SECURITY
LICENSING
PERFECTION IN PROTECTION

# Document Protection

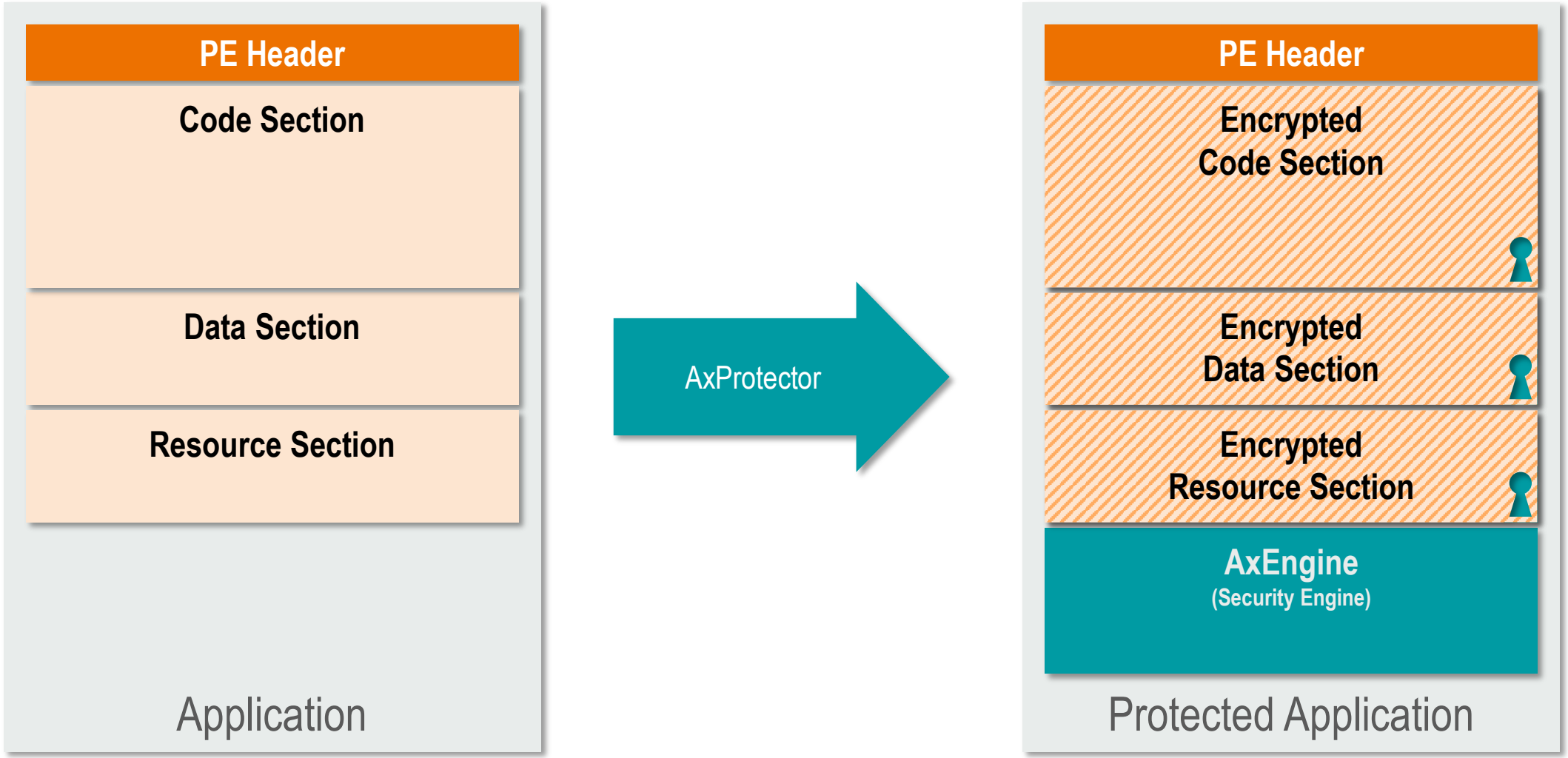
## A Shelter to Protect your Documents

Ruediger Kuegler | VP Sales
ruediger.kuegler@wibu.com

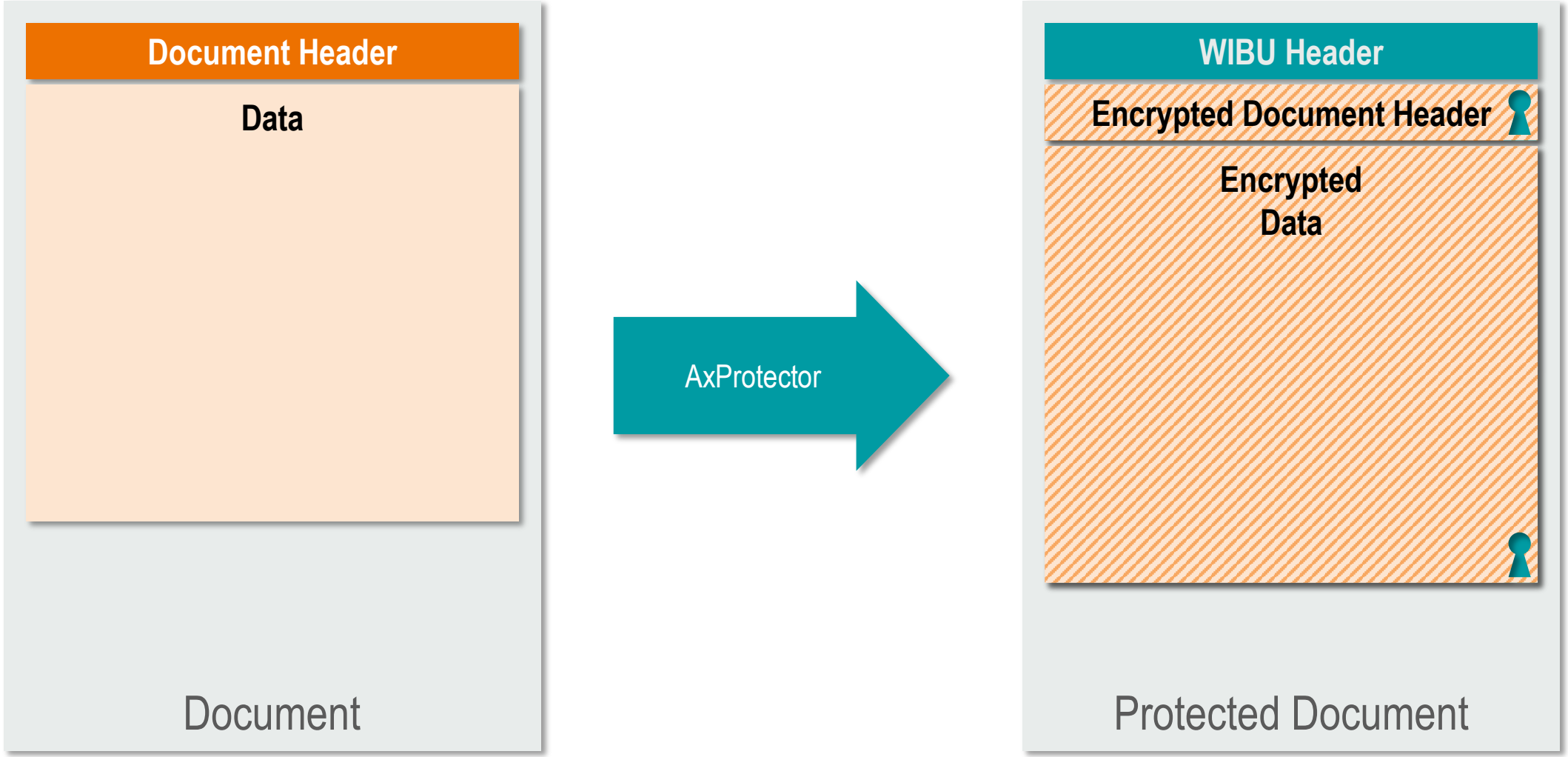
Stefan Bamberg | Senior Key Account Manager
stefan.bamberg@wibu.com

# Applications vs Documents

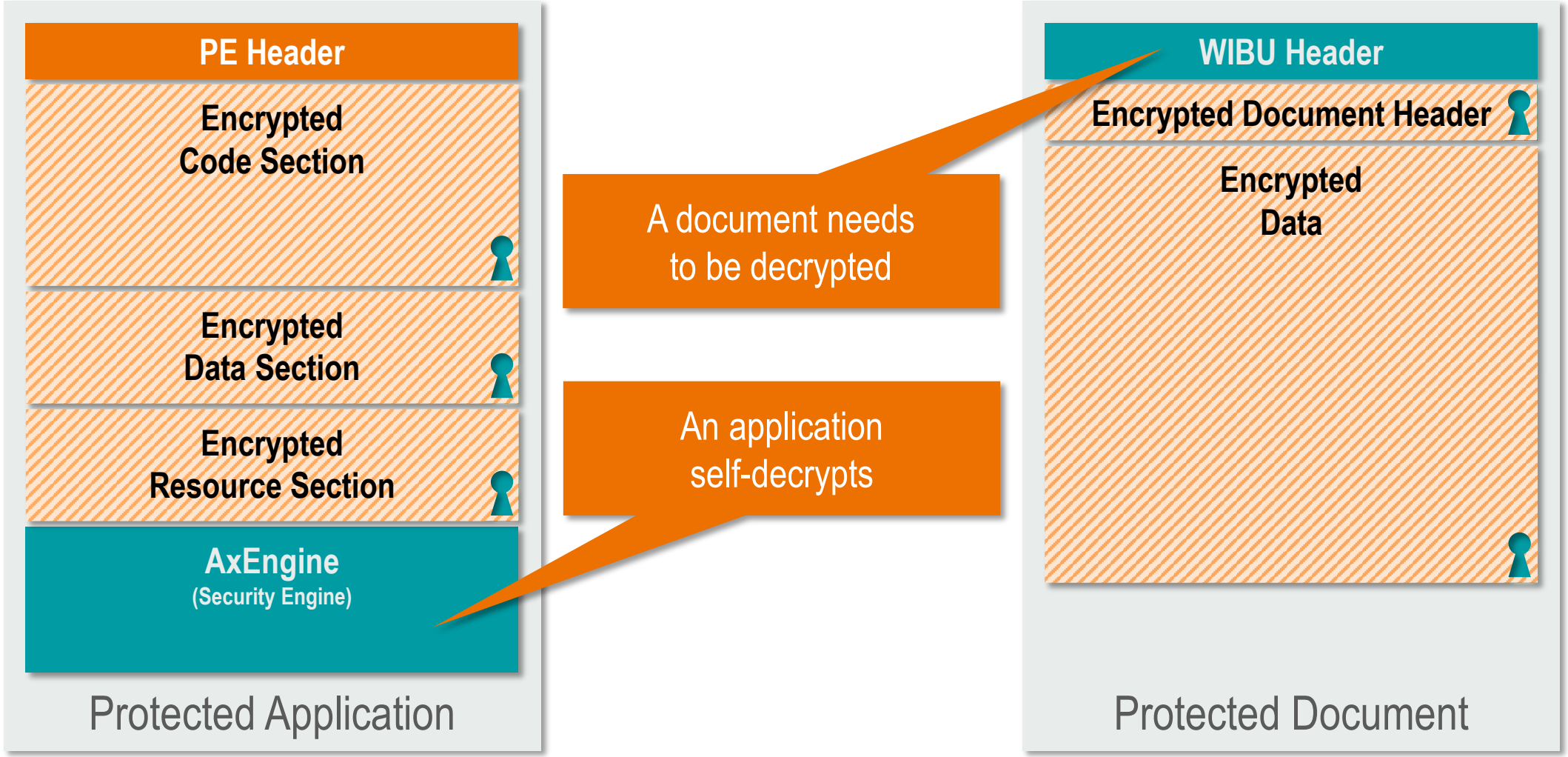
# Software Encryption with AxProtector



# Document Encryption with AxProtector



# Protected Application vs Protected Document



# Summary: Applications vs Documents

## Application

- AxProtector encrypts all sections
- AxProtector inserts decrypting code (AxEngine)
- AxProtector modifies the OEP in the PE Header
- The application is still executable and decrypts itself at start

## Document

- AxProtector encrypts the complete document, incl. possibly existing Headers
- AxProtector sets its own Header (WIBU Header) at the beginning of the document
- An application is required to recognize this document and decrypt it when loaded

# PDF Files

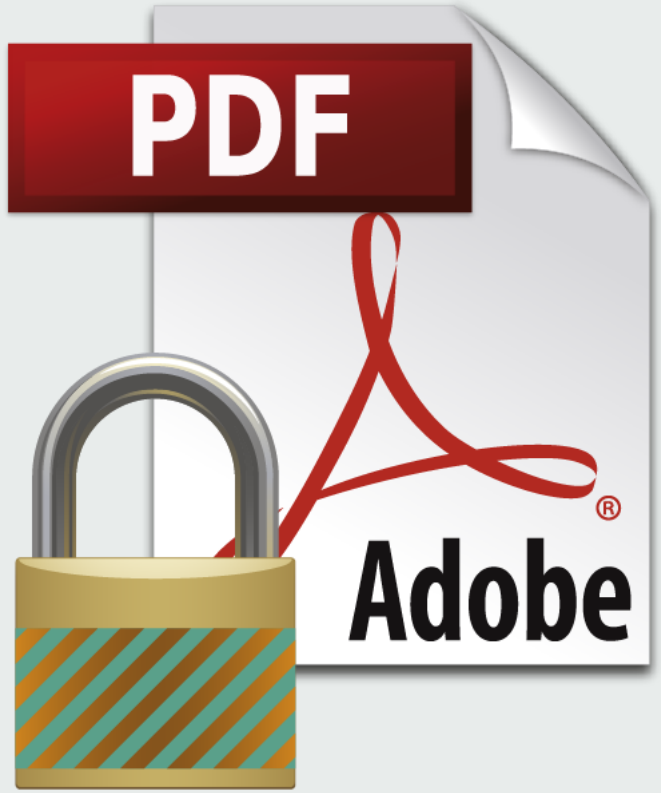
# Encrypting a PDF File


PDF File

SmartShelter|PDF

Adobe Acrobat
+
SmartShelter|PDF Author
+
Firm Security Box

*Firm Code = 10*
*Product Code = 1000*

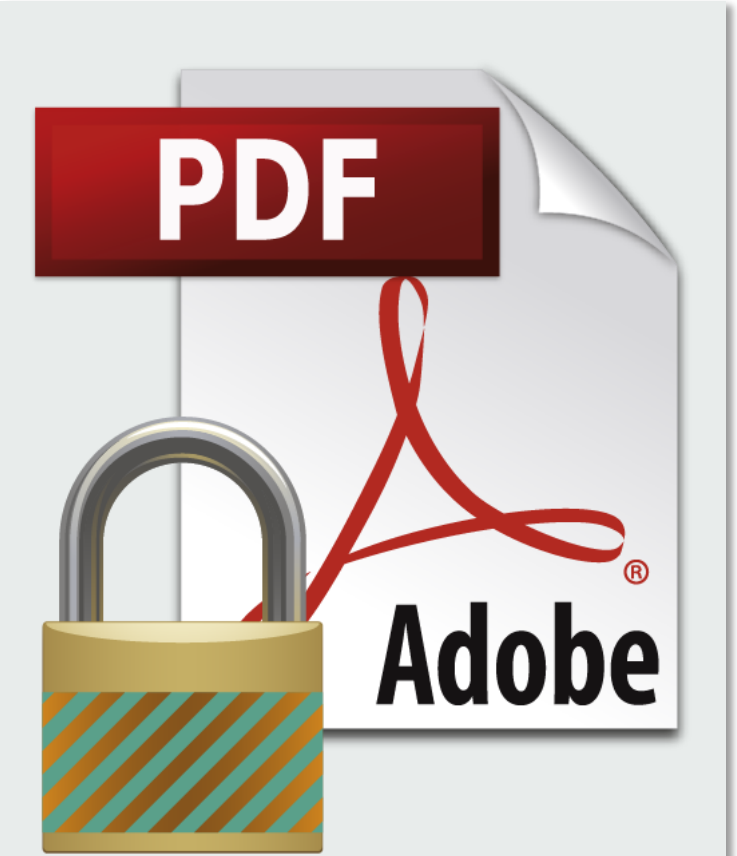



Protected PDF File

# Encrypting a PDF File

- SmartShelter|PDF relies on the native encryption mechanism from Adobe Acrobat (AES, 256 bit)
- SmartShelter|PDF creates a password for this encryption mechanism
  - Invisible to the user and thus impossible to copy
  - Derived from Firm Code and Product Code
  - With high entropy and therefore secure against brute force attacks
- SmartShelter|PDF detects unwanted programs
- SmartShelter|PDF supports all CodeMeter Product Item Options
- SmartShelter|PDF supports all the security features of Adobe Acrobat

# Decrypting a PDF File




Protected PDF File

SmartShelter|PDF

Adobe Acrobat Reader
+
SmartShelter|PDF Reader
+
License (CmDongle / CmActLicense)

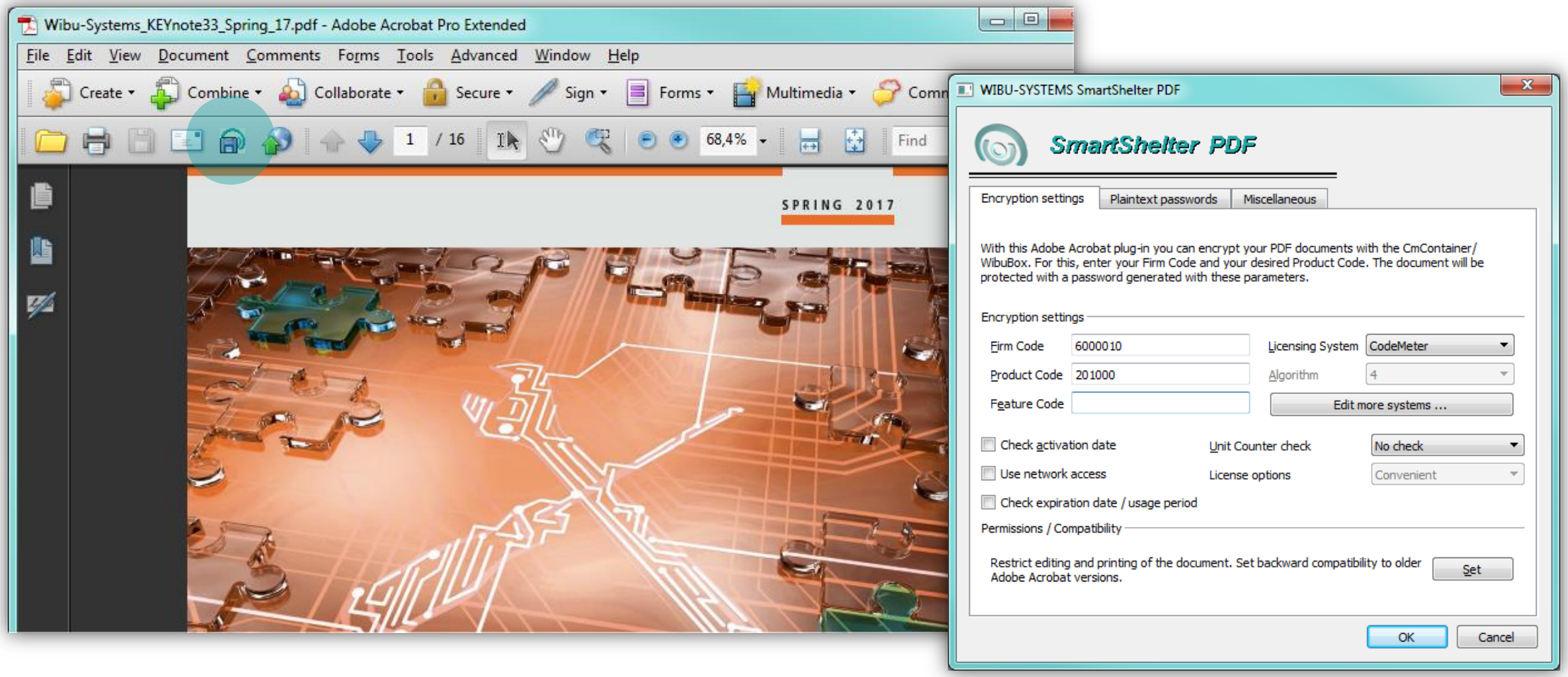
*Firm Code = 10*
*Product Code = 1000*


PDF File in Memory

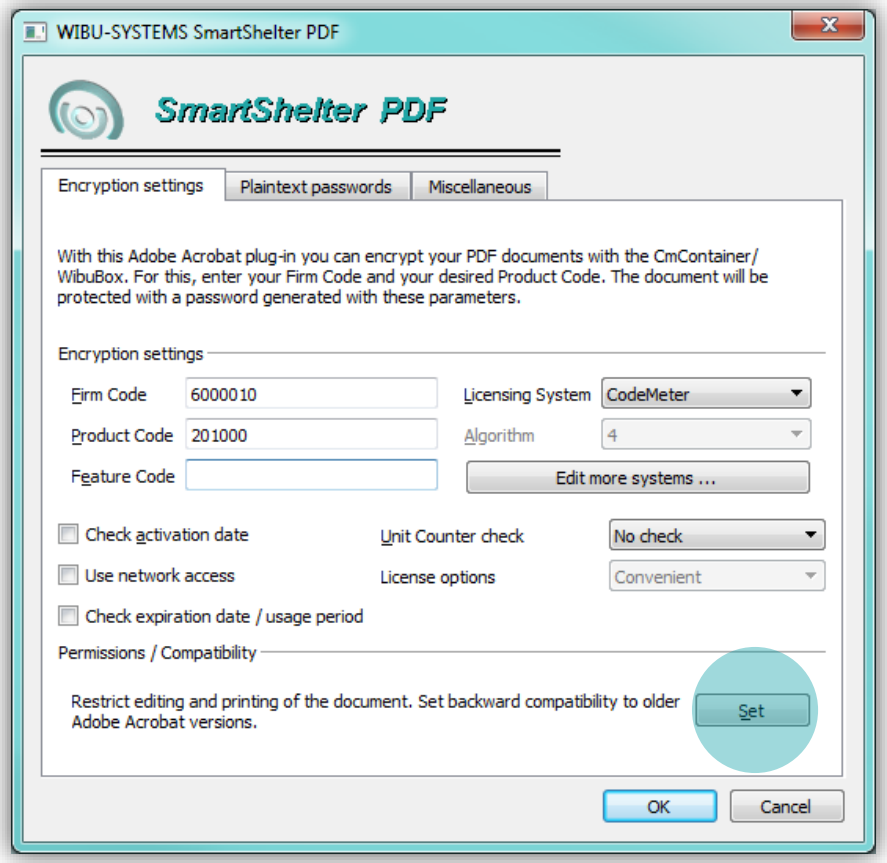
# Decrypting a PDF File

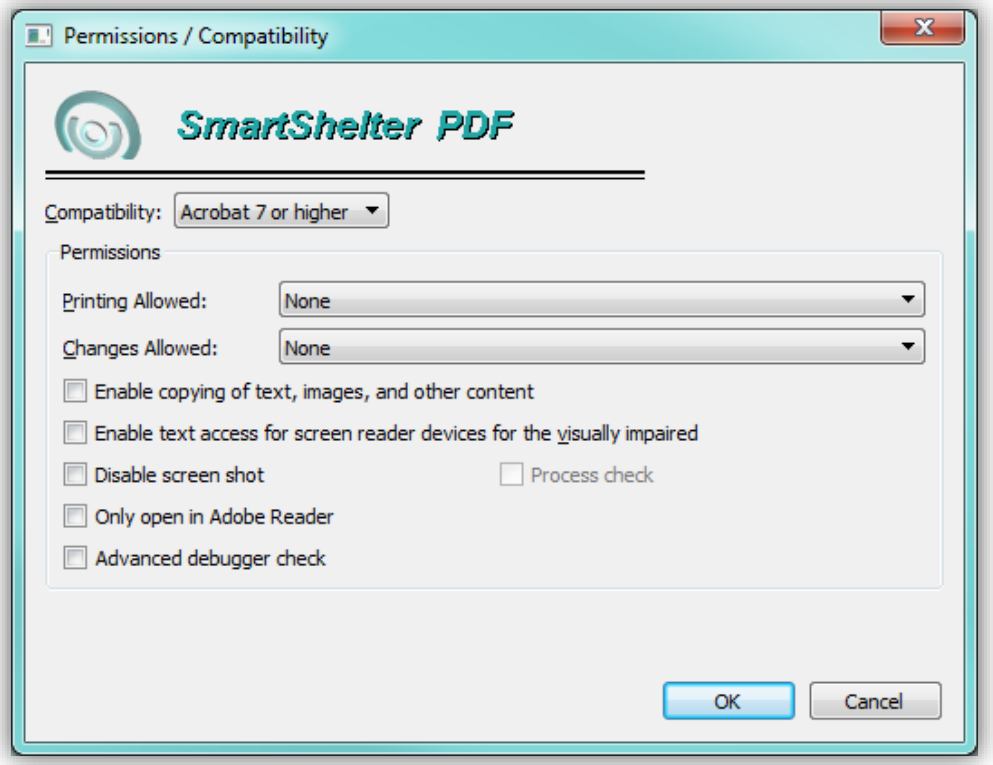
- SmartShelter|PDF verifies if a valid license (Firm Code / Product Code) is available
- SmartShelter|PDF generates the decryption password
  - The user is not aware of the used password
  - The document is decrypted in the memory
- SmartShelter|PDF is on the alert for unwanted programs and ready to close the PDF file immediately if any unwanted program is detected
- SmartShelter|PDF provides an interface for customized error dialogs
- SmartShelter|PDF supports all CodeMeter Product Item Options

# Protecting PDF Documents (1)

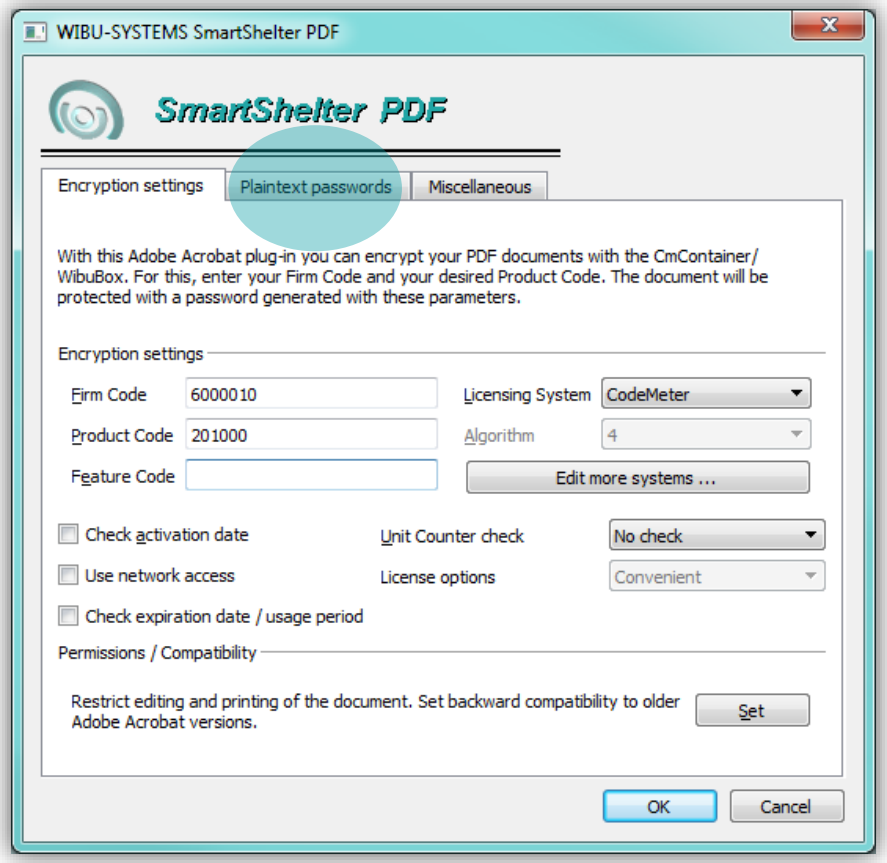


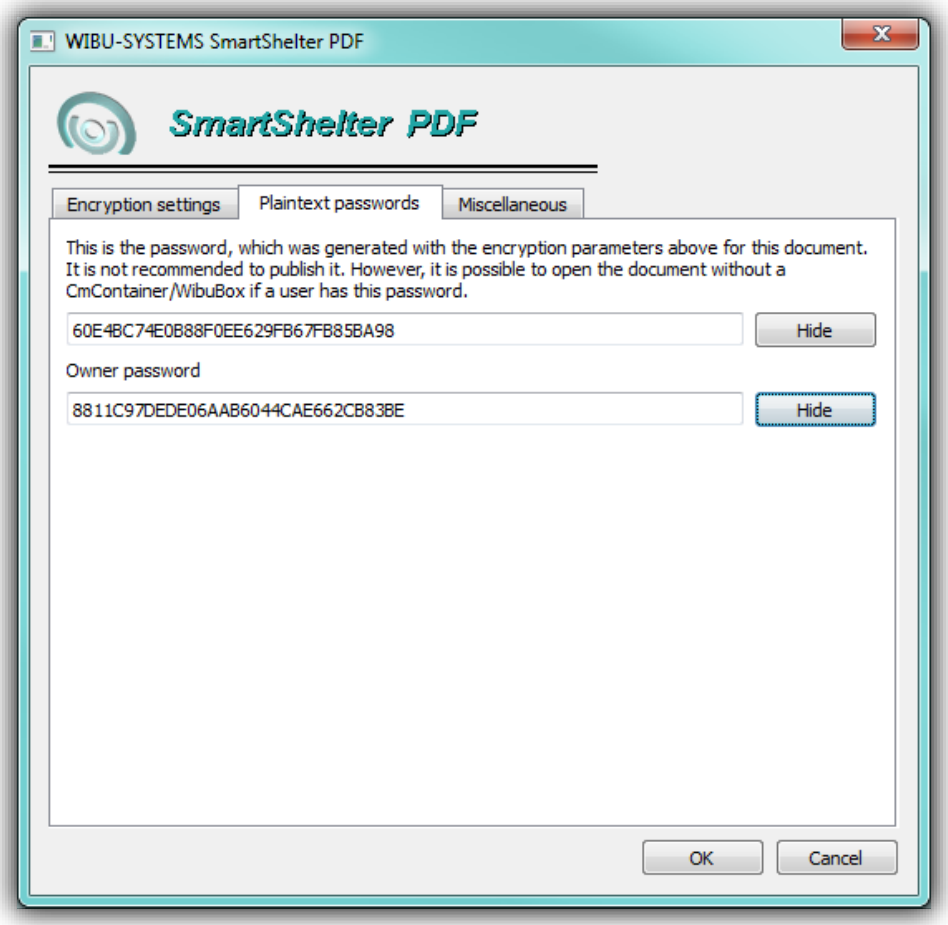
# Protecting PDF Documents (2)



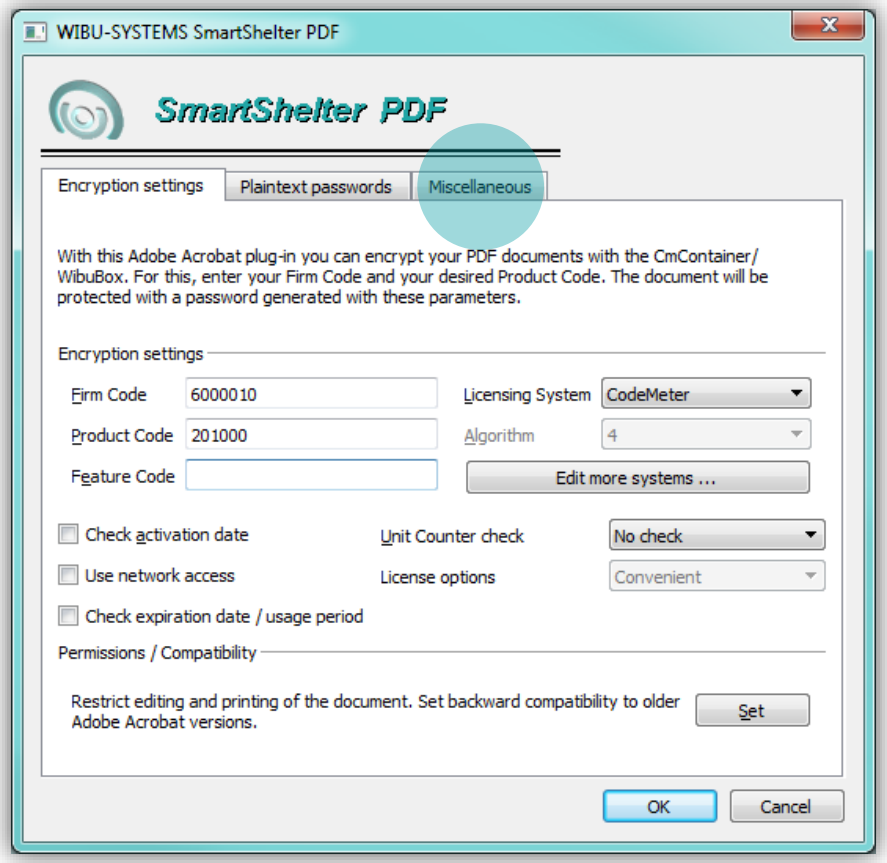



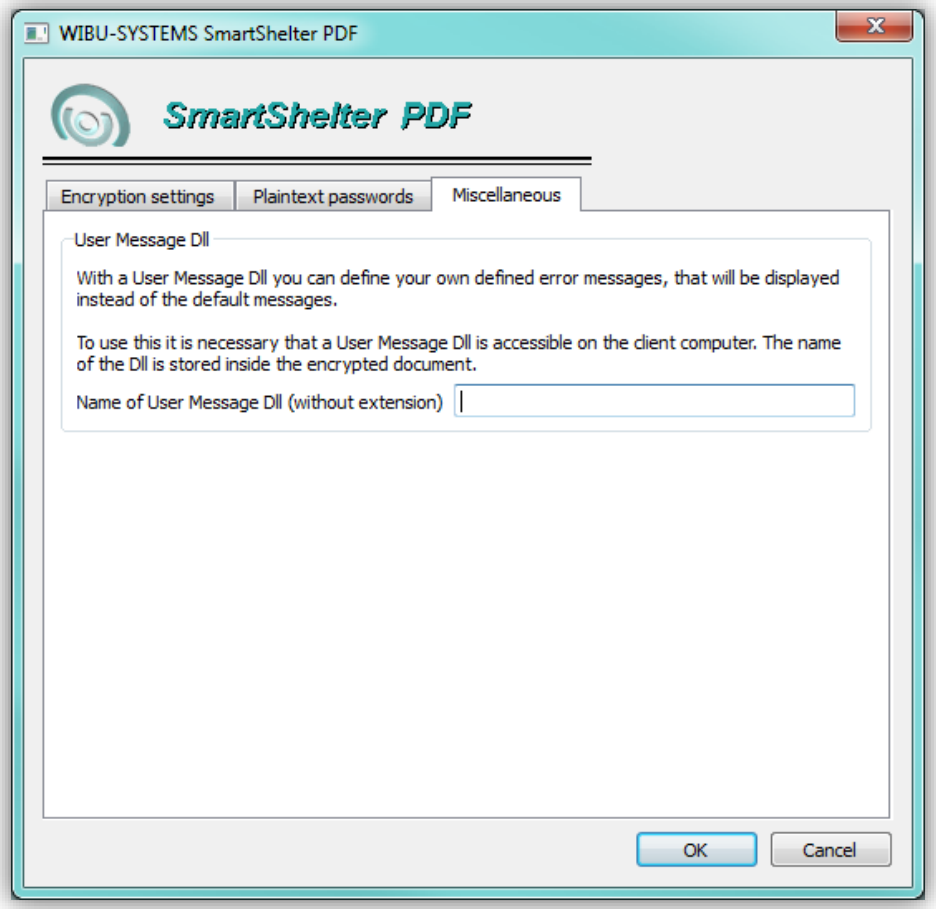
# Protecting PDF Documents (3)



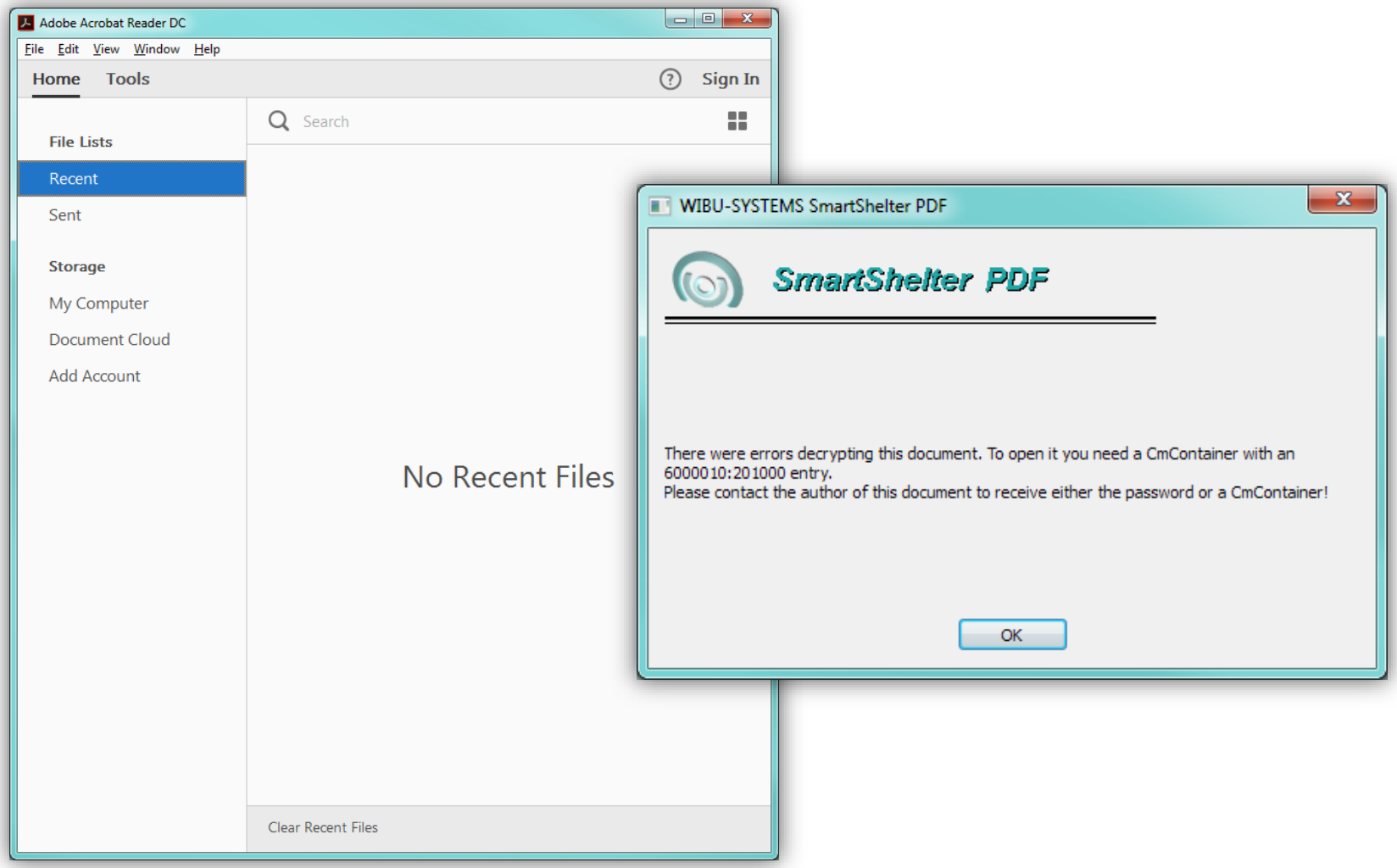



# Protecting PDF Documents (4)

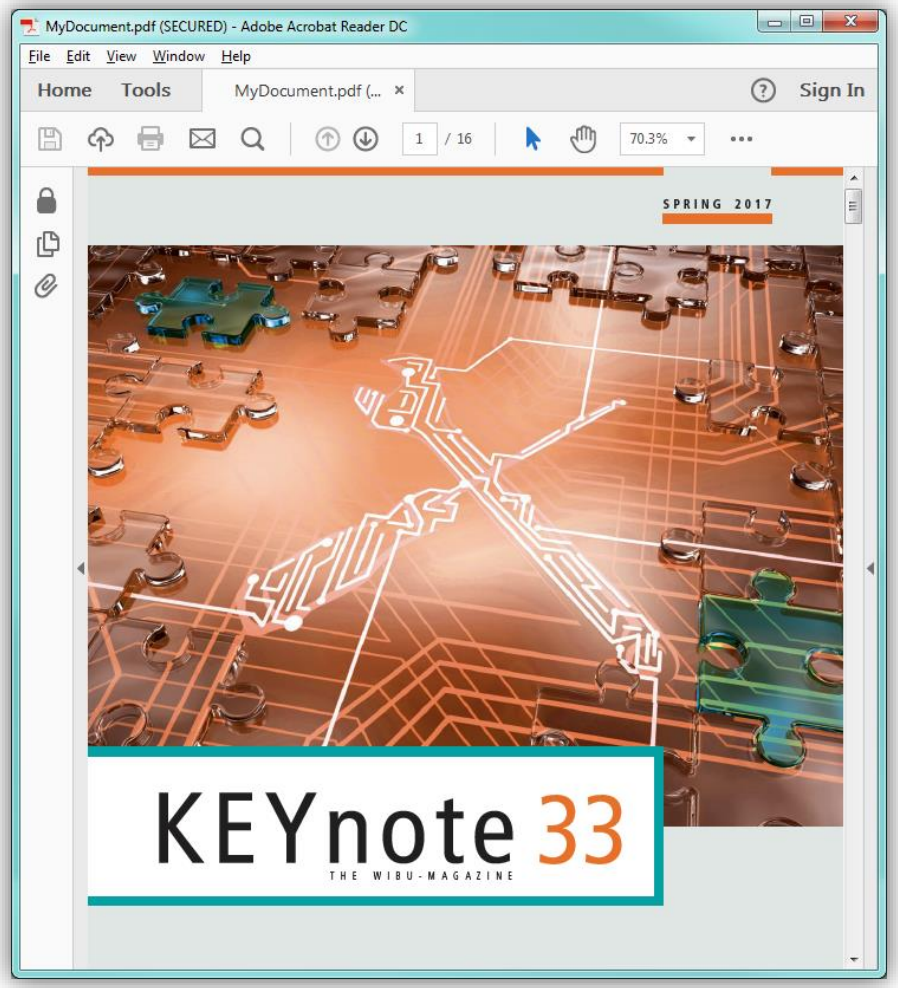


# Open a Protected PDF-Document (No License Available)

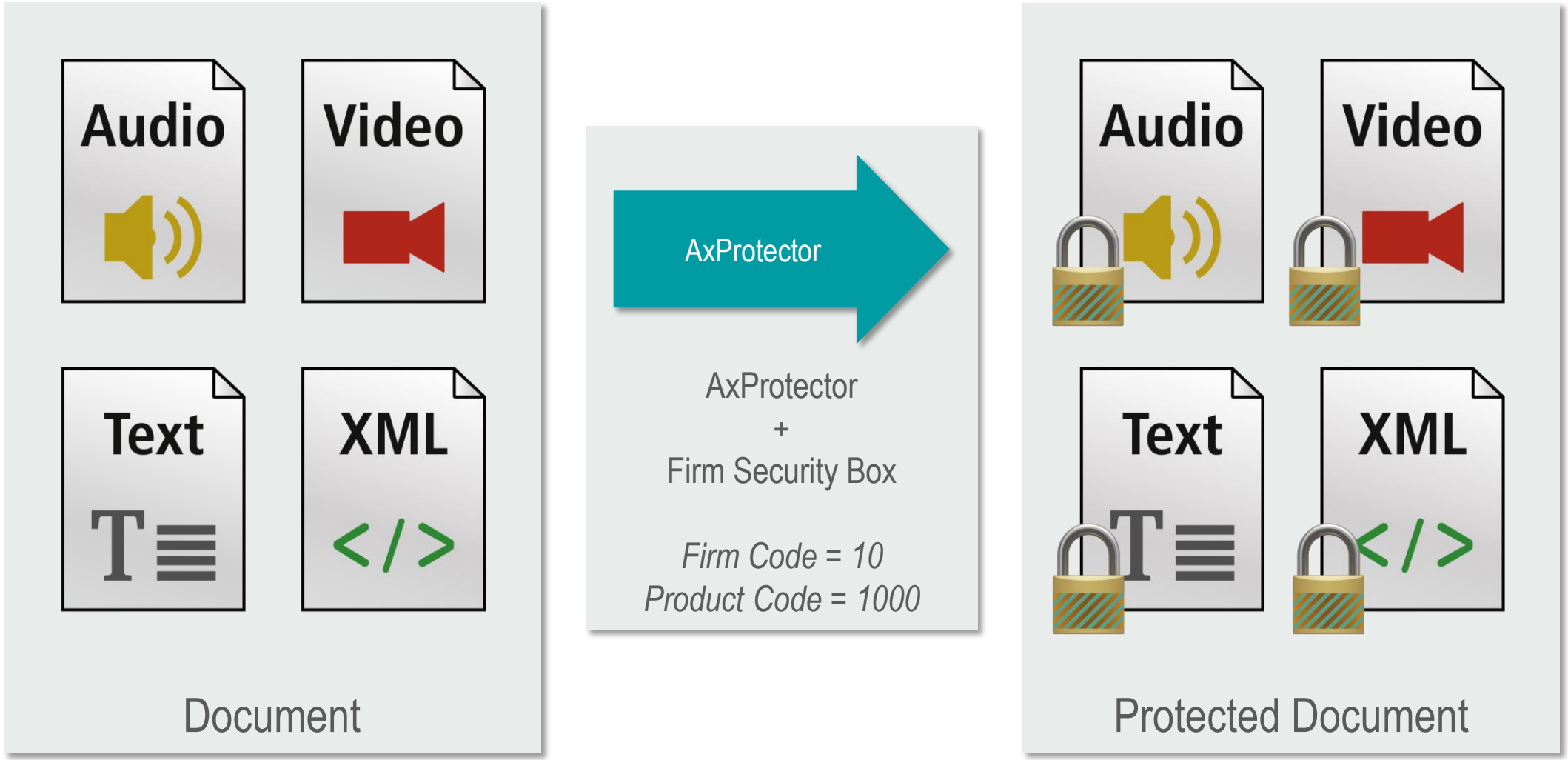


# Open a Protected PDF-Document (License Available)

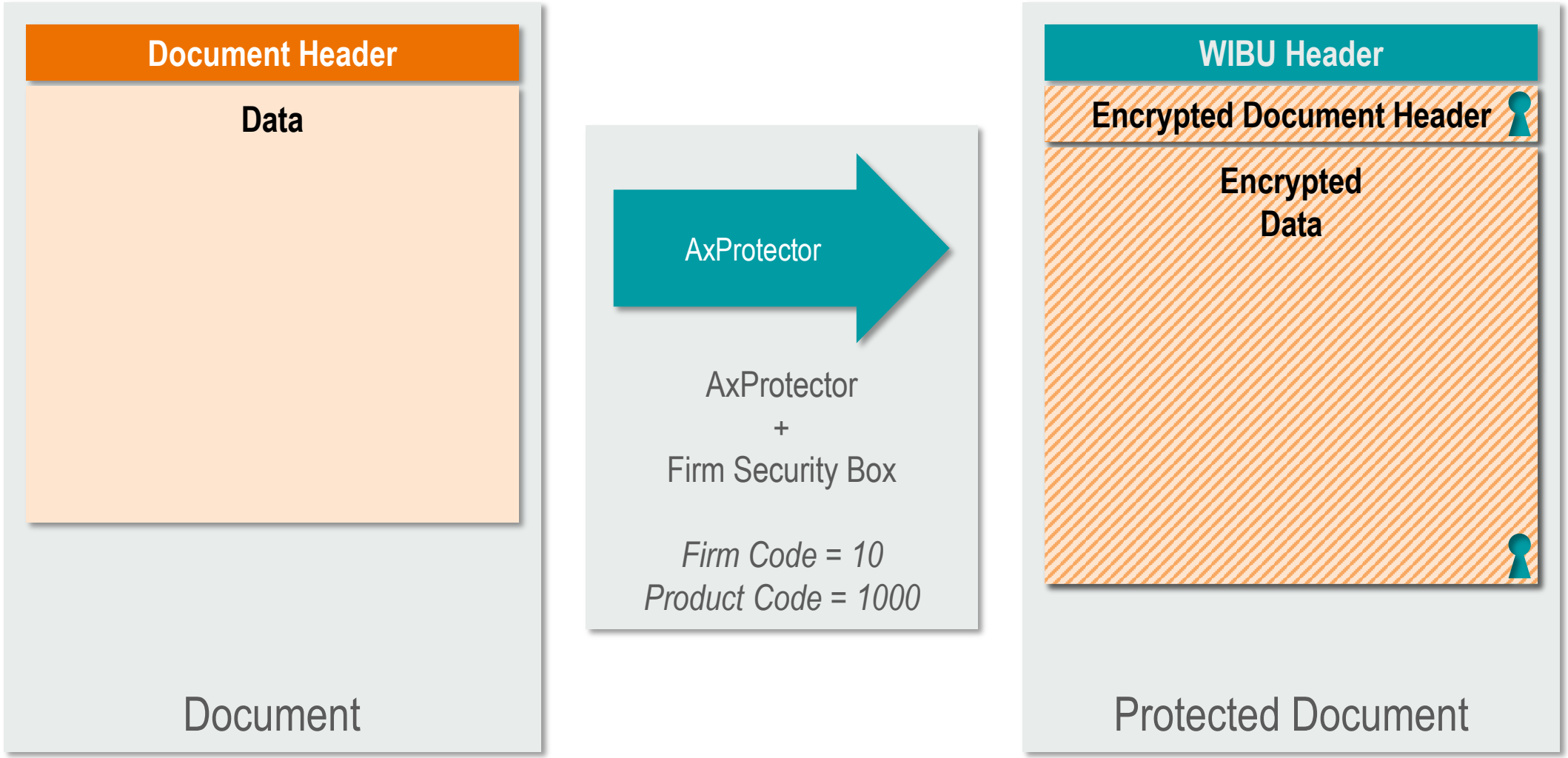


# Standard Documents in a Viewer

# Encrypting a Standard Document



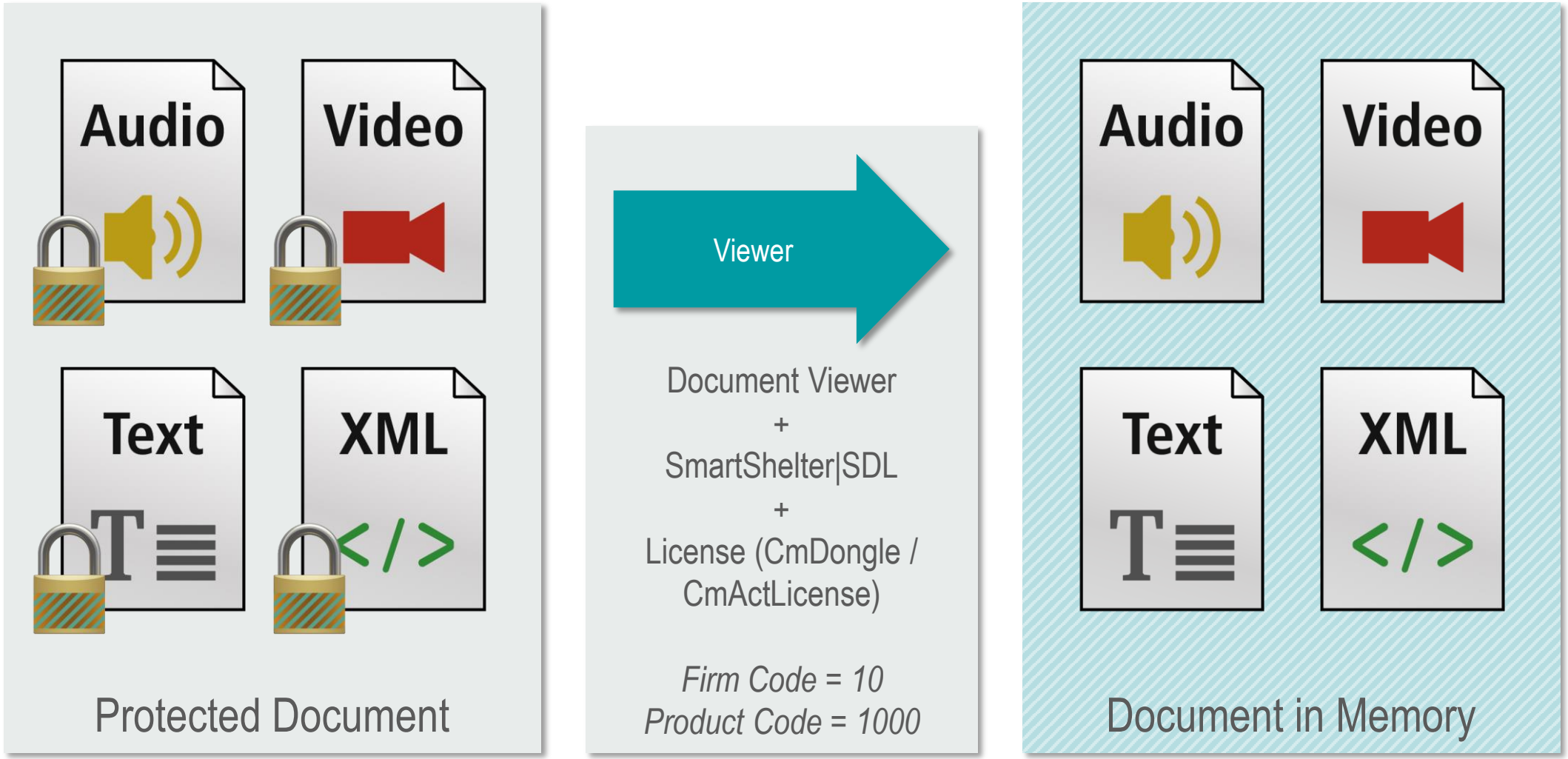
# Encrypting a Standard Document with AxProtector



# Encrypting a Standard Document with AxProtector

- AxProtector fully encrypts the file
  - Including a possibly existing *Document Header*
  - Algorithm: AES, 128 bit
  - The key is derived from Firm Code and Product Code
- AxProtector inserts a specific WIBU Header at the beginning of the protected document
- The decryption can be restricted to specific viewers
  - Professional Services can help with customizations
- AxProtector supports all CodeMeter Product Item Options

# Decrypting a Standard Document



# Decrypting a Standard Document

- SmartShelter|SDL is located between the viewer and the operating system
  - Viewer is encrypted with AxProtector
  - An encrypted .dll gets injected into the process of the viewer (by an SDL-start application)
- SmartShelter|SDL verifies if a valid license (Firm Code / Product Code) is available
- SmartShelter|SDL decrypts the document
  - The user is not aware of the used password
  - The document is decrypted in the memory

# Decrypting a Standard Document

- SmartShelter|SDL is on the alert for unwanted programs closes the application immediately if any unwanted program is detected
- SmartShelter|SDL controls all writing operation of the viewer
  - Unencrypted
  - Always encrypted
  - Writing forbidden
- SmartShelter|SDL provides an interface for customized error dialogs
- SmartShelter|SDL supports all CodeMeter Product Item Options

# Standard Documents

# Writing Standard Documents?

- SmartShelter|SDL can verify if the application is writing files
  - New file: Unencrypted | Writing forbidden | Encrypted
  - Existing file: Condition maintained | Writing forbidden | Encrypted
- SmartShelter|SDL can monitor "*Copy & Paste*"
  - Mode: Always allowed | Always forbidden
- SmartShelter|SDL does not know the context of the application
  - "*Copy & Paste*" and "*Save As*" can only be monitored for all documents unconditionally

# Writing Standard Documents?

- Protecting standard documents is also technically possible, if the application can write these documents
- A strong protection is only possible if the options "*Copy & Paste*" and "*Save As*" are completely disabled
  - Usually not compatible with the requirements in use
  - Knowledge of the internal processes of the application is required (for example, generation of temporary files)

# Proprietary Documents

# Use Cases

- Music recording software
  - My software should always run and record music
  - Recorded music should only be accessible with a valid license
- Demo version
  - My demo version can do anything but:

    a) Saved files cannot be opened

    b) No files may be saved
- Individual configuration files
  - My customers are not allowed to exchange configuration files

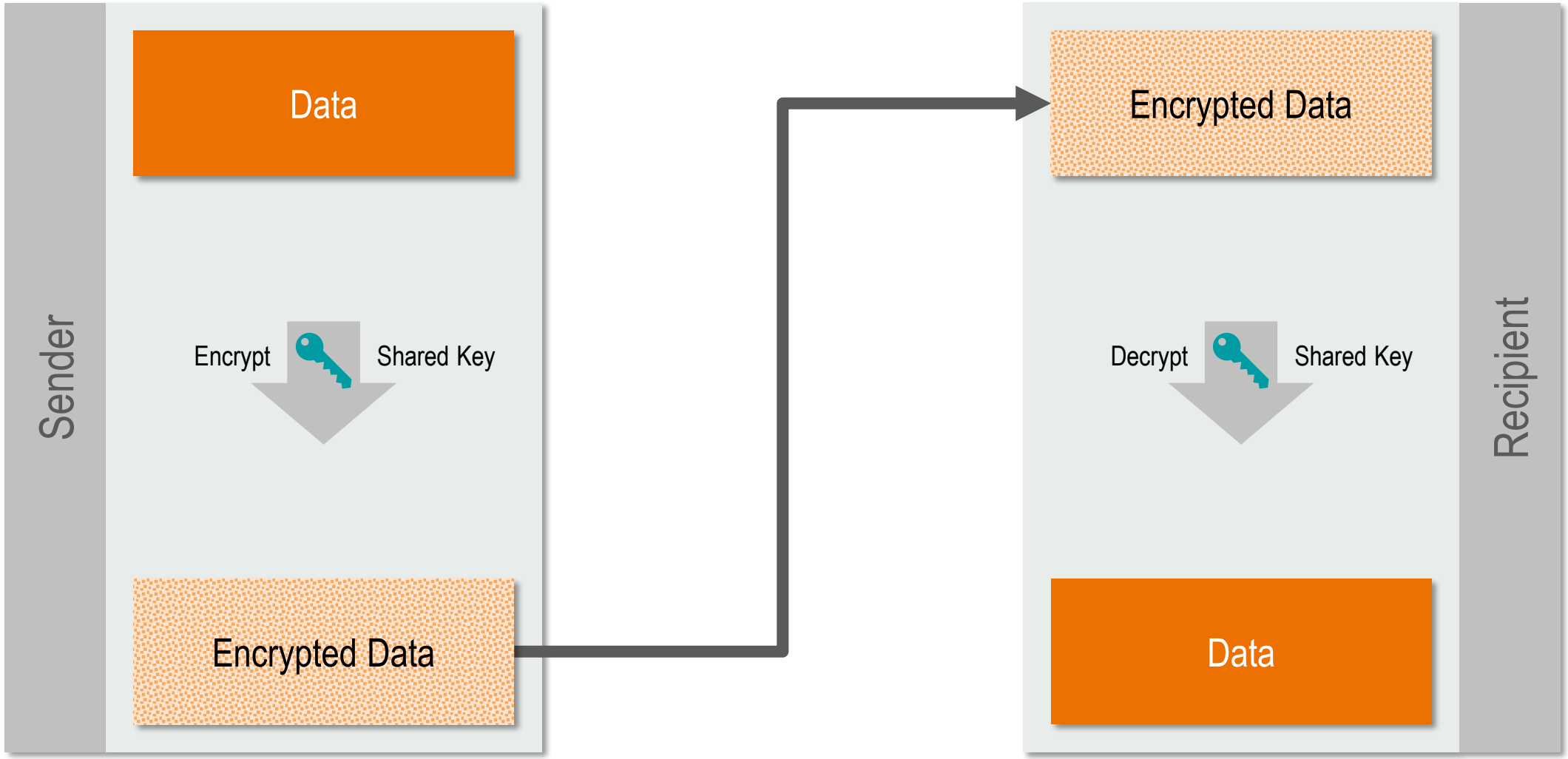
# Use Cases

- Protection of user's data
  - My users should be able to protect files themselves because these files contain sensitive and trustworthy data
- Data from an authorized partner
  - My software should only be able to process data that was created with my software or the software of an authorized partner
- Data for authorized partners
  - The data created with my software may only be processed with my software or that of an authorized partner

# The Solution

- CodeMeter provides top-of-the-breed cryptography
- CodeMeter provides secure key storage
- Solution approaches:
  - A common symmetric key for all
  - An individual symmetric key for everyone
  - Encryption with an asymmetric keypair
  - Signature with an asymmetric keypair

# Symmetric Encryption



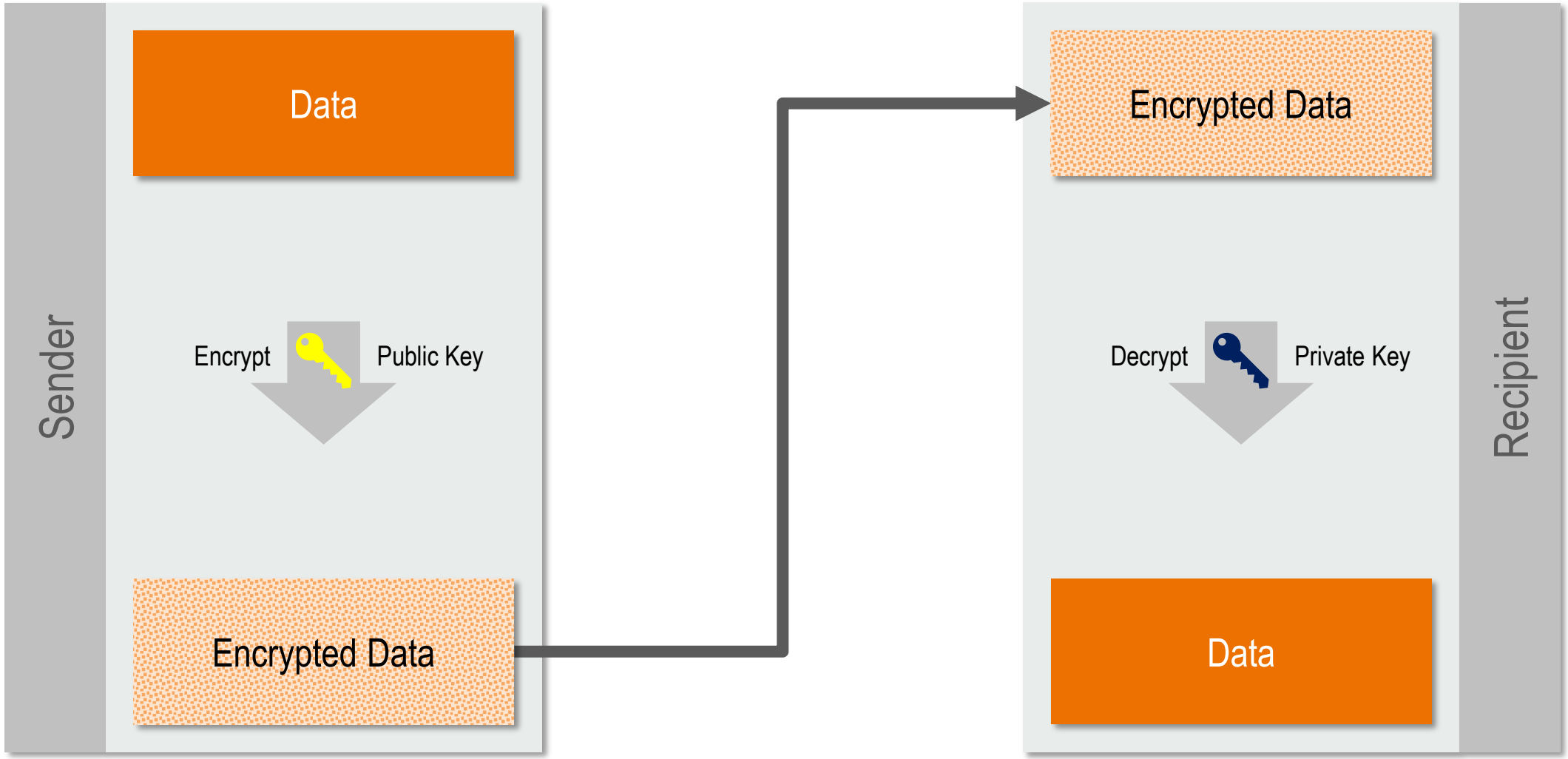
# A Common Symmetric Key for All

- The key is generated automatically via a license entry (Firm Code and Product Code)
- All users who are allowed to decrypt the protected data will receive the same license entry
- Groups of users and even individual users can be assigned different licenses
- Use cases
  - Sales of protected content, and not just software
  - Protection of configuration files

# An Individual Symmetric Key for Everyone

- Individual key for every user
  - A separate license entry for every user
  - A common license entry with randomly selected Secret Data or Hidden Data entries
  - User-specific key, alternatively in separate Firm Code or the Firm Code of the vendor
- Use cases
  - Individually protected configuration files
  - Protection of user's data

# Asymmetric Encryption



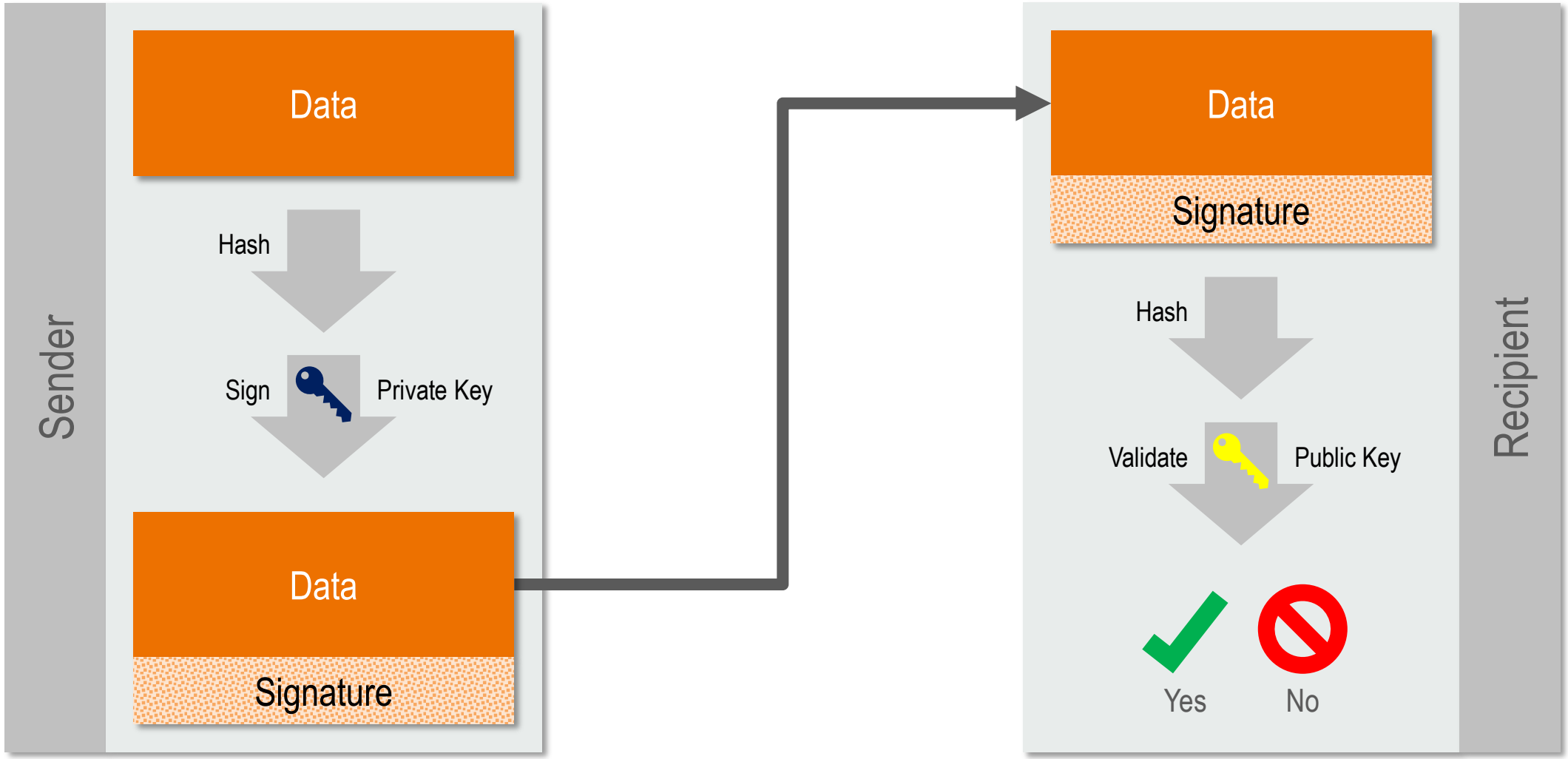
# Generation of an Asymmetric Keypair

- A keypair is generated and the private key is stored securely in the CmContainer
  - The manufacturer generates the keypair
  - The user generates the keypair
- The public key is known to the system
  - As a mini-certificate with root key (public key) in the application
  - Hard-coded in the application
- The mini-certificate is signed with an individual root key (private key)

# Encryption with an Asymmetric Keypair

- Encryption with the public key is always possible
- Decryption is possible only if a valid license with the matching private key is available
- Use cases
  - Data encryption with a demo version
  - Music recording software
  - Data may only be processed by the software of an authorized partner who holds the correct license

# Signature



# Generation of an Asymmetric Keypair

- A keypair is generated and the private key is stored securely in the CmContainer
  - The manufacturer generates the keypair
  - The user generates the keypair
- The public key is known to the system
  - As a mini-certificate with root key (public key) in the application
  - Hard-coded in the application
- The mini-certificate is signed with an individual root key (private key)

# Signature with an Asymmetric Keypair

- The signature can only be created if a valid license with the matching private key is available
- The verification of the signature with the public key is always possible
- The authentication of the public key can be ensured with a mini-certificate
- Use cases
  - Protection of application data against tampering
  - Secure (engine) journal
  - Only data from an authorized partner is processed

# Summary

# Summary

- PDF Files
  - Higher protection standards with SmartShelter|PDF
- Standard Documents
  - Easy integration into a viewer and high protection with SmartShelter|SDL
  - Integration into an Office application only under certain conditions
- Proprietary Documents
  - Use of CodeMeter Core API
  - Versatile applications

WIBU SYSTEMS

# Thank you very much for your attention!


Germany: +49-721-931720

USA: +1-425-7756900

China: +86-21-55661790

http://www.wibu.com

info@wibu.com