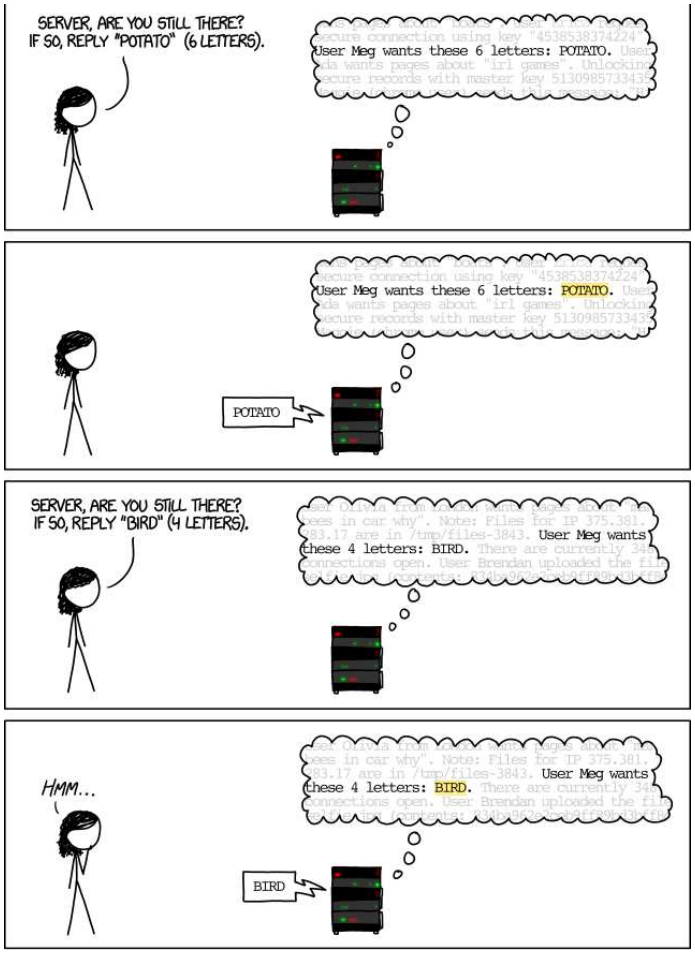


Recent Attacks on TLS

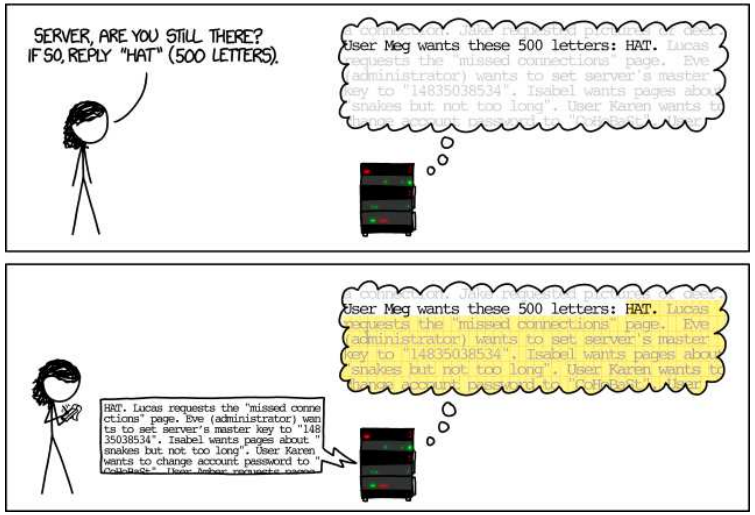
HOCHSCHULE DER MEDIEN

- Cryptographic Attacks:
  - December 2008: MD5 Considered Harmful
  - August 2012: Microsoft starts blocking RSA Keys weaker than 1024 bits
  - March 2013: RC4 dead
  - January 2016: SHA-1 Certificates no longer allowed to be issued
- Implementation Bug:
  - April 2014: Heartbleed Attack
- Protocol Attacks:
  - September 2011: BEAST Attack against TLS v1.0 publicly noticed
  - October 2014: POODLE Attack on TLS V1.2
  - March 2015: FREAK Attack

Heartbleed Attack



https://xkcd.com/1354/



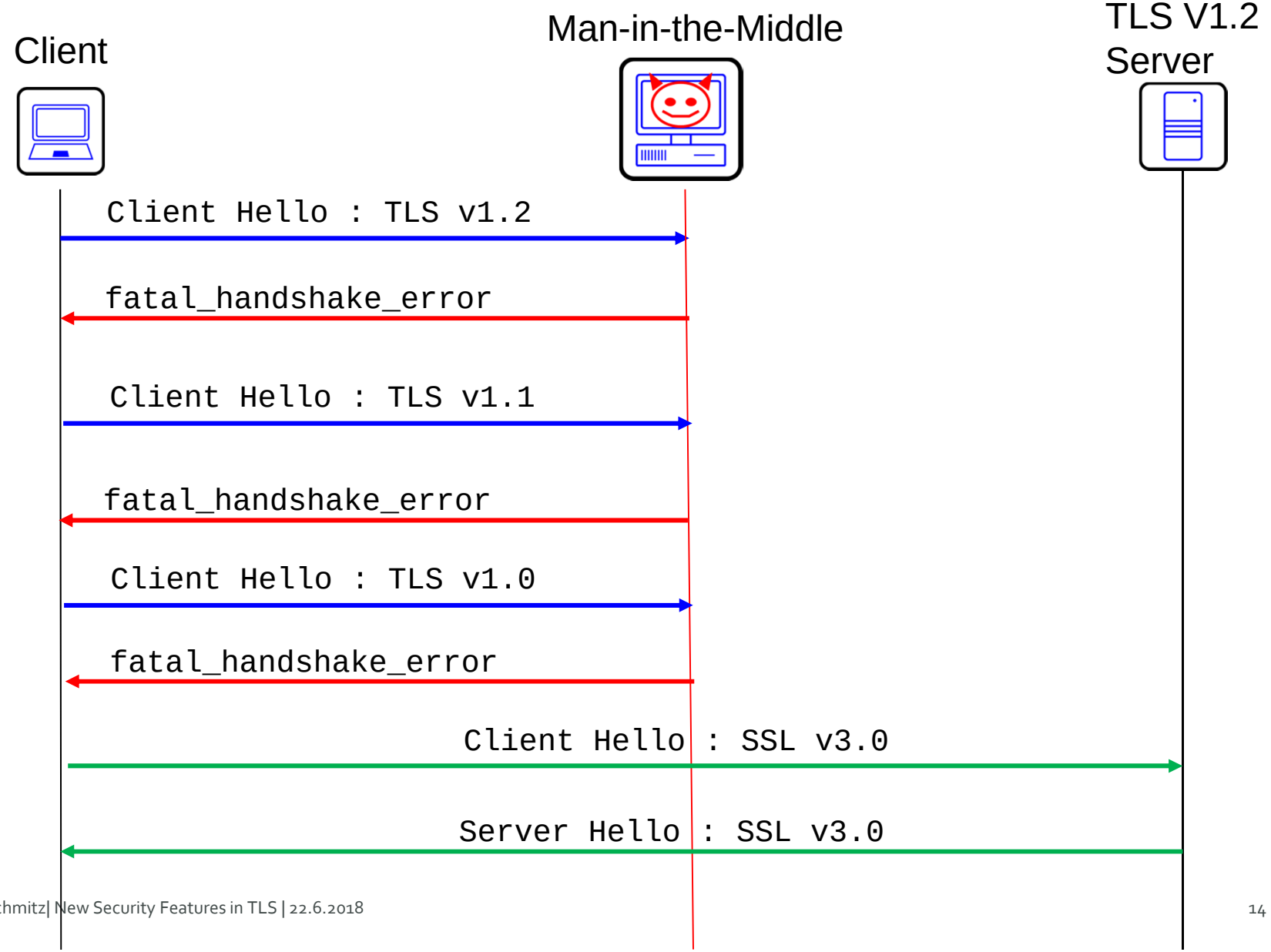
- Buggy Heartbeat code published March 2012 (OpenSSL V1.0.1)
- Heartbleed Attack discovered April 2014

The POODLE Attack of 2014

HOCHSCHULE DER MEDIEN

- Padding Oracle On Downgraded Legacy Encryption
- Possible because:
  - Downgrade Dance to SSL v3.0
  - SSL v3.0 uses CBC-Mode to encrypt and allows arbitrary padding
  - MAC-then-Encrypt order of operations when forming the records leaves padding data unprotected by MAC

The Downgrade Dance



The FREAK Attack of 2015

HOCHSCHULE DER MEDIEN

- Factoring RSA_Export Keys
- Some TLS V1.2 servers still support RSA_Export cipher suites with 512 bit key length (although explicitly deprecated in TLSv1.1)
- A Man-in-the-Middle can trick a client *C* into accepting a weak RSA public key by manipulating the `ClientHello` message
- Attacker factorizes weak RSA key either on the fly or before the attack happens

Reasons for Insecure TLS Configurations

HOCHSCHULE DER MEDIEN

- Main problem: The „Legacy Bloat“:
  - Co-Existence of many versions and ciphersuites which are still supported for backward compatability
  - Administrators tend to sacrifice security for maximum backwards compatibility
- Two quotes from expert security auditors:
  - *„Backward Compatibility is the major showstopper regarding proper TLS configurations.*“
  - Admins are *„afraid of using crypto*“, fearing that strong crypto might lead to incompatibilities.

New Security Features in TLS V1.3

HOCHSCHULE DER MEDIEN

Cited from draft-ietf-tls-tls13-28:

> The list of supported symmetric algorithms has been pruned of all algorithms that are considered legacy. Those that remain all use Authenticated Encryption with Associated Data (AEAD) algorithms.

> Static RSA and Diffie-Hellman cipher suites have been removed; all public-key based key exchange mechanisms now provide forward secrecy.

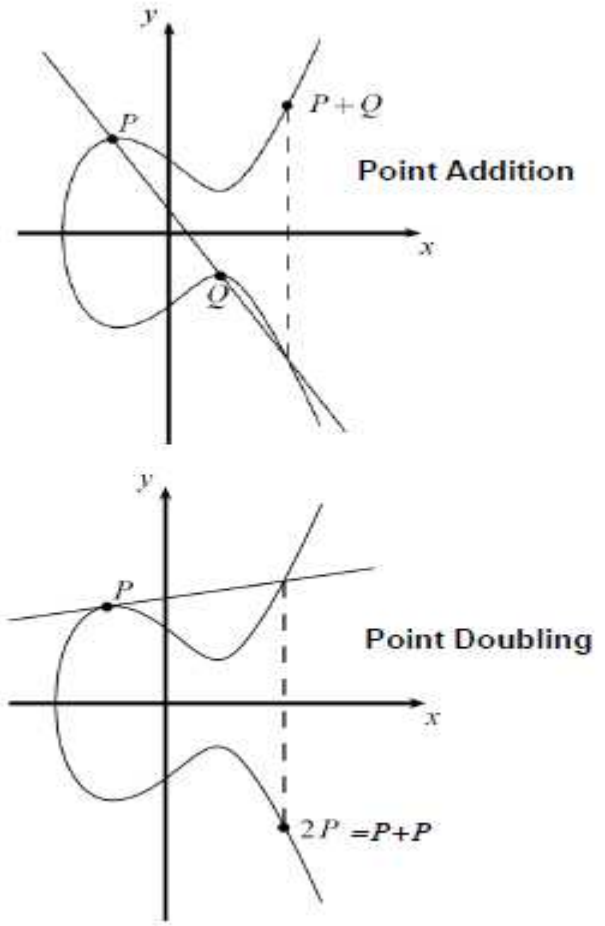
New Security Features in TLS V1.3: No Insecure Algorithm Options

HOCHSCHULE DER MEDIEN

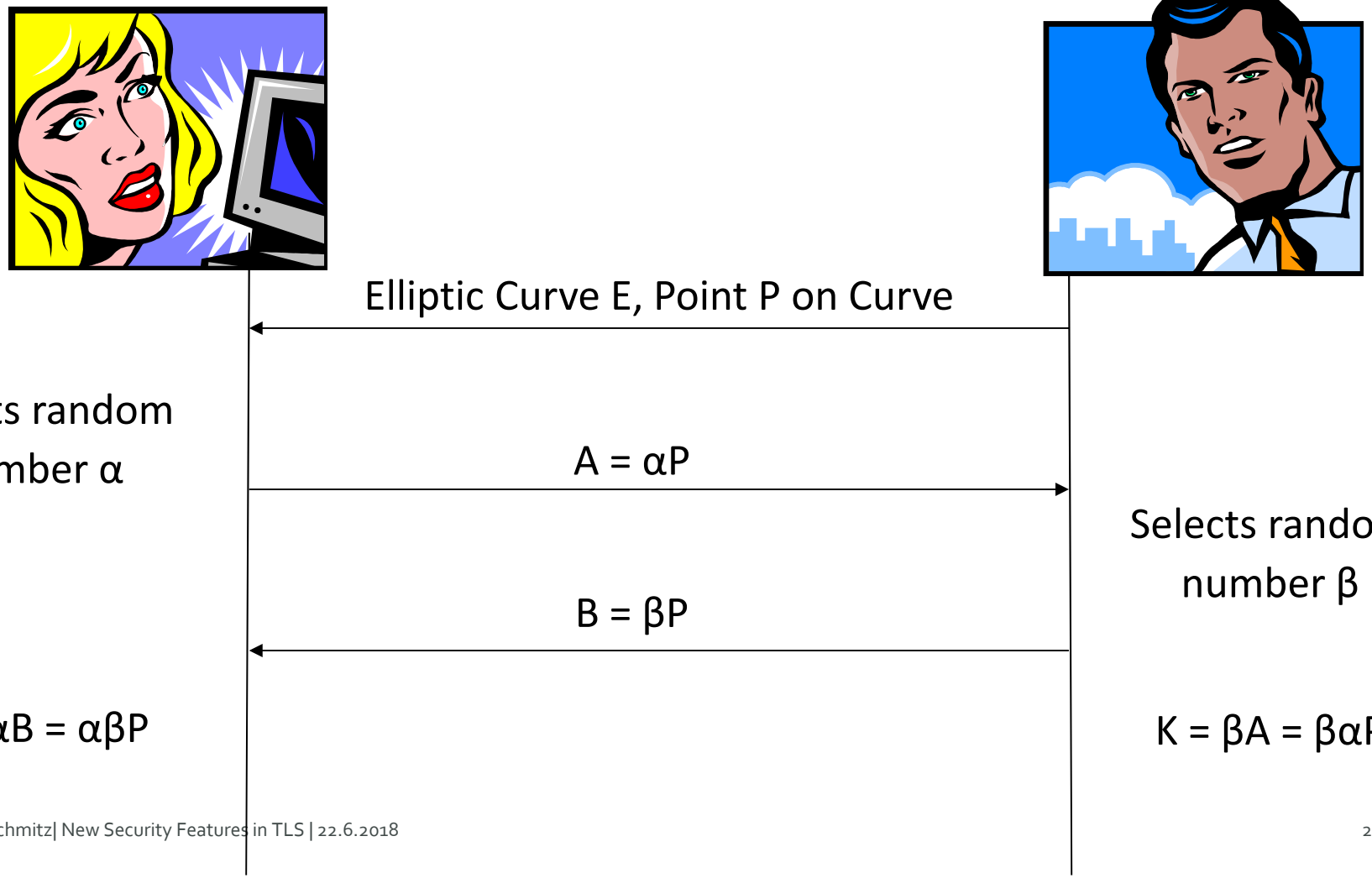
- Get rid of all insecure algorithm options:
  - No export ciphers
  - No RC4
  - No MD5, No SHA-1
  - Use SHA-256
- No Key Exchange via Static RSA or DH Keys
  - Promote Forward Secrecy through ephemeral public keys
  - Three Basic Key Exchange Modes:
    - (EC)DHE
    - (EC)DHE with Pre-Shared Key (PSK)
    - PSK
- Mandatory Authenticated Encryption
  - Use Galois Counter Mode (GCM)

ECDHE

- Elliptic Curve Diffie-Hellman Ephemeral:
  - Use Diffie-Hellman over Elliptic Curves for Key Exchange
  - Use a fresh DH Keypair for each session to provide *forward secrecy*
- DH Keypairs are authenticated by a long-lived keypair or PSK
- If this keypair is lost, only future connections are endangered



Diffie – Hellman Key Exchange over Elliptic Curves



Elliptic Curve
Diffie-Hellman Key Exchange Toy Example

- Public Parameters:

$$E : y^2 = x^3 + x + 1 \bmod 97$$

$$P = (95, 66)$$

- Alice's Secret Key:

$$\alpha = 16$$

- Alice sends to Bob:

$$A = \alpha P = (82, 54)$$

- Bob's Secret Key:

$$\beta = 94$$

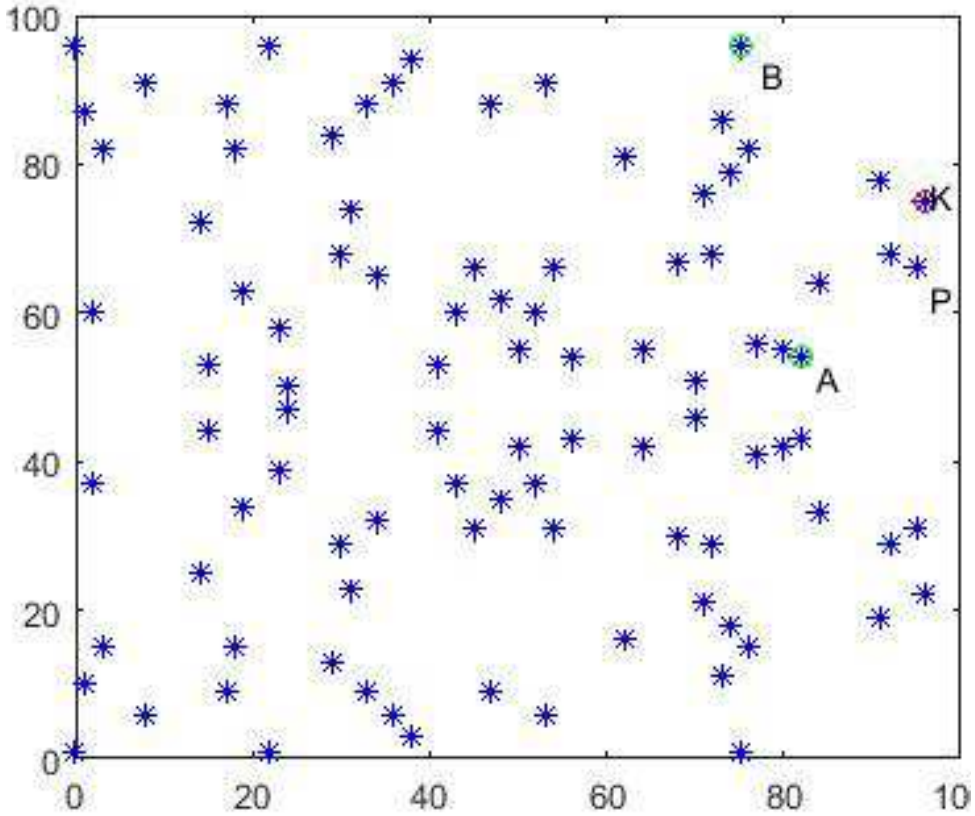
- Bob sends to Alice:

$$B = \beta P = (75, 96)$$

- Alice computes:

$$K_{Alice} = \alpha B = (96, 75)$$

- Bob computes: $K_{Bob} = \beta A = (96, 75)$

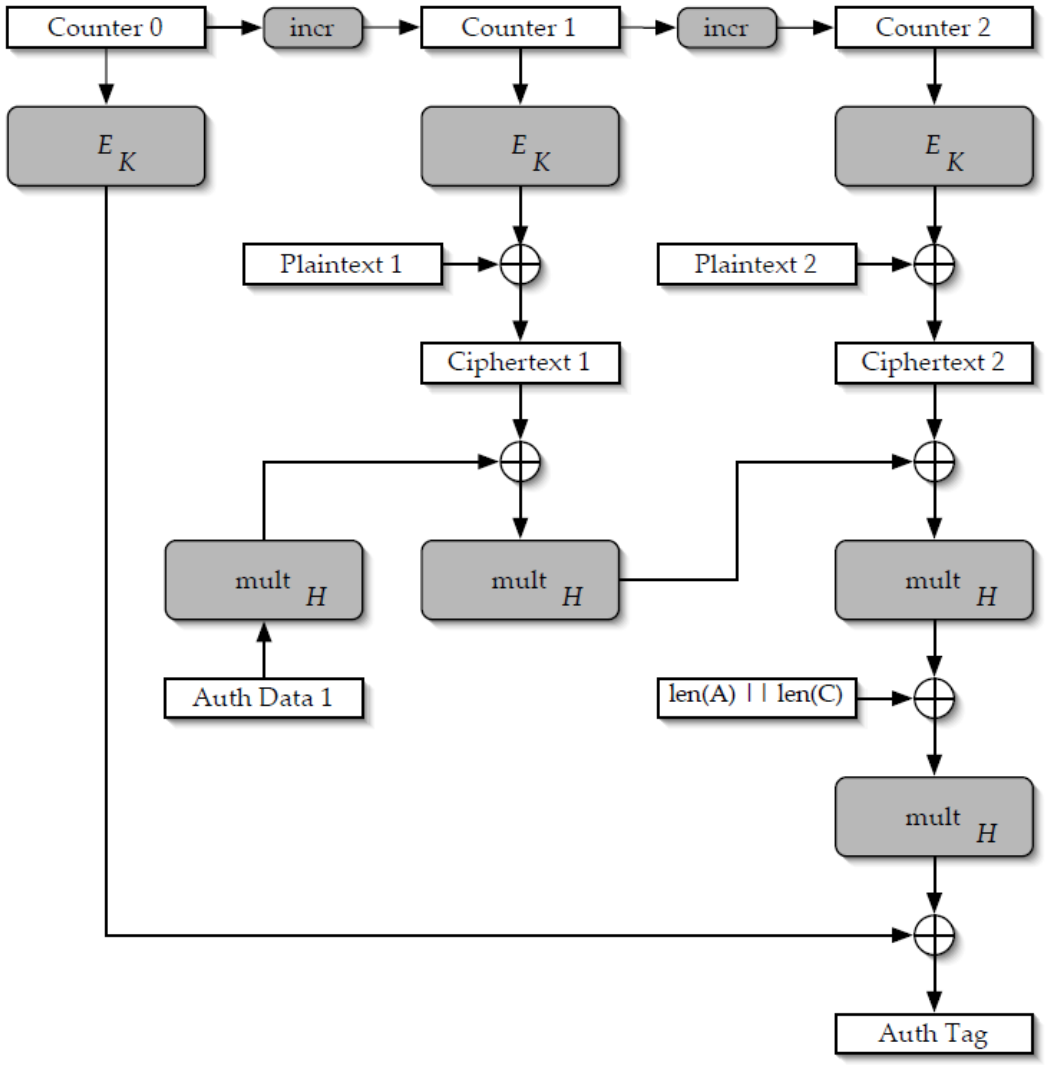


Elliptic Curve Crypto Security Level in Bits

HOCHSCHULE DER MEDIEN

Symmetric Key	RSA Key	ECDH Key
80	1024	160
112	2048	224
128	3072	256
192	7936	384
256	15360	512

Authenticated Encryption: The Galois Counter Mode (GCM)

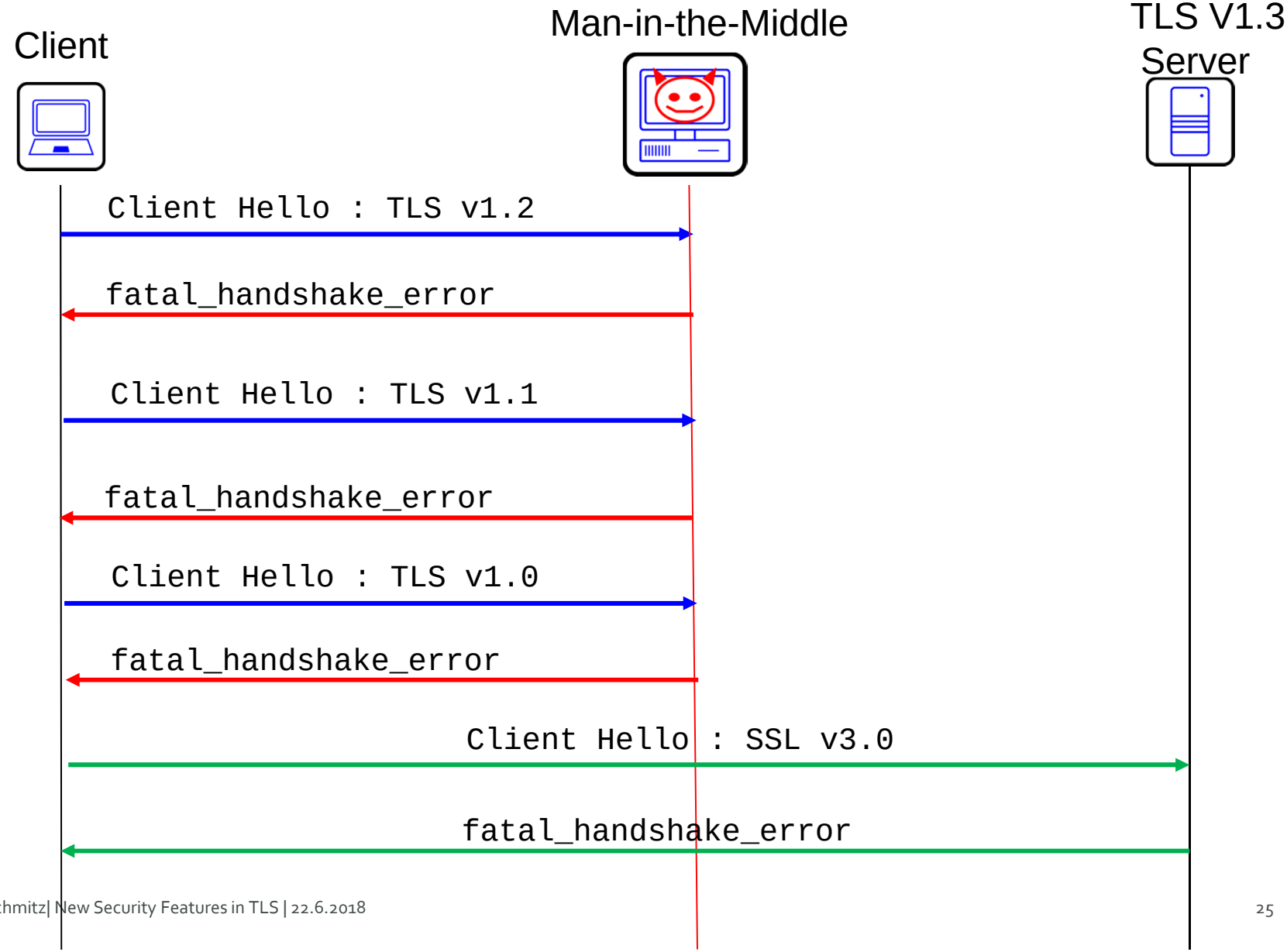


New Security Features in TLS V1.3: Protocol Modifications

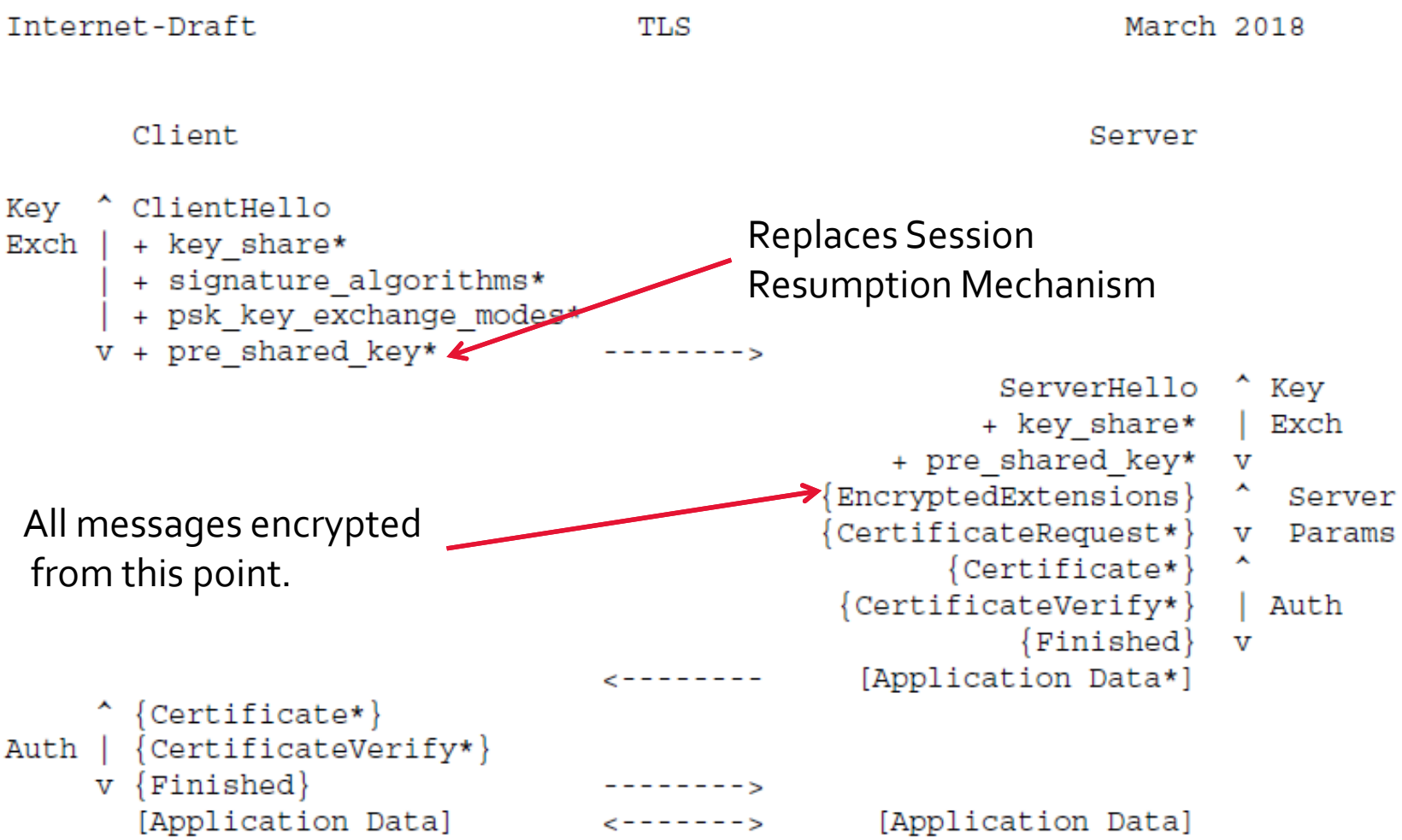
HOCHSCHULE DER MEDIEN

- All handshake messages after the ServerHello are now encrypted.
- SSL negotiation is prohibited
- Downgrade attempts are signalized by servers
  - If negotiating TLS 1.2 or lower, TLS 1.3 servers MUST set the last seven bytes of their random value to the bytes: 44 4F 57 4E 47 52 44 (ASCII for DOWNGRD)
  - `ServerKeyExchange` Messages include a signature over random values
  - TLS 1.3 clients receiving a `ServerHello` indicating TLS 1.2 or below MUST check that the last seven bytes are not equal to these values.

Sorry, No Downgrade Dance with TLS V1.3



TLS V1.3 Handshake Overview



References

HOCHSCHULE DER MEDIEN

- TLS History:
  https://www.feistyduck.com/ssl-tls-and-pki-history/
- POODLE:
  https://www.openssl.org/~bodo/ssl-poodle.pdf
- FREAK:
  B. Beurdouche et al.: A Messy State of the Union: Taming the Composite State Machines of TLS, 2015 IEEE Symposium on Security and Privacy
- GCM:
  https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf
- TLS Usability:
  K. Krombholz et al.: I Have No Idea What I'm Doing: On the Usability of Deploying HTTPS, 2017 USENIX Security Symposium
- TLS V1.3 Specification:
  https://tools.ietf.org/pdf/draft-ietf-tls-tls13-28.pdf

HOCHSCHULE
DER MEDIEN

Thank You for Your Attention!