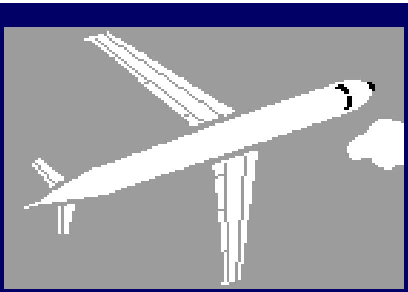
TEE:
TLS Authentication Using EAP *draft-nir-tee-pm-00.txt*

Yoav Nir
Yaron Sheffer
Hannes Tschofenig

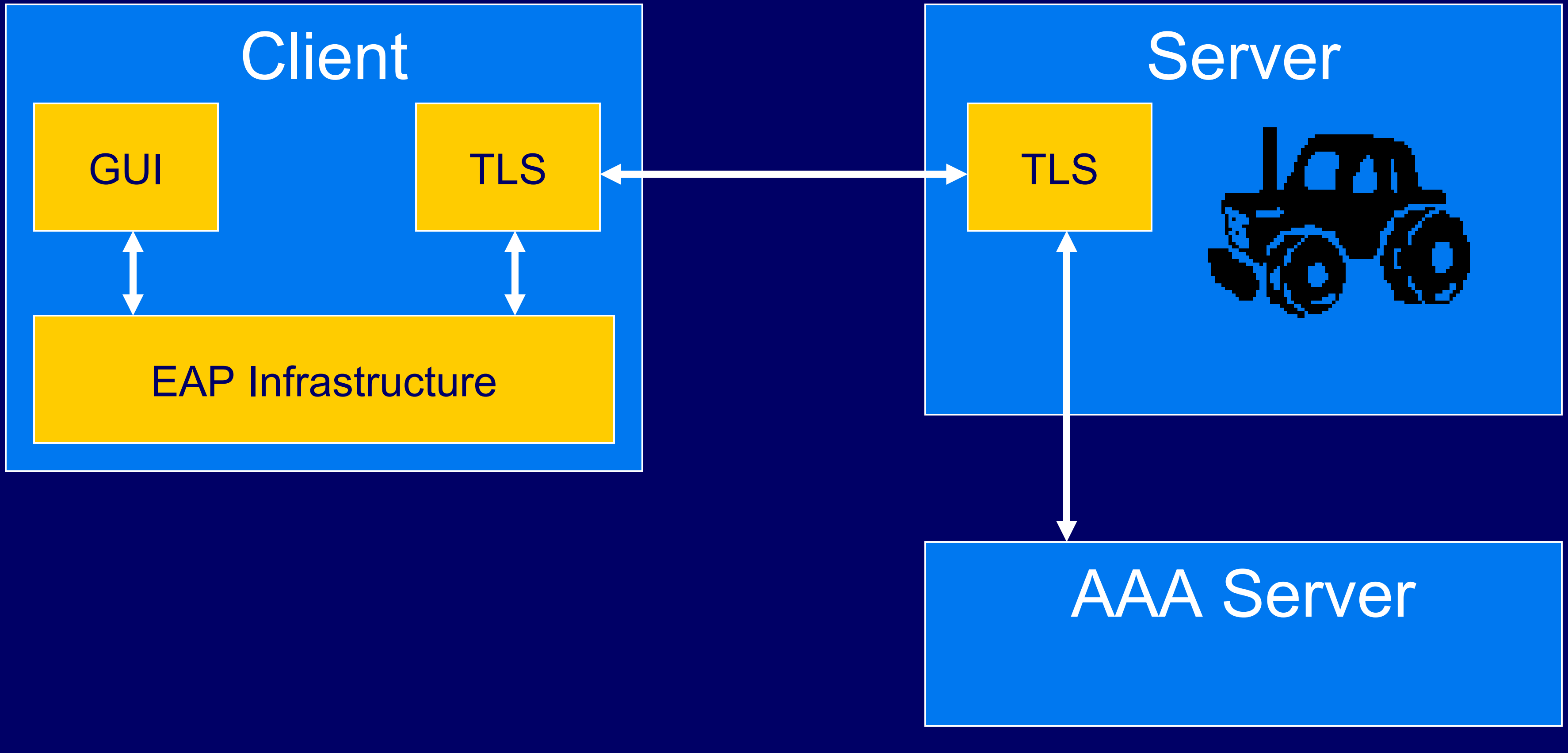
IETF-68, Prague, March 2007

Goals



- Include protocol-level authentication in TLS
  - To eliminate many uses of application-level authentication
  - To provide improved binding between TLS and the authentication
- Fact: EAP is heavily used for authentication
  - PPP, 802.1X, 802.11i (WPA), IKEv2 and more
  - Supported by all AAA servers
  - "The only secure way for password authentication"
- Reuse EAP capabilities available on hosts
  - Specifically, EAP-SIM for hands-free auth
- Have a good acronym
  - TEE: TLS-EAP Extension

Architecture



Client

GUI

TLS

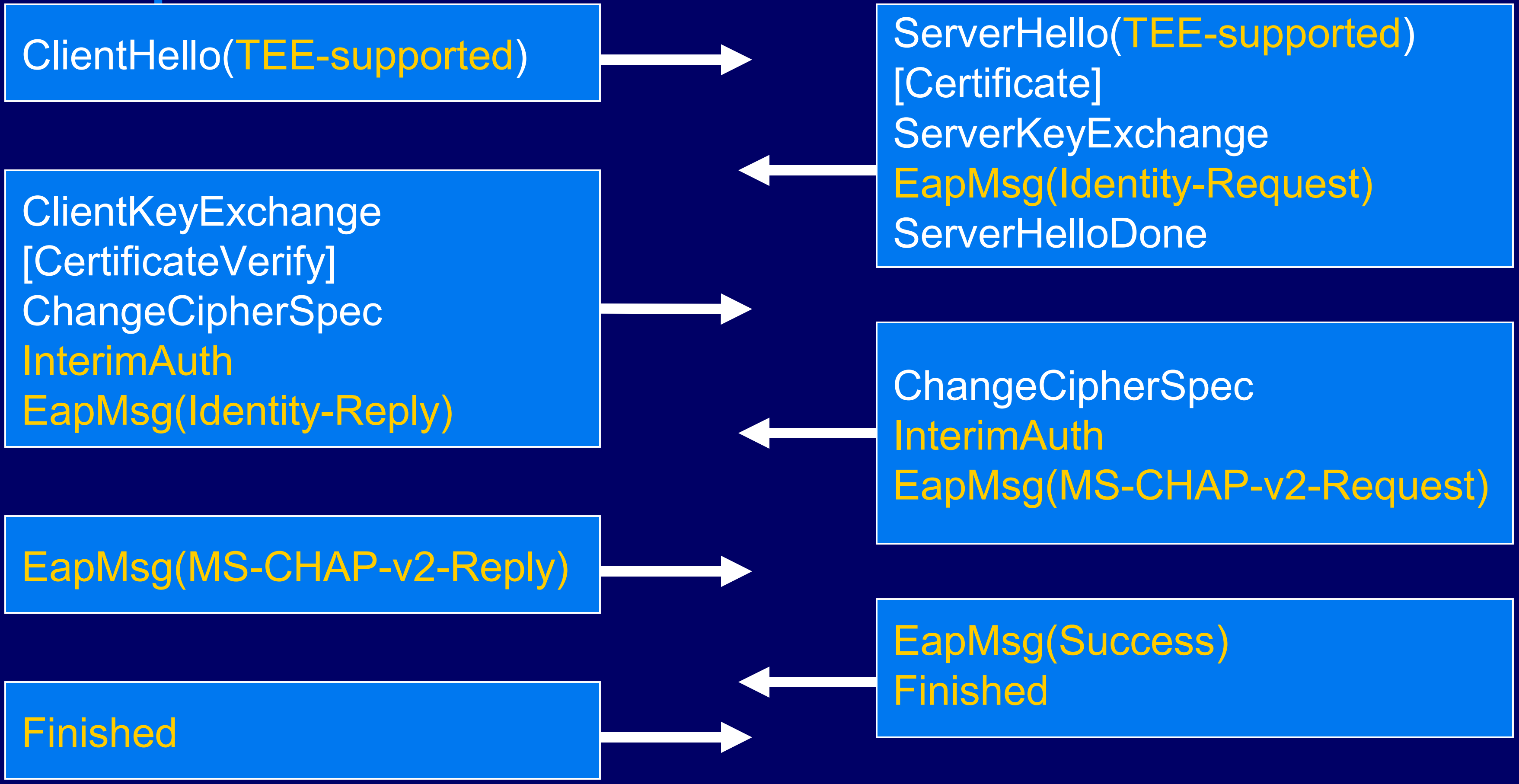
EAP Infrastructure

Server

TLS

AAA Server

Protocol Example



ClientHello(TEE-supported)

ServerHello(TEE-supported)
[Certificate]
ServerKeyExchange
EapMsg(Identity-Request)
ServerHelloDone

ClientKeyExchange
[CertificateVerify]
ChangeCipherSpec
InterimAuth
EapMsg(Identity-Reply)

ChangeCipherSpec
InterimAuth
EapMsg(MS-CHAP-v2-Request)

EapMsg(MS-CHAP-v2-Reply)

EapMsg(Success)
Finished

Finished

Protocol Principles

- Each EAP message translates into an EapMsg TLS message
- InterimAuth signs partial exchange before moving into the EAP stage
- Full identity protection for the client
  - Tradeoff: one round trip
- “Finished” MAC incorporates key from EAP where available
  - Using key generating EAP methods is RECOMMENDED
- “Finished” sent by server first!

Thank You!

yaronf@checkpoint.com