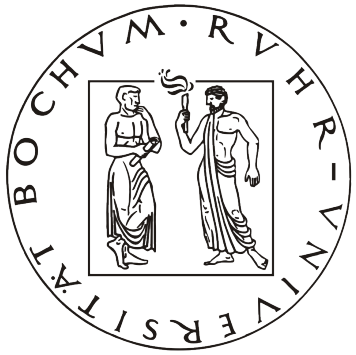


The Second ACM Workshop on Scalable Trusted Computing (STC'07)
Friday Nov. 2, 2007

hgi
Horst-Görtz Institut für IT Sicherheit

Beyond Secure Channels

Yacine Gasmi, Ahmad-Reza Sadeghi, Patrick Stewin, Martin Unger

Horst-Görtz Institute for IT Security
Ruhr-University Bochum

N. Asokan

Nokia Research Center, Helsinki, Finland

“using encryption on the Internet is the equivalent of arranging an armored car to deliver credit card information from someone living in a cardboard box to someone living on a park bench”

(Gene Spafford)

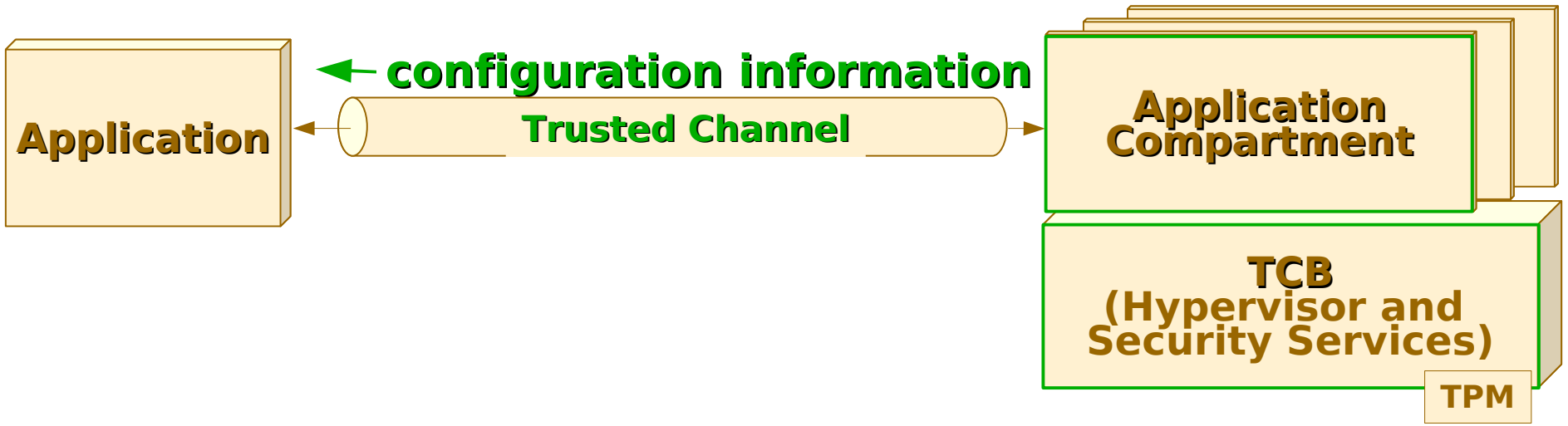
Outline

- Motivation for Trusted Channels
- Basic idea of proposed approach
- Requirements
- Generic approach
- Mapping to SSL/TLS
- Analysis and related work
- Summary and future work

Motivation

- **Secure channel properties**
  - Authenticity of endpoints, Data confidentiality, Freshness and Integrity
- **Problem: Trustworthiness of end point**
  - TLS ensures secure channel properties during transport
  - Malware on server can publish or corrupt received data
- **Solution: Verifiable statement on trustworthiness of endpoint**
- **Possible approach**
  - Extending secure channel protocols by Trusted Computing functionalities (use case Transport Layer Security)
- **Secure Channel becomes Trusted Channel**

Basic Idea: Establish Trusted Channel

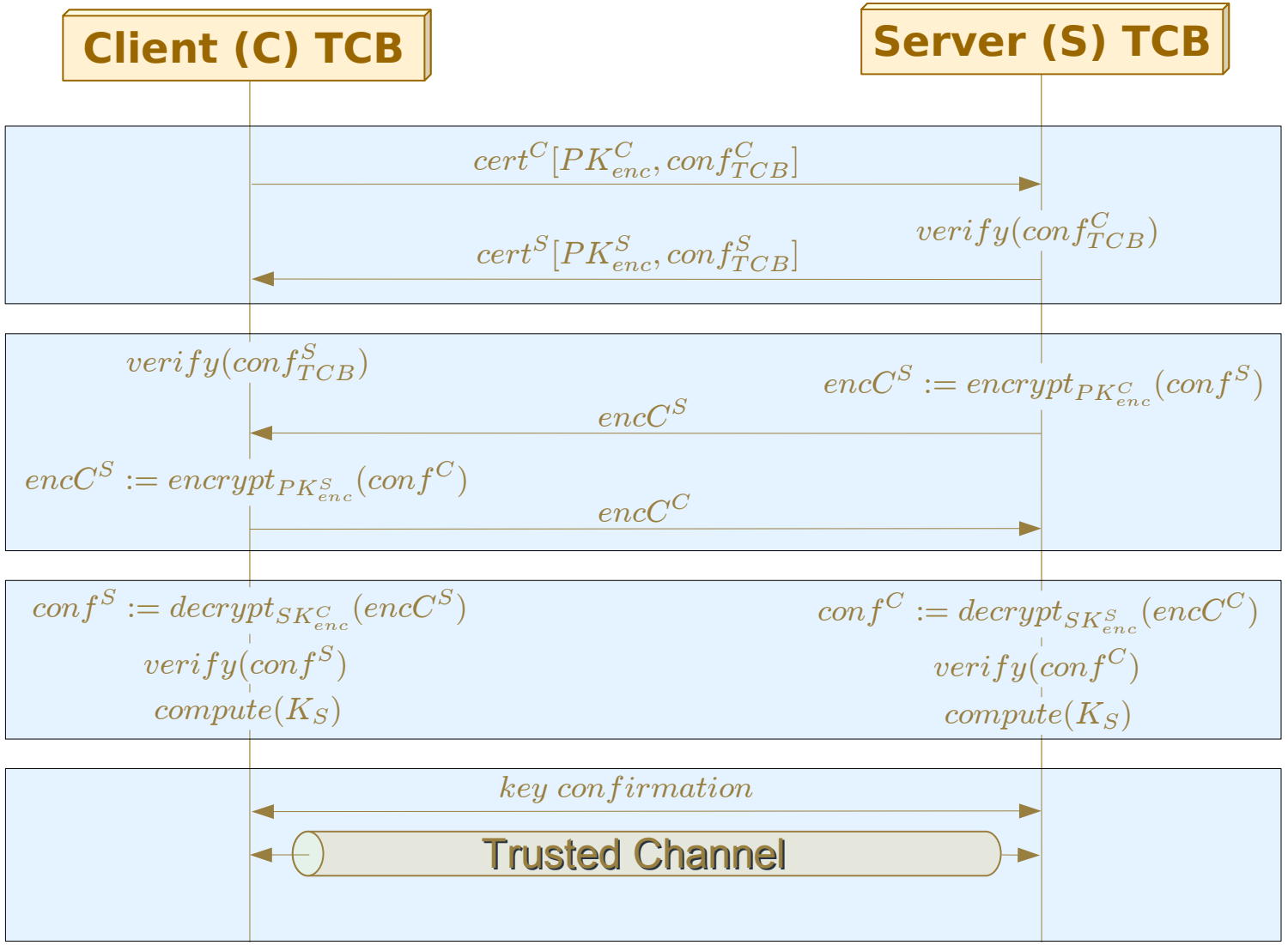


- Measure configuration information of
  - Trusted Computing Base (TCB) (static configuration)
  - Application compartment (dynamic configuration)
- Transmit configuration information to peer
- Generate Session Key bound to configuration
- Allow application to use, but not get, Session Key

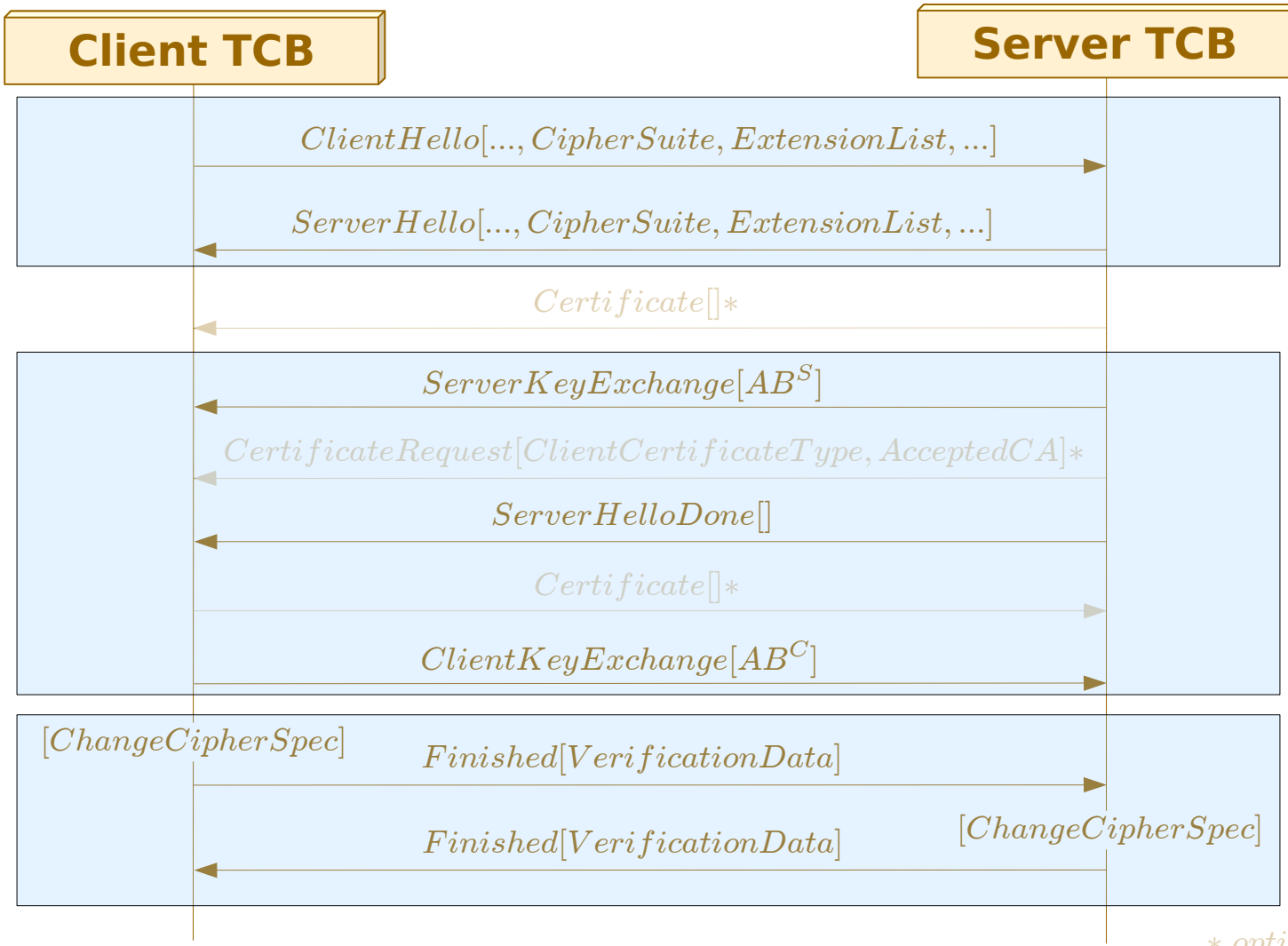
Requirements

- **Retain secure channel properties**
- **Securely link peer configuration and channel**
  - **exchange configuration securely**
    - **at establishment time**
    - **whenever configuration changes**
- **Privacy**
  - **minimize disclosure of configuration information**

Generic Approach



SSL/TLS Protocol with Extensions

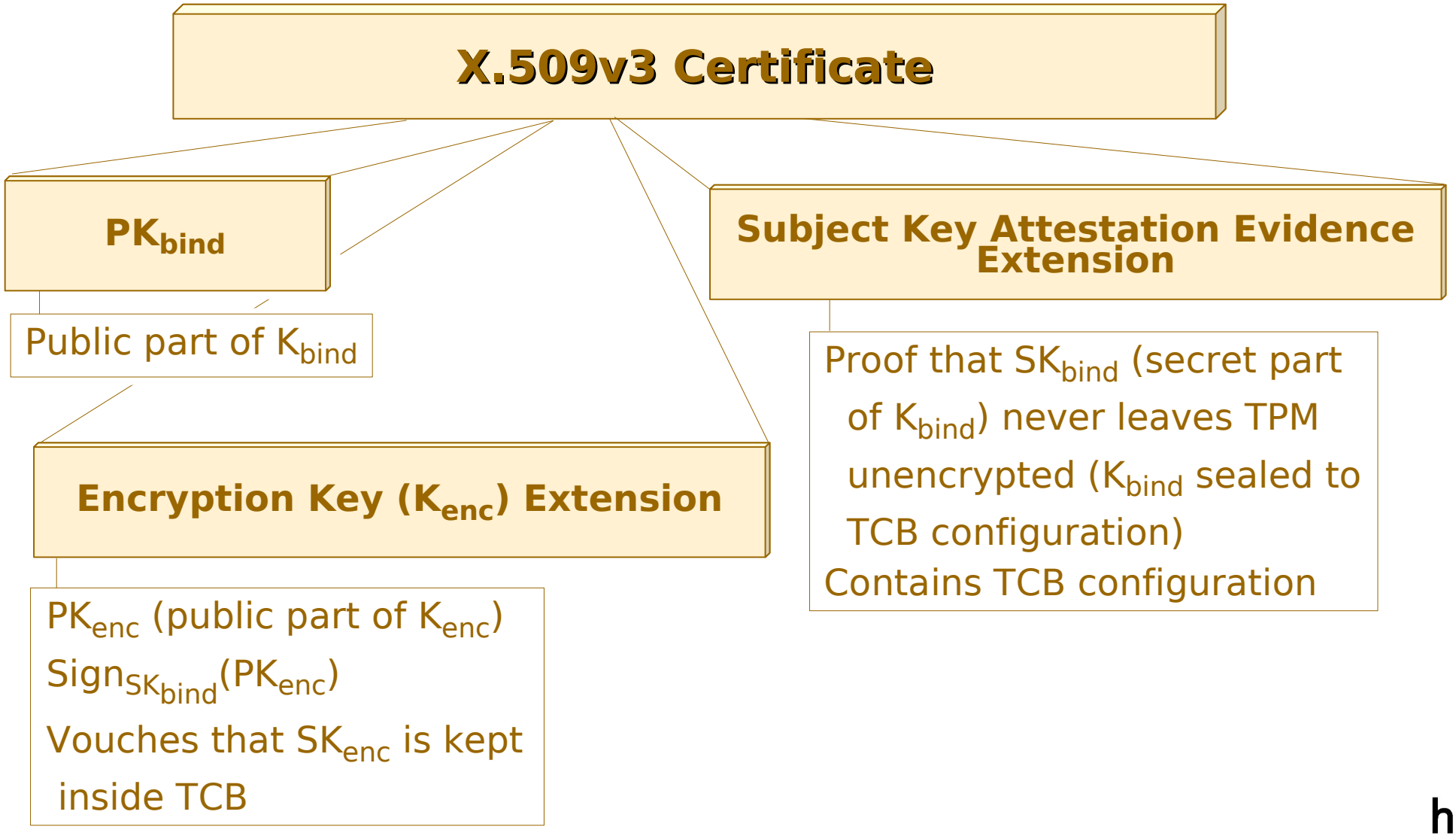


Certificate Exchange + TCB Configuration

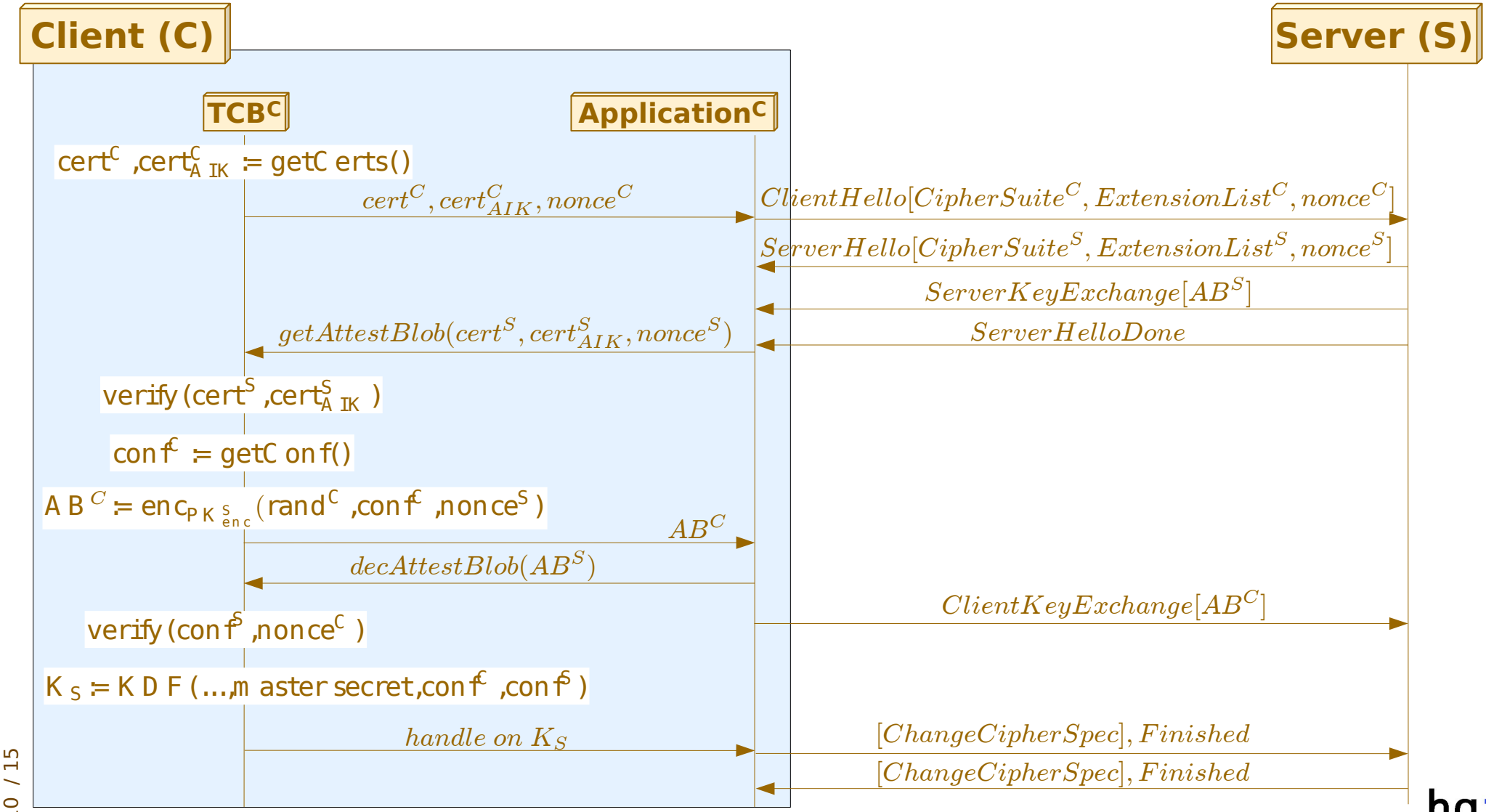
Transmit configuration information to peer (Key Material + Compartment Configuration)

Trusted Channel established

Certificate Extensions



Trusted Channel Protocol



Dealing with State Changes

- **Prohibit state changes**
  - TCB tears down channel (cf. [GPS06])
- **Report state changes to peer**
  - Monitoring agent to detect
  - TCB revokes access to session key from application compartment
  - TCB forces application to report new state to peer
  - New session key calculated as function of old session key and new state

Related Work and Contribution

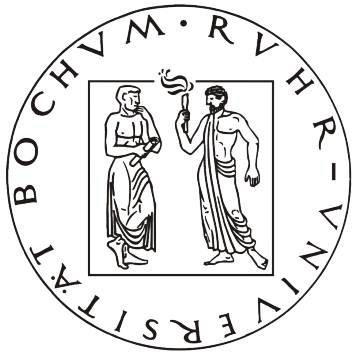
	[STR+06]	[JSM01]	[MSW+04]	[MPP+07]	[SSW+07]	[GSS+07]
TPM	yes	no	yes	yes	yes	yes
SSL/TLS	no	yes	yes	pos.	pos.	yes
Mutual Attestation	no	no	no	no	pos.	yes
State Changes	no	no	no	no	no	pos.
Isolation	no	yes	yes	yes	yes	yes

Summary

- **Configuration information (both static and dynamic)**
  - **authenticated using trusted computing functionalities information**
  - **exchanged using TLS Extensions**
- **Trusted Channel fulfills requirements**
  - **Retain secure channel properties**
  - **Securely link peer configuration and channel**
  - **Privacy of compartment configuration**

Next Steps

- **Finalizing implementation**
- **Performance evaluation**
- **Formal security analysis**
- **Reliable runtime-integrity measurement agent**
- **Adapting approach to other secure channel protocols**



The Second ACM Workshop on Scalable Trusted Computing (STC'07)
Friday Nov. 2, 2007

hgi
Horst-Görtz Institut für IT Sicherheit

Beyond Secure Channels

Yacine Gasmi, Ahmad-Reza Sadeghi, Patrick Stewin, Martin Unger

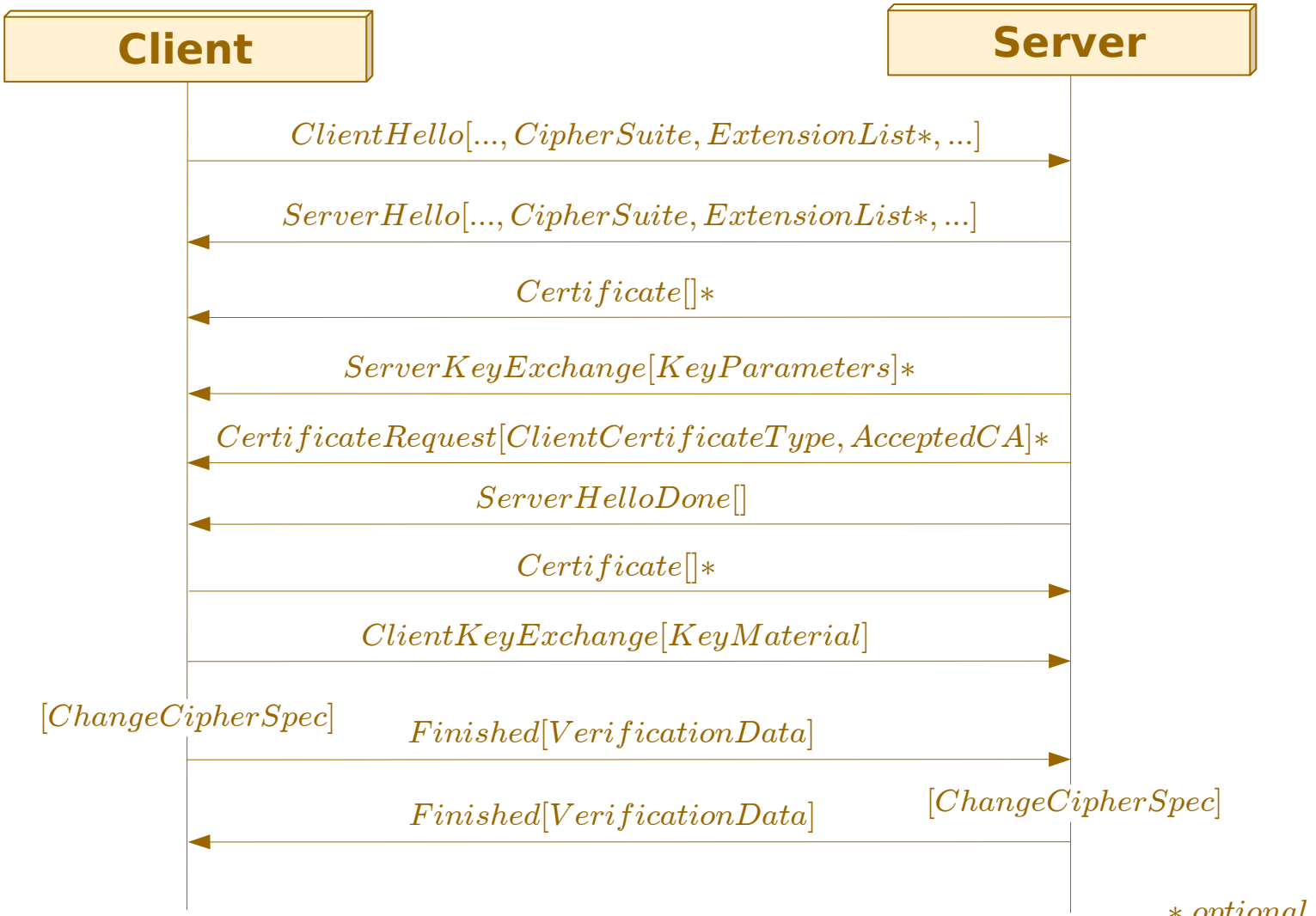
Horst-Görtz Institute for IT Security
Ruhr-University Bochum

N. Asokan

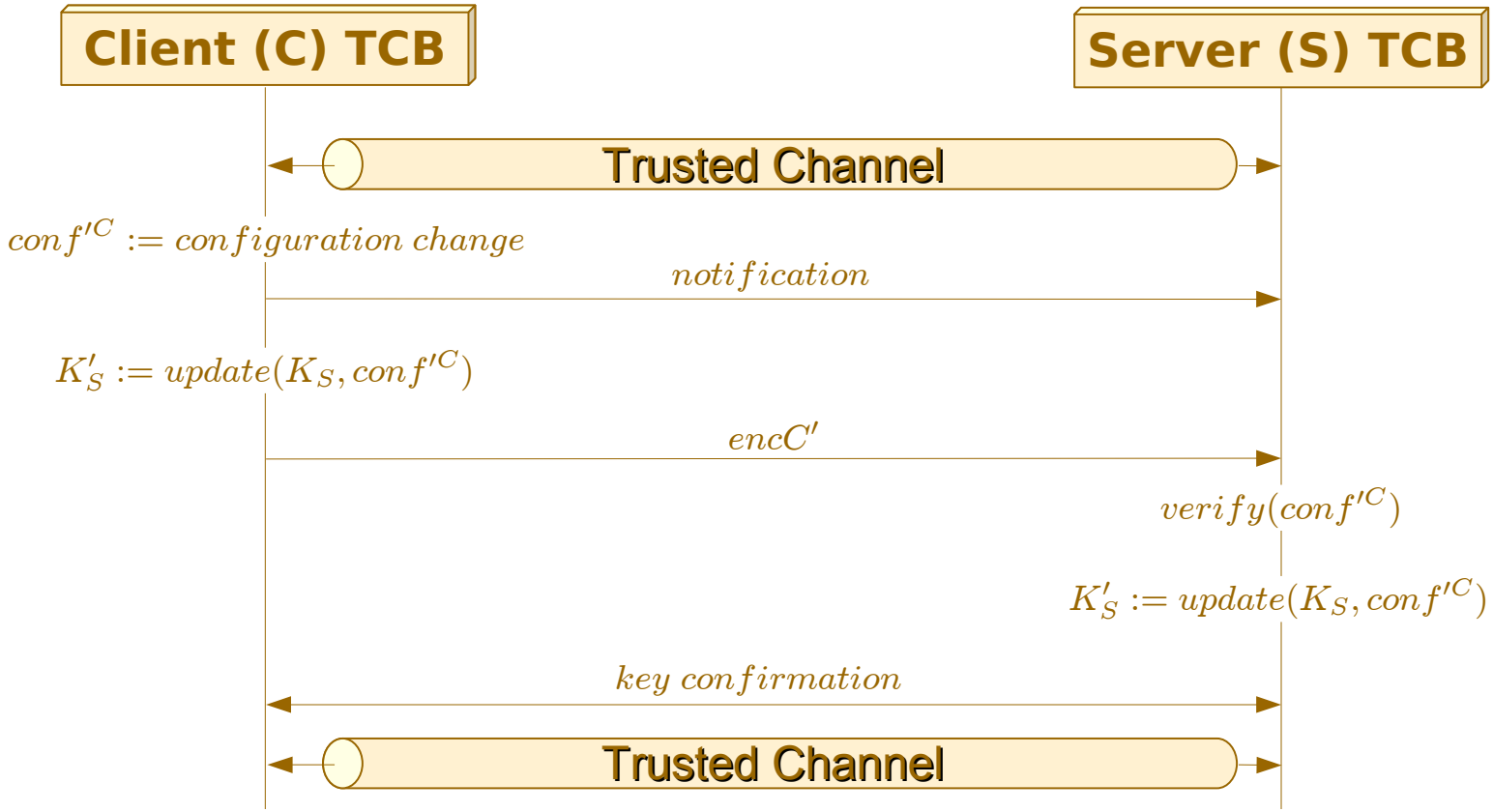
Nokia Research Center, Helsinki, Finland

Backup

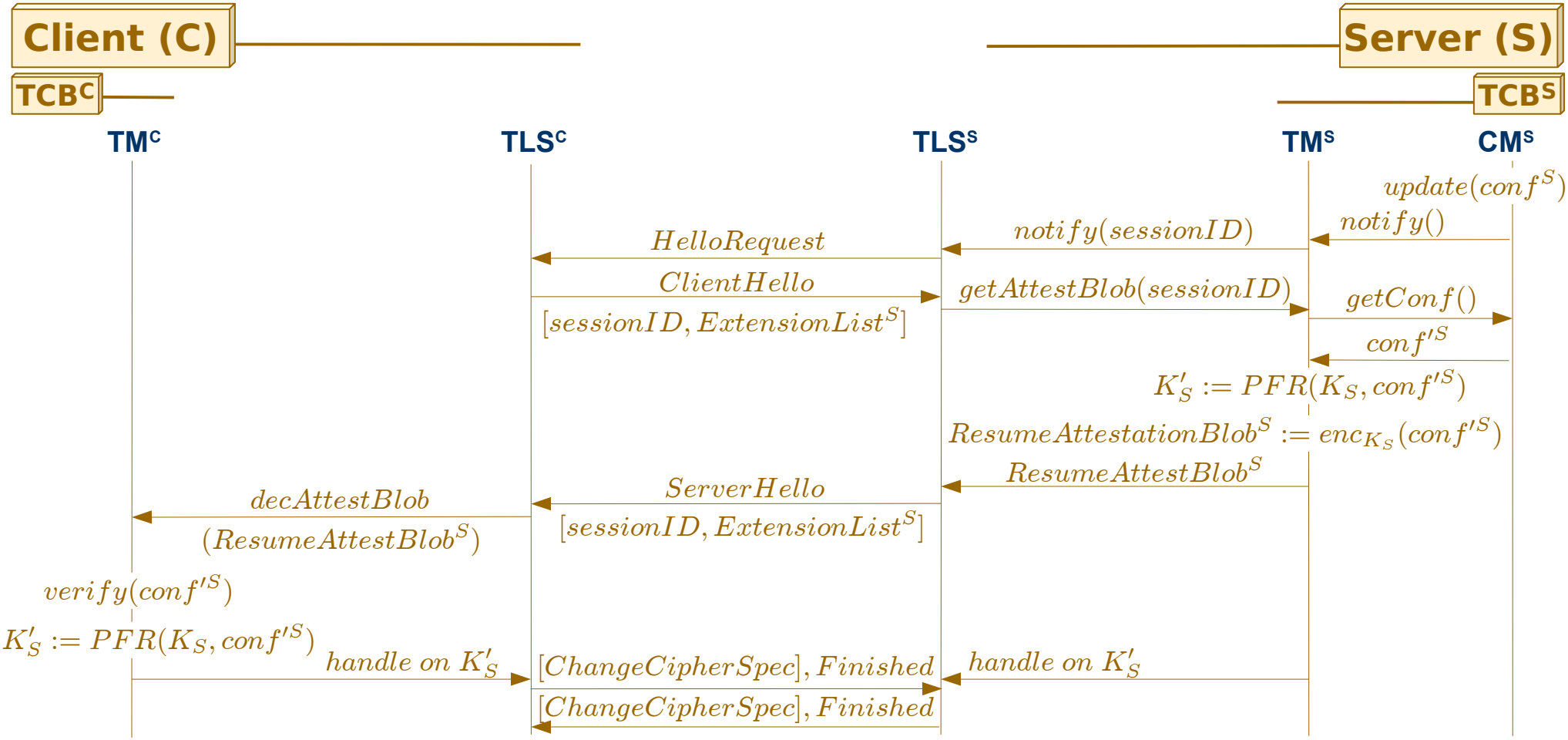
SSL/TLS Protocol (Handshake)



State Change (generic)

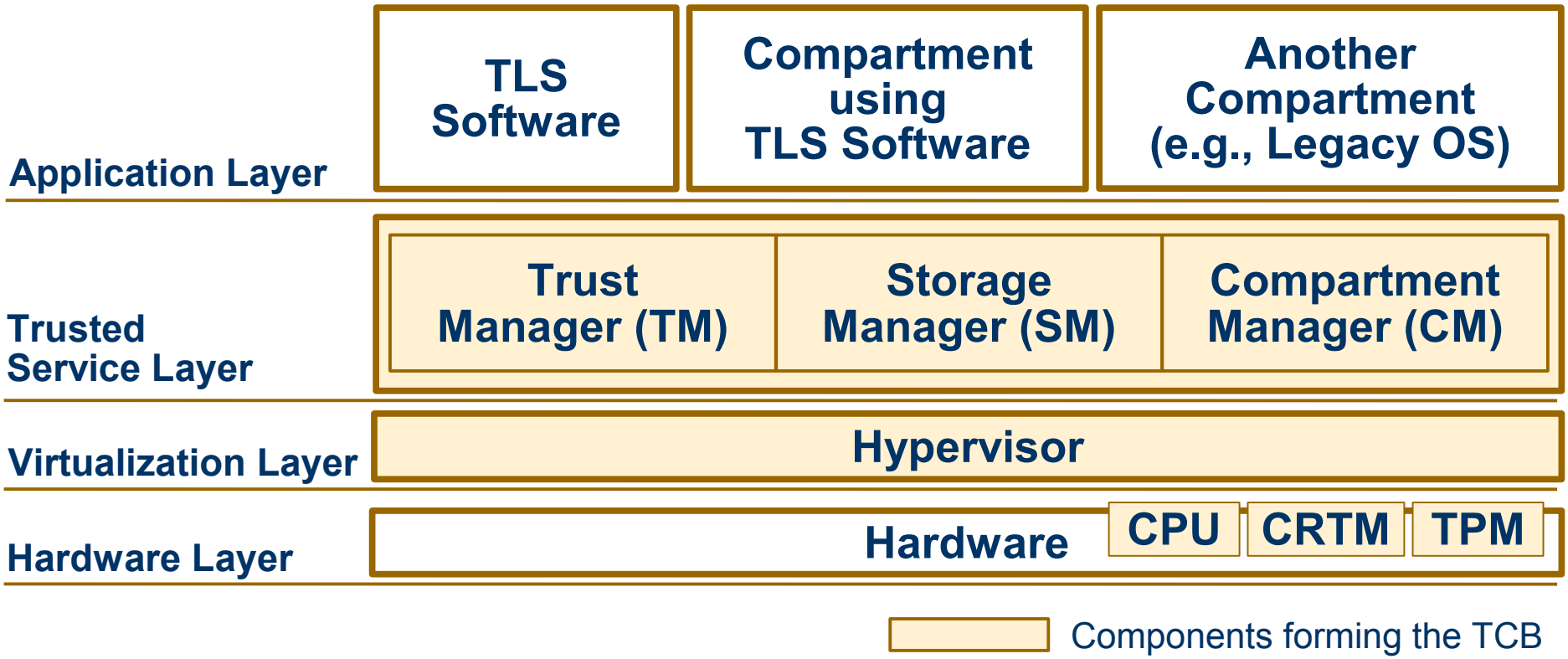


State Change



PRF – $PseudoRandomFunction$

Underlying Security Architecture



- **Provides trusted initialization (Chain of Trust) and isolation**
- **Manages credentials (certificates for K_{bind}, K_{enc} and Attestation Identity Key)**

Trusted Service Layer

- **Trust Manager**
  - Functions for establishing Trusted Channels
  - Keys computed and held in TCB
- **Storage Manager**
  - Persistent data storing (e.g., keys and credentials)
  - Isolation of stored data
  - Sealing of data
  - Trusted Storage (availability, authenticity, confidentiality, integrity and freshness with the help of the monotonic counter feature of the TPM)

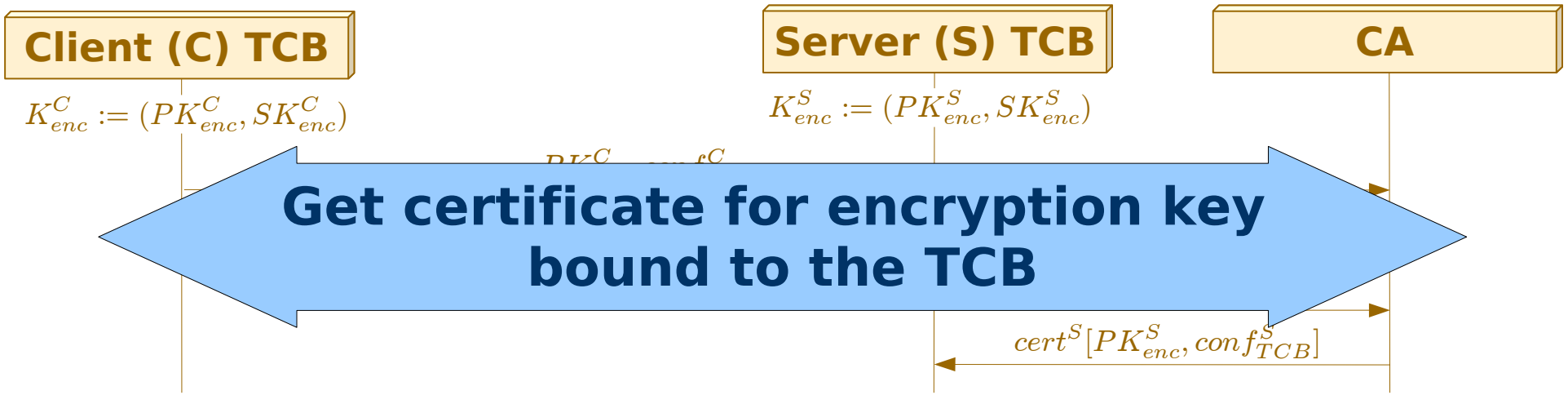
Trusted Service Layer

- **Compartment Manager**
  - **Measures and starts compartments**
  - **Manages compartment ID and configuration via Configuraton Data Structure (CDS)**
  - **Monitoring agent (implementation in progress) for detecting state changes**

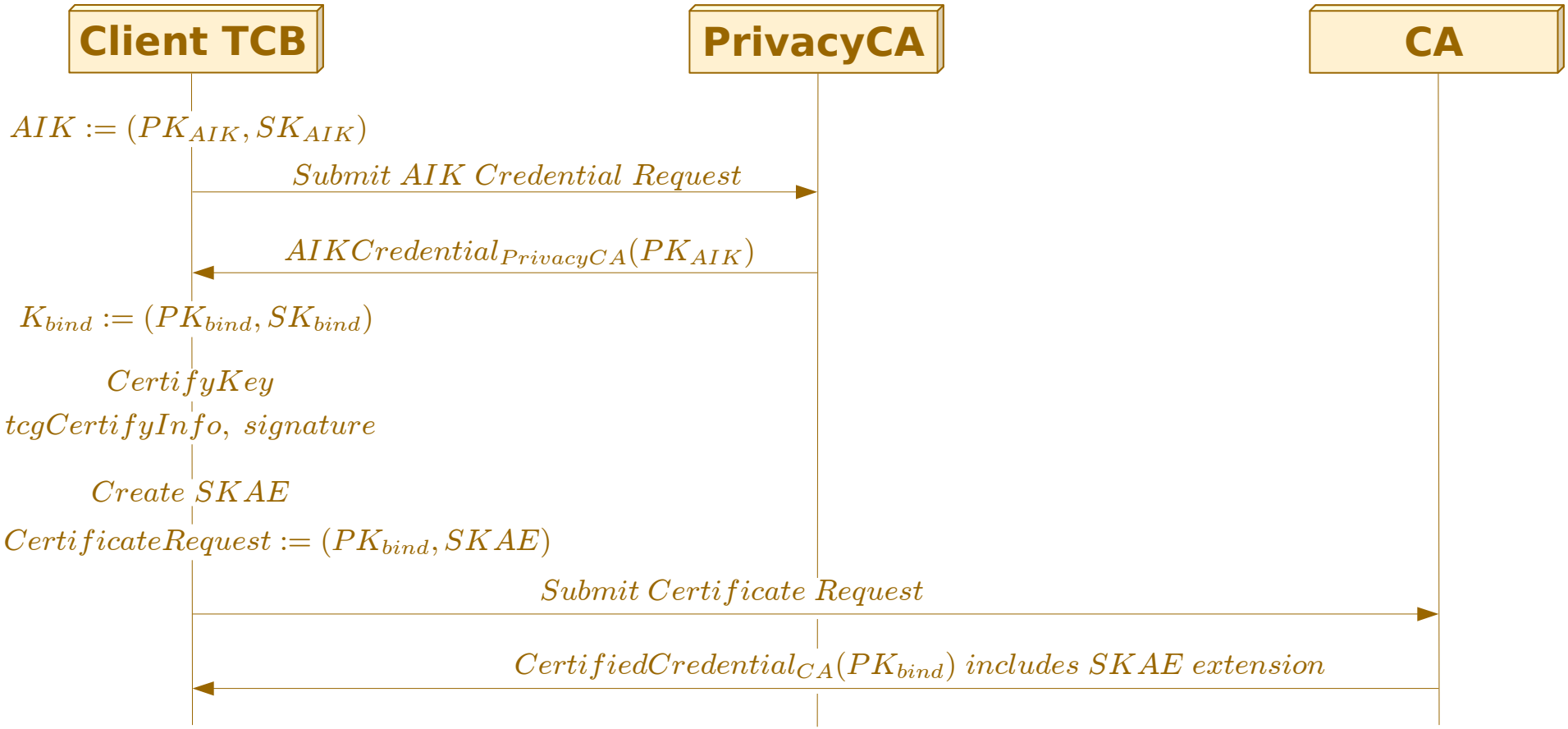
Subject Key Attestation Evidence (SKAE)

- **Specified by the Trusted Computing Group**
- **Can vouch for:**
  - Certain key created within well known trustworthy hardware environment
  - Key only used in corresponding trusted hardware environment
  - Key bound a certain identity, that may be verified (by AIK certificate)
- **Can be integrated in standard X.509 certificates**

Creating certificate



Subject Key Attestation Evidence (SKAE)



(cf. [TCG05] page 11)

Subject Key Attestation Evidence (SKAE)

- **Necessity for using SKAE:**
  - **Private part of K_{enc} is held in TCB**
  - **AIK cannot be applied directly for signing K_{enc} (cf. [TGC03] page 18)**
  - **K_{bind} signed by AIK and certified using SKAE**
    - **Similar security assumption as uttered for AIK (private part never leaves TPM unencrypted) apply for K_{bind}**
- **Standardized usage of CertifyKey command and TPM_CERTIFY_INFO2 structure (result: X.509 certificate extension)**

Privacy of configuration information

- **Separation of static and dynamic configuration**
  - Dynamic configuration not visible to eavesdroppers
- **Dynamic configuration only consists of application compartment**
  - peer does not learn about other compartments

Bibliography

[GPS06] K. Goldman, R. Perez, and R. Sailer. Linking remote attestation to secure tunnel endpoints. In STC ’06: Proceedings of the first ACM workshop on Scalable trusted computing, pages 21–24, New York, NY, USA, Nov. 2006. ACM Press.

[JSM01] S. Jiang, S. Smith, and K. Minami. Securing Web Servers against Insider Attack. In ACSAC ’01: Proceedings of the 17th Annual Computer Security Applications Conference, page 265, Washington, DC, USA, 2001. IEEE Computer Society.

[MSWSB04] J. Marchesini, S. W. Smith, O. Wild, J. Stabiner, and A. Barsamian. Open-Source Applications of TCPA Hardware. In ACSAC ’04: Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC’04), pages 294–303, Washington, DC, USA, 2004. IEEE Computer Society.

[MPPRS07] J. M. McCune, B. Parno, A. Perrig, M. K. Reiter, and A. Seshadri. Minimal TCB Code Execution. In SP ’07: Proceedings of the 2007 IEEE Symposium on Security and Privacy, pages 267–272, Washington, DC, USA, 2007. IEEE Computer Society.

[SSWAE07] A.-R. Sadeghi, C. Stüble, M. Wolf, N. Asokan, and J.-E. Ekberg. Enabling Fairer Digital Rights Management with Trusted Computing, 2007. To be presented at ISC07, Information Security Conference 2007.

[Spa97] G. Spafford. Attributed to in Risks Digest 19.37 review of @LARGE, by David H. Freedman and Charles C. Mann, Sept. 1997. http://catless.ncl.ac.uk/Risks/19.37.html.

[STRE06] F. Stumpf, O. Tafreschi, P. Röder, and C. Eckert. A robust Integrity Reporting Protocol for Remote Attestation. In Proceedings of the Second Workshop on Advances in Trusted Computing (WATC ’06 Fall), Tokyo, Dec. 2006.