Inspecting TLS

Frans Schippers

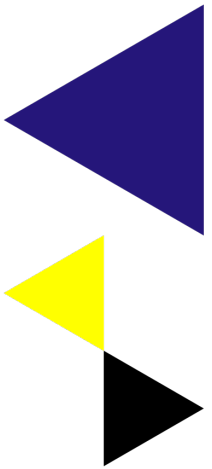
Hogeschool van Amsterdam
Amsterdam University of Applied Sciences
Docent / Onderzoeker

f.h.schippers@hva.nl
frans@xsupport.nl

Creating Tomorrow

Inspecting TLS

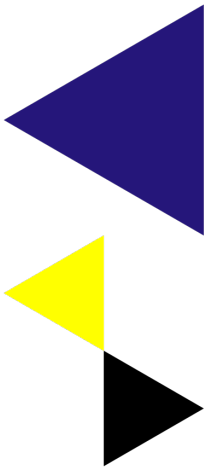
- Encryption (TLS) is everywhere
- The need for Inspection
- Implementation of Inspection
- Demonstration
- Next steps
- Conclusion



Encryption everywhere

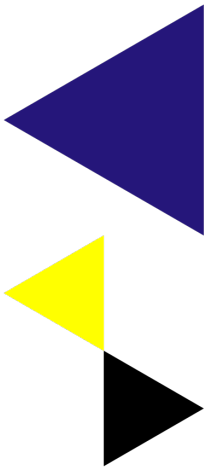
- Internet traffic is encrypted:
  - 95-99% [google.com]

- Application
  - Web-pages (HTTPS)
  - Mail SMTP / POP / IMAP
  - VPN
  - Applications (TLS/HTTPS)
    - API (Strongly Advised)



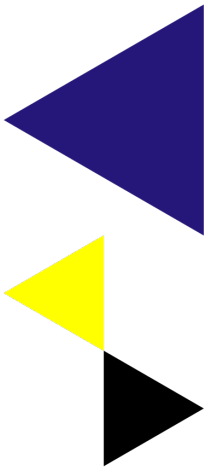
Solid Grounds for Encryption

- Confidentiality
  - Protection against disclosure by third parties
- Integrity
  - Protection against tampering
- Non-Repudiation
  - Proof of action



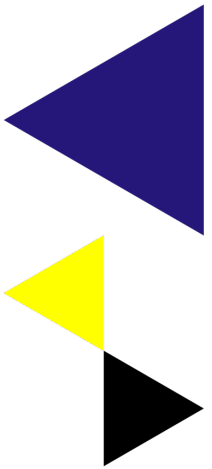
The need for Inspection

- What does an Application send?
  - Some applications disclose (in layman's terms)
  - Some applications don't disclose
  - Some people don't care
  - Some people really do care
- Most of the time you have to surrender
  - No consent no service
  - Coveted service
    - Enforced acceptance Terms of Use.



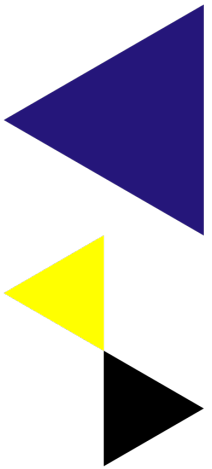
Information Transferred

- Medical (Vitals, Fitness and Health apps)
- Location (Traffic apps, Tracking apps)
- Financial Information (Banking apps)
- Purchases
- Music / Movies / Game Interests
- Social Media (Location, Photo's, Chats)



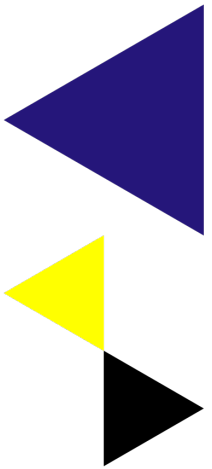
Information is Protected

- Third Party
  - Encryption
    - Protection towards third parties
    - Third parties can't read the encrypted information exchange
- First party (you or your organisation)
  - No clue what is transferred
- Second party (service provider)
  - Trustworthy
    - Reputation, Transparency
  - Full disclosure



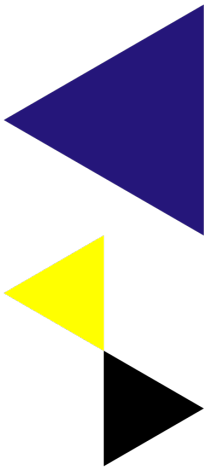
Trust must be earned

- Transparency
  - Insight in the information flow
  - Inspecting the information flow
  - Too hard for one person to inspect
  - Need for an inspecting party (auditing)



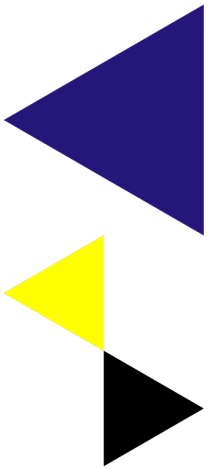
Ownership of Data

- Person
- Personal Data (name, date of birth, color eyes, …)
- Assigned Data (BSN, Employee ID)
- Measured Data (Blood presure, location, ..) by a App
- Stored Data (Pictures in the Cloud)
- Processed Data (Health Status), added value
- Collected Data (Anomized)
- Concluded Data



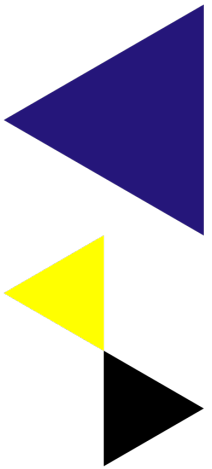
GDPR

- Lawfulness, fairness and transparency
- Purpose limitation
- Data minimisation
- Accuracy
- Storage limitation
- Integrity and confidentiality
- Right of access, correction and deletion and data portability
- Responsibility



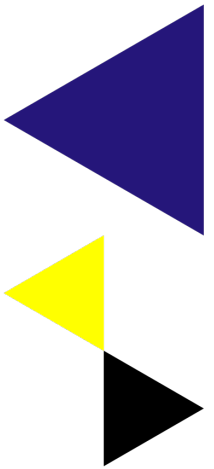
Problems we Face

- TLS encryption
  - How can we allow inspection
  - How can we enable inspection
  - How do we inspect
- Interpretation of the encrypted stream
  - Protocol / Data format
- Willingness to disclose
    - Trade secrets
    - Exposure of flaws
    - Non-consent information



TLS Handshake

- Explain the TLS1.2 & TLS1.3 handshake
  - Overview initial handshake
- Shared Secret Generation
  - Generation (oa. **x25519**)
  - Without revealing the Shared Secret
- Shared Secret Use
  - Basis for the rolling encryption keys
  - **Basis for inspection**



Client

TLS 1.2

Server

ClientHello
* TLSVersion
* ClientRandomData
* Supported Ciphers & Compression
* Extensions
** ServerNameIndicator (SNI)

ServerHello
* TLSVersion
* ServerRandomData
* Chosen Ciphers & Compression
* Extensions

Certificate (SNI)
* Certificate(s)

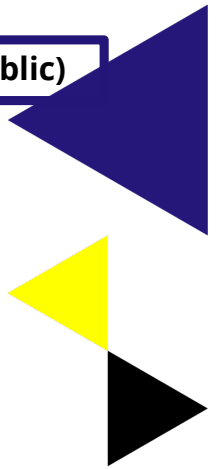
Calculate Server Keys (Private/Public)

ServerKeyExchange
* CurveInfo
* Public ServerKey
* Signature

ServerHelloDone

Calculate Client Keys (Private/Public)

ClientKeyExchange
* Public ClientKey



Client

TLS 1.2

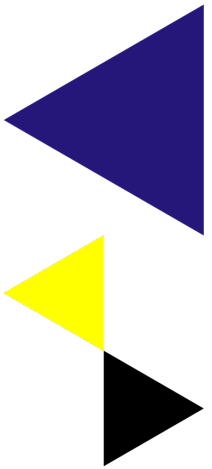
Server

Client Encryption Key Calculation

Server Encryption Key calculation

ClientChangeCipherSpec

ClientHandshakeFinished
* VerifyData [Encrypted]



ServerChangeCipherSpec

ServerHandshakeFinished
* VerifyData [Encrypted]

Client Master Key Calculation

Shared Secret

Server Master Key Calculation

Client

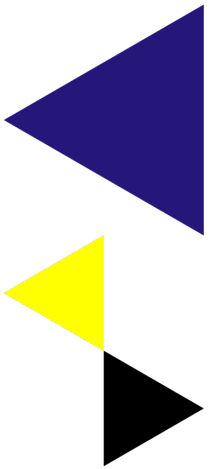
Shared Secret

Server

InspectorInformation
* InspectorCertificate
* Shared Secret (Encrypted)

ApplicationData
* Payload [Encrypted]

ApplicationData
* Payload [Encrypted]



Client

Calculate Client Keys (Private/Public)

ClientHello
* TLSVersion
* ClientRandomData
* Supported Ciphers & Compression
* Extensions
** ServerNameIndicator (SNI)
** KeyShare

Client Master Key Calculation

ClientChangeCipherSpec

ClientHandshakeFinished

TLS 1.3

Shared Secret

Server

Calculate Server Keys (Private/Public)

ServerHello
* TLSVersion
* ServerRandomData
* Chosen Ciphers & Compression
* Extensions
** KeyShare

Server Master Key Calculation

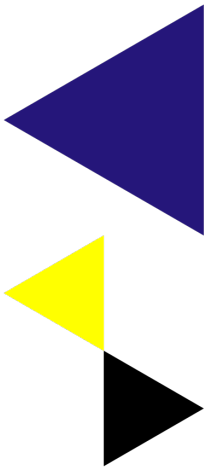
ServerChangeCipherSpec

Excrypted Extensions

Certificate (SNI)
* Certificate(s)

Certificate Verify
* Signature

ServerHandshakeFinished



Client

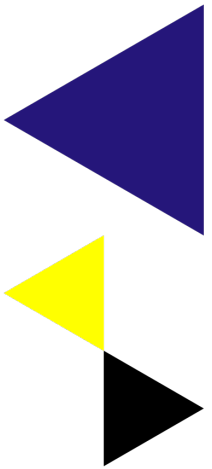
Shared Secret

Server

InspectorInformation
* InspectorCertificate
* Shared Secret (Encrypted)

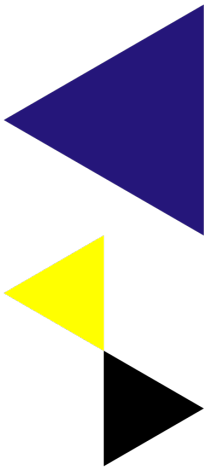
ApplicationData
* Payload [Encrypted]

ApplicationData
* Payload [Encrypted]



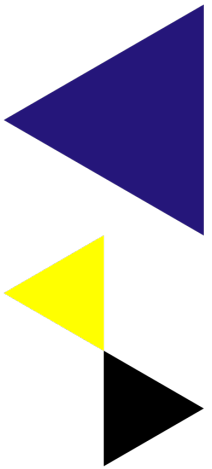
Solid Grounds for Encryption

- Confidentiality
  - Protection against disclosure by third parties
- Integrity
  - Protection against tampering
- Non-Repudiation
  - Proof of action



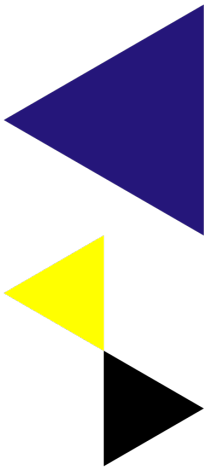
Inspector Information Message

- Created by either side (usually client)
- Certificate(s) to indicate inspectors
- Shared Secret encrypted with Inspectors Public Key(s)
- Key renegotiation
  - Another Inspector Information Message



Capturing

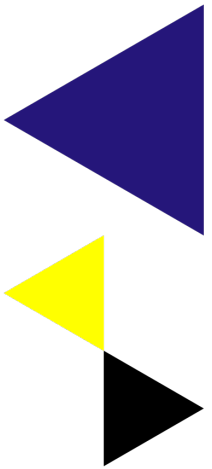
- Application
  - Export Message Capture
    - Saved as file
    - Send to a repository
- External
  - Capture Traffic (Packet Capture)
    - Locally (tcpdump)
    - WiFi-network
    - Tap



Demo

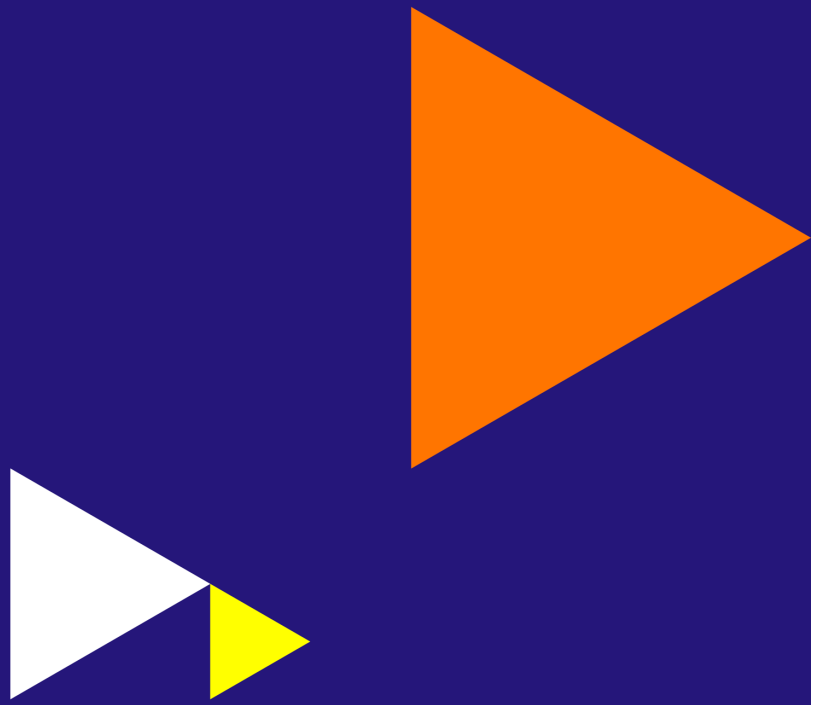
Connection:

- Server: Apache 2.4.56 with LetsEncrypt Certificate
- XsCurl: Implementation of TLS with exported shared key
- Capture: tcpdump
- XsSniffer: Using SharedKey to reveal transported data



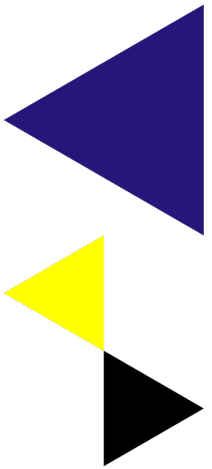
- The Illustrated TLS 1.3 Connection Michael Driscoll
  https://tls13.xargs.org

Demo



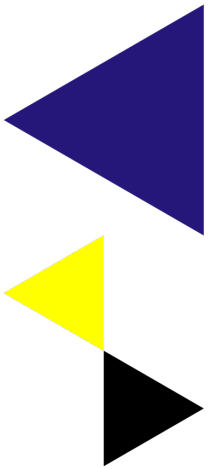
Tshark packets

```
  1   0.000000 TCP 92.66.213.1:50301 → 92.66.213.44:443  [SYN, ECN, CWR]
  2   0.000240 TCP 92.66.213.44:443  → 92.66.213.1:50310 [SYN, ACK, ECN]
  3   0.000282 TCP 92.66.213.1:50301 → 92.66.213.44:443  [ACK]
  4   1.024141 TCP 92.66.213.1:5031  → 92.66.213.44:443  TLSv1.2 ClientHello
  6   1.028519 TCP 92.66.213.44:443  → 92.66.213.1:50310 TLSv1.2 ServerHello
 10   1.071762 TCP 92.66.213.44:443  → 92.66.213.1:50310 TLSv1.2 Certificate,ServerKeyExchange,ServerHelloDone
 13   1.707675 TCP 92.66.213.1:50310 → 92.66.213.44:443  TLSv1.2 ClientKeyExchange
 15   2.213536 TCP 92.66.213.1:50310 → 92.66.213.44:443  TLSv1.2 ChangeCipherSpec
 17   2.724740 TCP 92.66.213.1:50310 → 92.66.213.44:443  TLSv1.2 EncryptedHandshakeMessage
 19   2.725644 TCP 92.66.213.44:443  → 92.66.213.1:50310 TLSv1.2 ChangeCipherSpec, EncryptedHandshakeMessage
 21   3.238150 TCP 92.66.213.1:50310 → 92.66.213.44:443  TLSv1.2 ApplicationData
 23   3.746594 TCP 92.66.213.1:50310 → 92.66.213.44:443  TLSv1.2 EncryptedAlert
 25   3.790795 TCP 92.66.213.44:443  → 92.66.213.1:50310 TLSv1.2 ApplicationData
 26   3.790891 TCP 92.66.213.44:443  → 92.66.213.1:50310 TLSv1.2 ApplicationData
 29   3.792075 TCP 92.66.213.44:443  → 92.66.213.1:50310 TLSv1.2 EncryptedAlert
 30   3.792078 TCP 92.66.213.44:443  → 92.66.213.1:50310 [FIN, ACK]
 33   4.247213 TCP 92.66.213.1:50310 → 92.66.213.44:4423 [FIN, ACK]
```



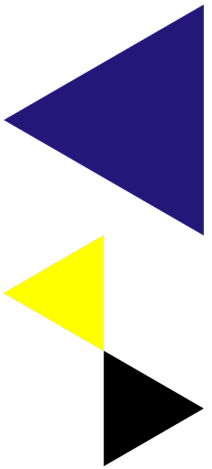
Handshake

```
>c> TlsClientHello
<s< TlsServerHello
<s< TlsServerCertificate
<s< TlsServerKeyExchange
<s< TlsServerHelloDone
>c> TlsClientKeyExchange
>c> TlsChangeCipherSpec
>c> TlsClientHandshakeFinished
*** Client Handshake Finished
<s< TlsChangeCipherSpec
<s< TlsServerHandshakeFinished
*** Server Handshake Finished
*** End of Handshake
```

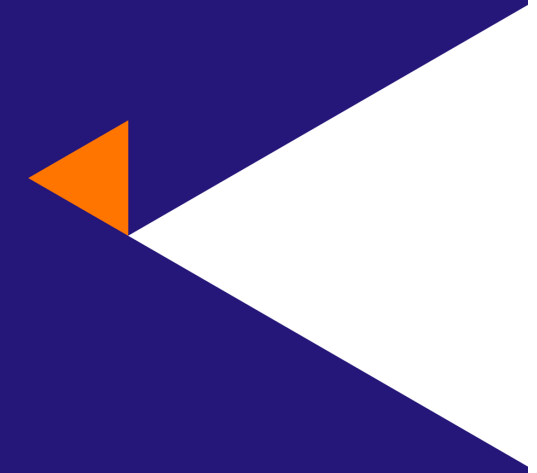


Handshake

```
>c> ApplicationData ...
Srvr:data b'GET /api/v1.0/test HTTP/1.1\r\n
     Host: sjxs.cs-hva.nl:443\r\n
     User-Agent: curl/7.64.1\r\n
     Accept: */*\r\n\r\n'
<s< ApplicationData ...
Clnt:data b'HTTP/1.1 200 OK\r\n
     Date: Thu, 22 Sep 2022 15:18:46 GMT\r\n
     Server: Apache/2.4.51 (Raspbian)\r\n
     Vary: Accept-Encoding\r\n
     Transfer-Encoding: chunked\r\nContent-Type: text/html\r\n\r\n
     6b\r\n<html>\n
       <body>\n
         <title>Test page </title>\n
           <br>Date: 2022-09-22 16:18:46.996355\n
       </body>\n
     </html>\n\r\n'
```

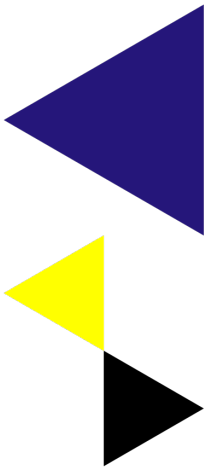


End of Demo



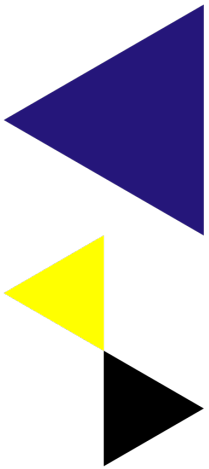
Next steps

- Advocating the need for Inspection
  - ISO/IEC 27071
    - Security recommendations for establishing trusted connections between devices and services' between devices and services
- Replaceable libraries of OpenSSL
  - Support for message capture
  - Exporting Inspector Information Message
  - Inserting Inspector Information Message into the TLS-stream
- Pitching the idea for next generation TLS



Conclusion

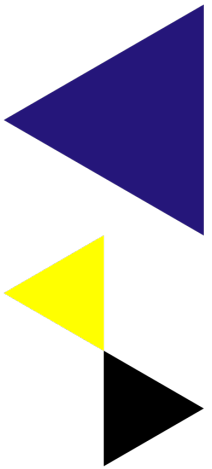
- People have the right to inspect (GDPR)
  - They own the data
  - They own the right to inspect the Information Flow
- To Inspect we need
  - Shared key
  - Captured transmission
  - Safely outsource ability to an inspector / auditor
- Limited time
  - Every session has a new shared key
  - Every session needs a unique Inspector Information Message



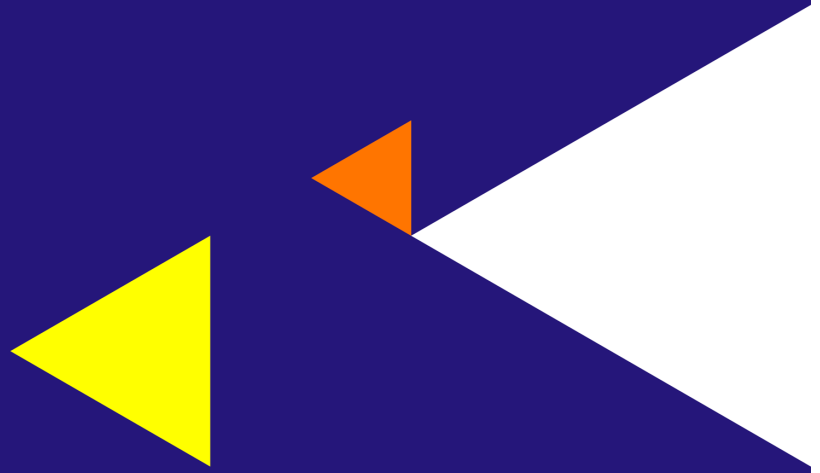
Questions

Contact:

Frans H. Schippers
HvA/AUAS HBO-ICT Cyber Security
f.h.schippers@hva.nl



Bedankt!



Creating Tomorrow