



**CISPA**

HELMHOLTZ CENTER FOR  
INFORMATION SECURITY

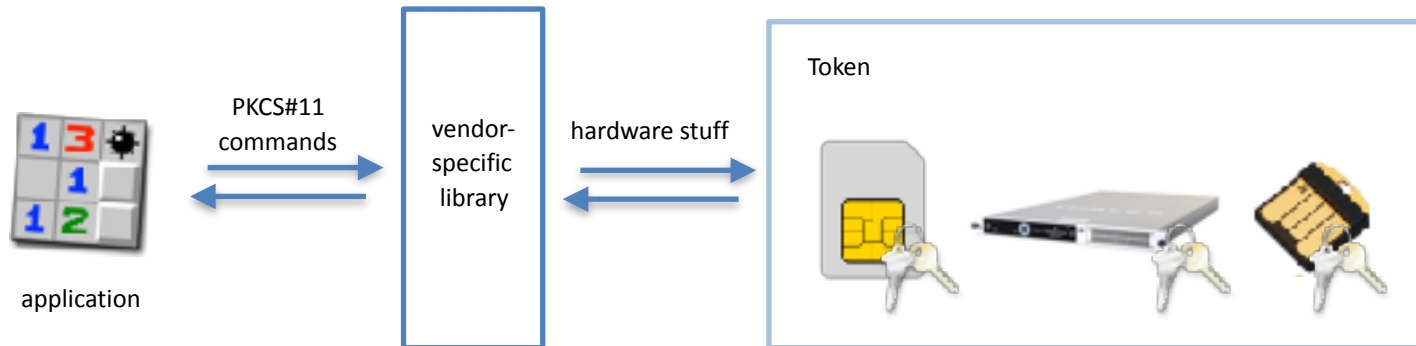
# How to wrap it up – A formally verified proposal for the use of authenticated wrapping in PKCS#11

Alexander Dax, **Robert Künnemann**, Sven Tangermann and Michael Backes

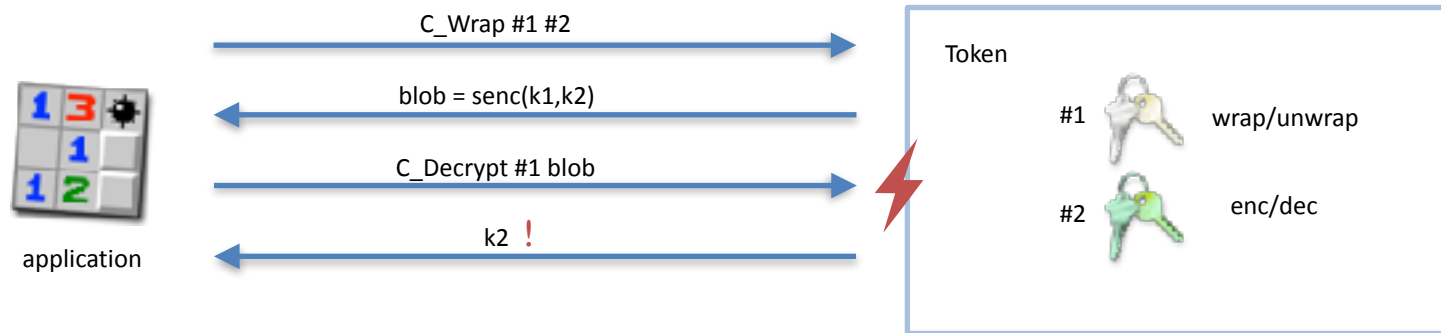


## PKCS #11 Cryptographic Token Interface Base Specification Version 2.40 **Plus Errata 01**

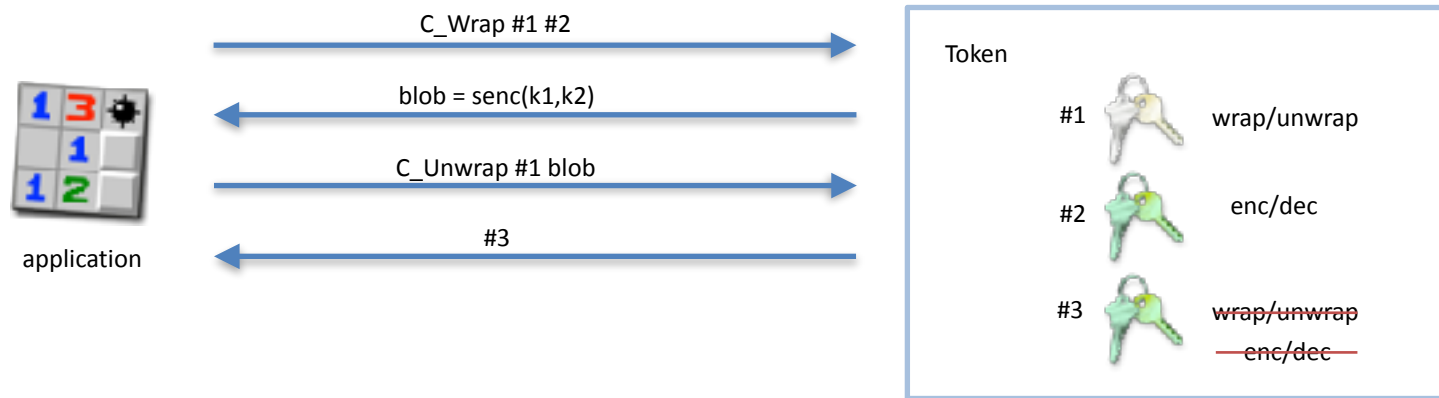
OASIS Standard **Incorporating Approved Errata 01**



# PKCS#11 - logical attacks



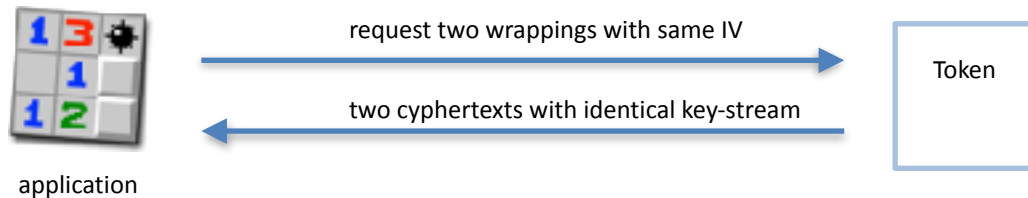
# PKCS#11 - logical attacks



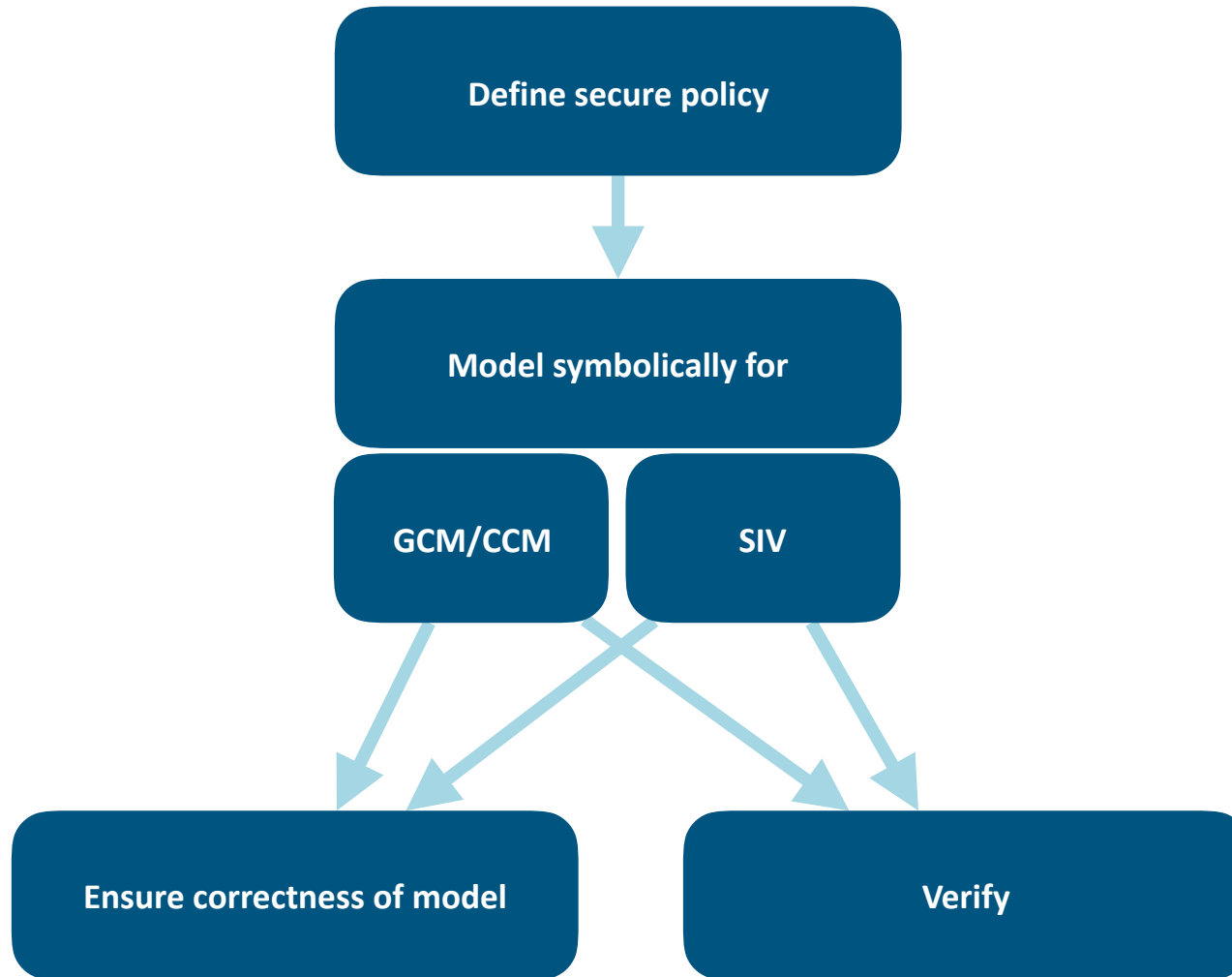
⇒ need to authenticate attributes!

# PKCS#11 - state of key-wrapping




- PKCS#11v2.40 introduces GCM and CCM. The end..?
- .. no, two-pad attack:



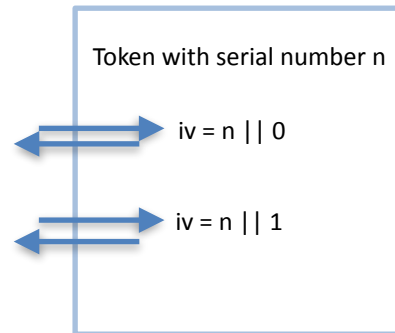
- PKCS#11v3.00 in drafting stage:
  - Can we fix this for GCM and CCM?
  - Is SIV (synthetic IV) an alternative?
  - Is authenticated key-wrapping an improvement?



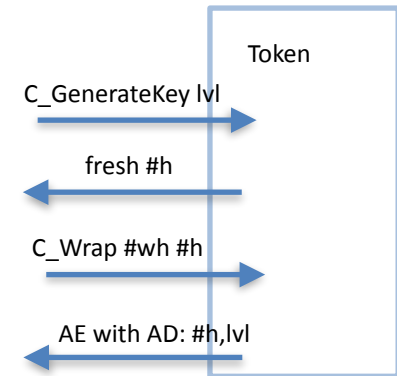
## key hierarchy

lvl	type	permitted operations
$\geq 3$		management: wrap/unwrap
2		usage: enc/dec, MAC, sig
1	 0101 0110 1000 1000	payload

## globally unique counters



## authenticated handles



- **provably secure:** key-secrecy and handle-integrity
- **more functionality** than other provable secure policies, where either
  - one cannot backup wrapping keys
  - keys have less attributes after unwrapping, they "degrade"
- **downside:** static hierarchy

- PKCS#11v2.40 added GCM,CCM, but insecurely
- v3.00 in draft:
  - C\_Encrypt and C\_Wrap key cannot output IV (historically user supplied)
  - new interface C\_EncryptMessage specifically for AEAD
    - keep cryptographic state for multiple messages with possibly different IVs, additional data
    - application can request internal IV generation, pointer to IV is thus either input or out parameter
  - need same convention for wrap (not even a new interface!)
  - FIPS basically requires internal IV-generation for GCM



(see paper)

- IV generation is part of protocol, hence IV needs to be exposed
- equational theory:

$$\begin{aligned}\text{sdec}(k, iv, h, \text{senc}(k, iv, h, m)) &= m \\ \text{sdecSuc}(k, iv, h, \text{senc}(k, iv, h, m)) &= \text{true}() \\ \text{getHeader}(\text{senc}(k, iv, h, m)) &= h \\ \text{getIV}(\text{senc}(k, iv, h, m)) &= iv\end{aligned}$$

- sound for GCM, CCM, SIV?
- DAE-N security:  $A^{O_k^{Enc}(\cdot, \cdot, \cdot), O_k^{Dec}(\cdot, \cdot, \cdot)} \approx A^{\$(\cdot, \cdot, \cdot), \perp(\cdot, \cdot, \cdot)}$  as long as A does not reuse IVs or query previous encryptions.
- GCM and CCM are AEAD secure implies DAE-N security.

- **Intuition:** computational adversary can only deduce information if the symbolic adversary can, too.
- **Pro:** composability, thus lots of PKCS#11 functionality covered
- **Contra:** covers only secrecy, not integrity. necessary, but not sufficient
- **Approach:**
  - assume injective function mapping terms to IVs (e.g., concat)
  - as IVs have fixed length, domain needs to be finite
  - impose use of this function at IV position
  - protocol condition: uniqueness of terms given to this function

☒ keys only at key position or  
within wrapping

.. no dynamic corruption :(

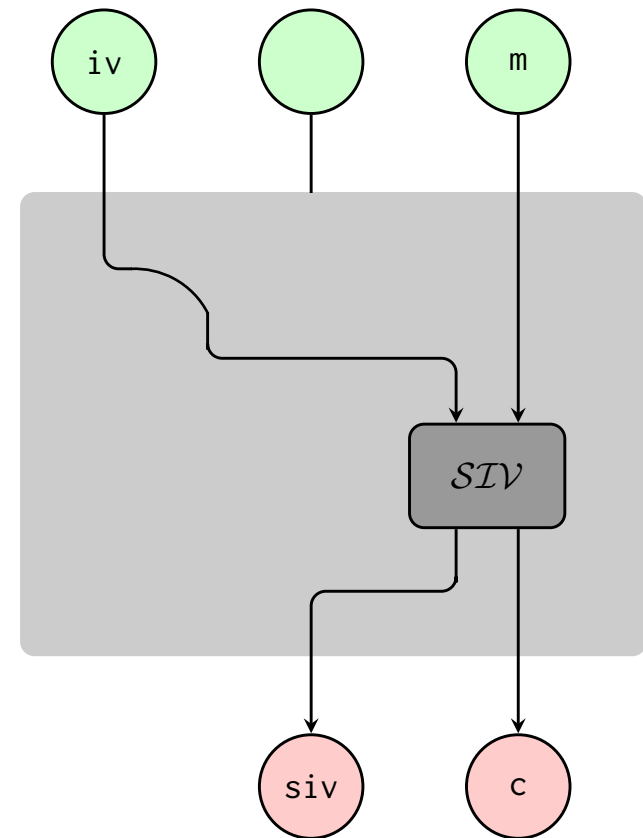
☒ no key-cycles

☐ each term at IV position is  
unique

☛ to check

# What about SIV mode?

- if we prepend IV to header, SIV is DAE-N secure
- but if protocol always sets  $h:=\epsilon$ , construction vanishes
- we obtain model for SIV mode without need for deduction soundness result by writing  $\text{senc}(k, \langle iv, h \rangle, \epsilon, m)$  in place of  $\text{senc}(k, iv, h, m)$
- PKCS#11v3.00 draft: interface spec would be fine, but SIV not part of "current mechanisms"



- IV uniqueness
- key-integrity: all keys are created on some device
- key-secrecy: no key can ever leak
- handle-integrity: keys retain the handle (and level) they were created with
- total verification time: 3mins (GCM/CCM), 3.5 min (SIV)
- three helping lemmas

# Wrap-up & Take-away

