PHILIPS

sense and simplicity

White-Box Cryptography

State of the Art

Paul Gorissen

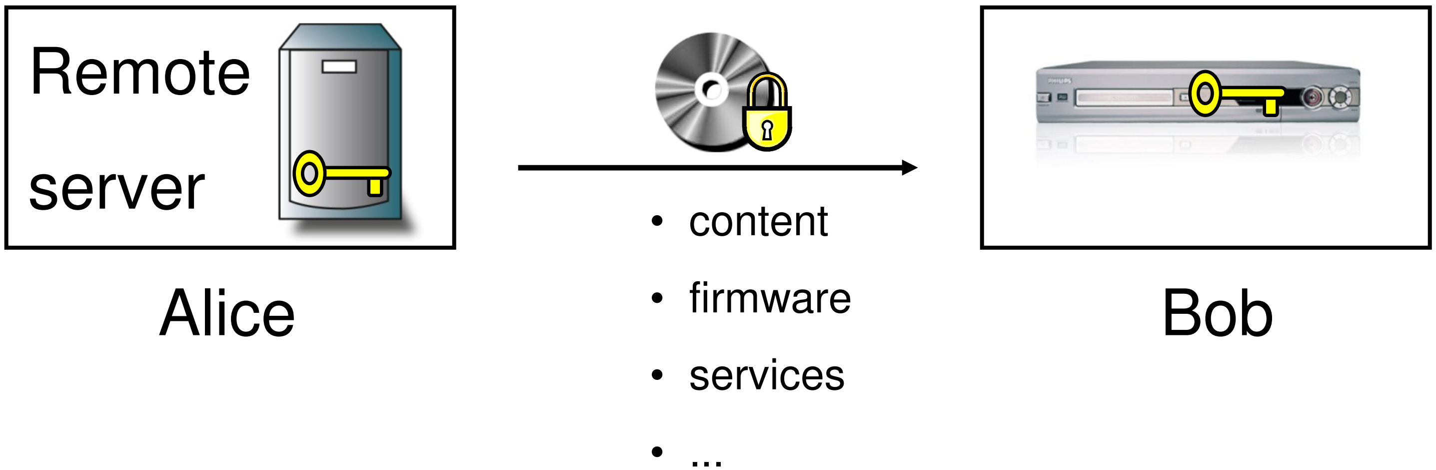
paul.gorissen@philips.com

PHILIPS

Outline

- Introduction
  - Attack models
- White-box cryptography
  - How it is done
  - Interesting properties
  - State of the art
- Conclusion

Introduction

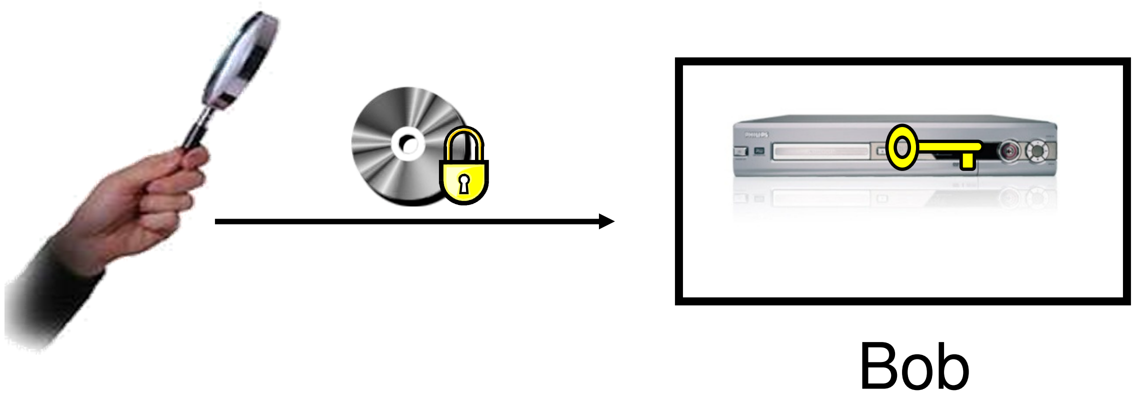


- One generally encrypts data to protect it from malicious use.
- To get the key, the device will be attacked instead of the link.
- Problem to be solved: How to protect a key on a device.
- Solution depends on attack model:

  to how much information does an attacker have access.

PHILIPS

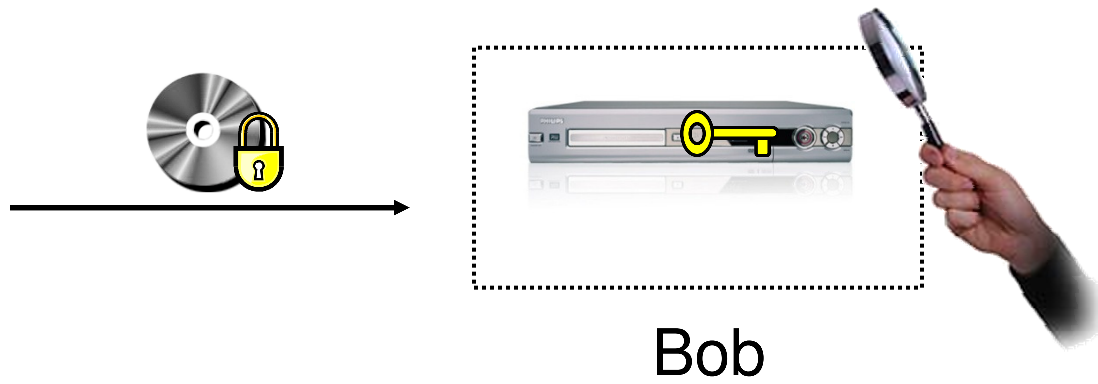
Black-box attack model



- Computation cannot be observed (device is a black box)
- Only the communication link is observable

- Assumptions may be too strong if communicating parties are not trusted.

Grey-box attack model

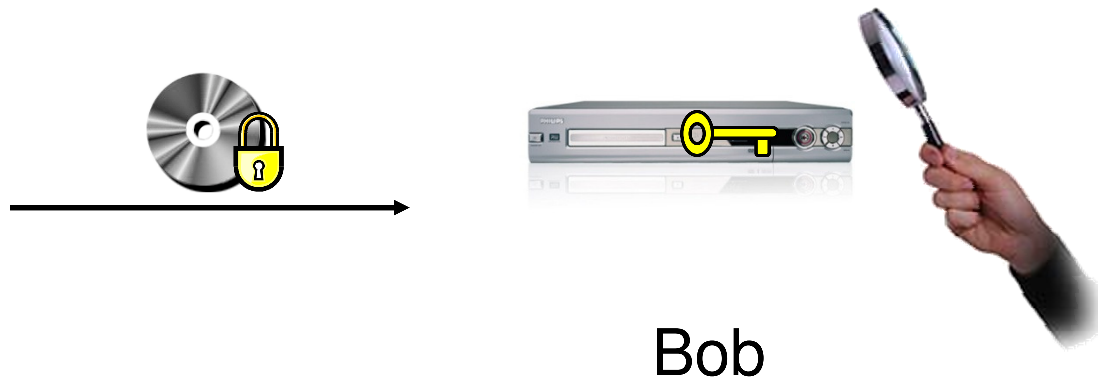


Performance characteristics of computation can be observed

- timing information
- power consumption
- sound

Problem: new types of side-channel attacks are found and published every few months.

White-box attack model



- Computation can be fully observed

  full access to and full control over the device

- Observation:

  if we have a secure implementation in this model, we are automatically secure against all possible side-channel attacks.

White-box cryptography

- Prevent an attacker from extracting the key from a software program that implements a cryptographic algorithm.
  - The key cannot be extracted by analyzing the code
  - The key cannot be extracted by analyzing the intermediate results during execution.

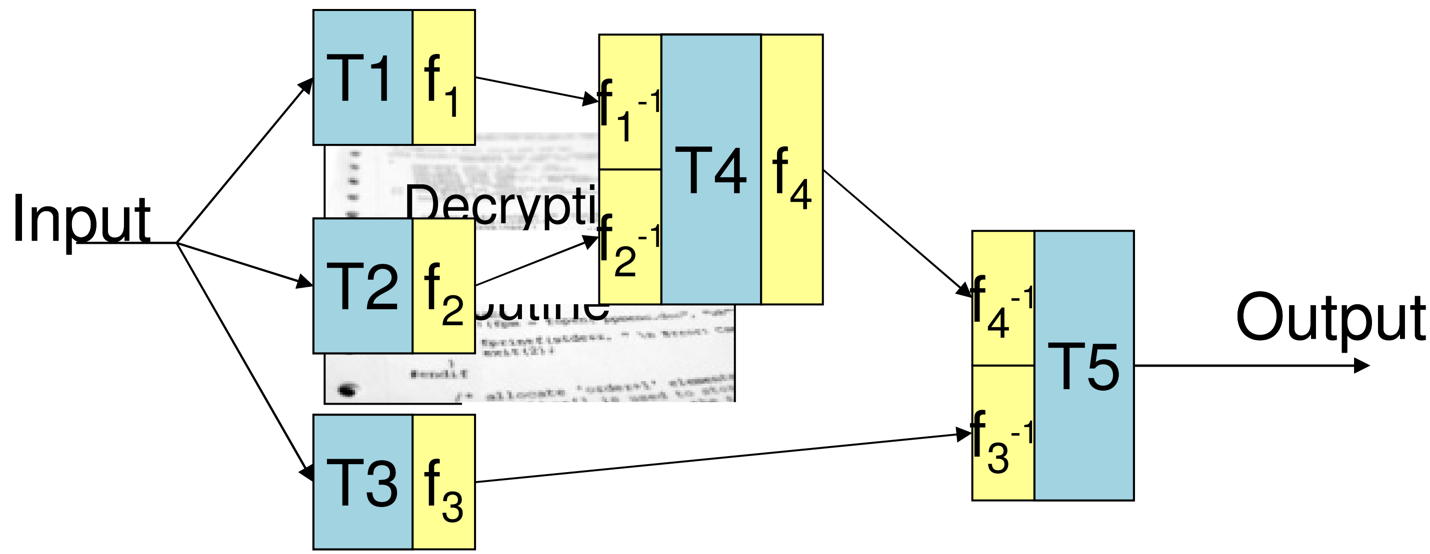
- while the attacker is assumed to have full access to the software implementation and full control over the execution environment (white-box attack model).

PHILIPS

Outline

- Introduction
  - Attack models
- White-box cryptography
  - How it is done
  - Interesting properties
  - State of the art
- Conclusion

White-Box Cryptography – How it is done

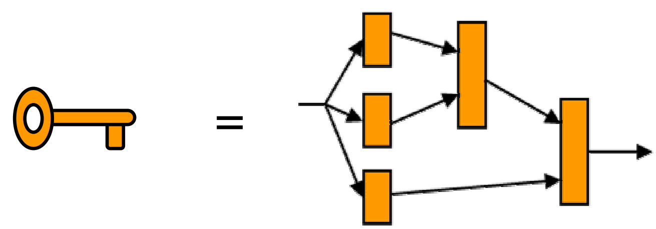


- A recipe for a white-box implementation of a symmetric block cipher
  - Convert the cipher to a lookup table implementation

    Key and algorithm are merged in lookup tables
  - Obscure the network of tables
  - Obfuscate inputs and outputs of each table

White-Box Cryptography – White-Box Key

- Usually white-boxing is achieved by hiding the secret key ("classic key") in a larger bit-string ("white-box key")



  - Symmetric ciphers: key is hidden in implementation of the algorithm
  - Asymmetric ciphers: key is replaced by a larger, equivalent key

PHILIPS

Outline

- Introduction
  - Attack models
- White-box cryptography
  - How it is done
  - Interesting properties
  - State of the art
- Conclusion

PHILIPS

Interesting property 1: side channel attacks

- White box implementations can increase the resistance against side-channel attacks (differential power analysis, timing analysis) if the execution is sufficiently randomized.
  - White-box implementations offer many opportunities for randomizing the execution

- White-box implementations can increase the resistance against the exploitation of software bugs and fault injection attacks if the white-box key is much larger than the classic key
  - Typically, one bug or fault will recover only a small part of the key, and it becomes increasingly difficult to find enough bugs to exploit, or different faults to inject, when the size of the key increases.

PHILIPS

Interesting property 2: asymmetry

- When implementing symmetric ciphers in a traditional way the difference between encryption and decryption is in the algorithm
  - Encryption and decryption key are identical
  - If you know the encryption key, you can also decrypt (and vice versa)
- In white-box implementations the encryption key is different from the decryption key
  - A system that performs encryption cannot be used for decryption (or vice versa)

PHILIPS

Interesting property 3: information binding

- Any arbitrary string of bits can be included in the white-box key
  - string length can be thousands of bits (size has practical limits, no theoretical limit)
  - a modification of the string will destroy the white-box implementation
- Examples of an included bit-string:
  - White-box key can be “locked” to hardware
    - By including hardware characteristics in the white-box key
  - Visible string can be included in the white-box key
    - e.g. “(C) Royal Philips Electronics”, or the name of the customer

```
"A43RS (C)Philips 96GDB" => "A43RS (C)Pirates 96GDB"
"A43RS (C)Philips 96GDB" => "GF45DJ326254UT53BVSA20"
```

  - Hidden (forensic) trace-mark can be put in the white-box key
    - Makes it possible to find the source of a leak (“traitor tracing”)

Forensic key watermarking

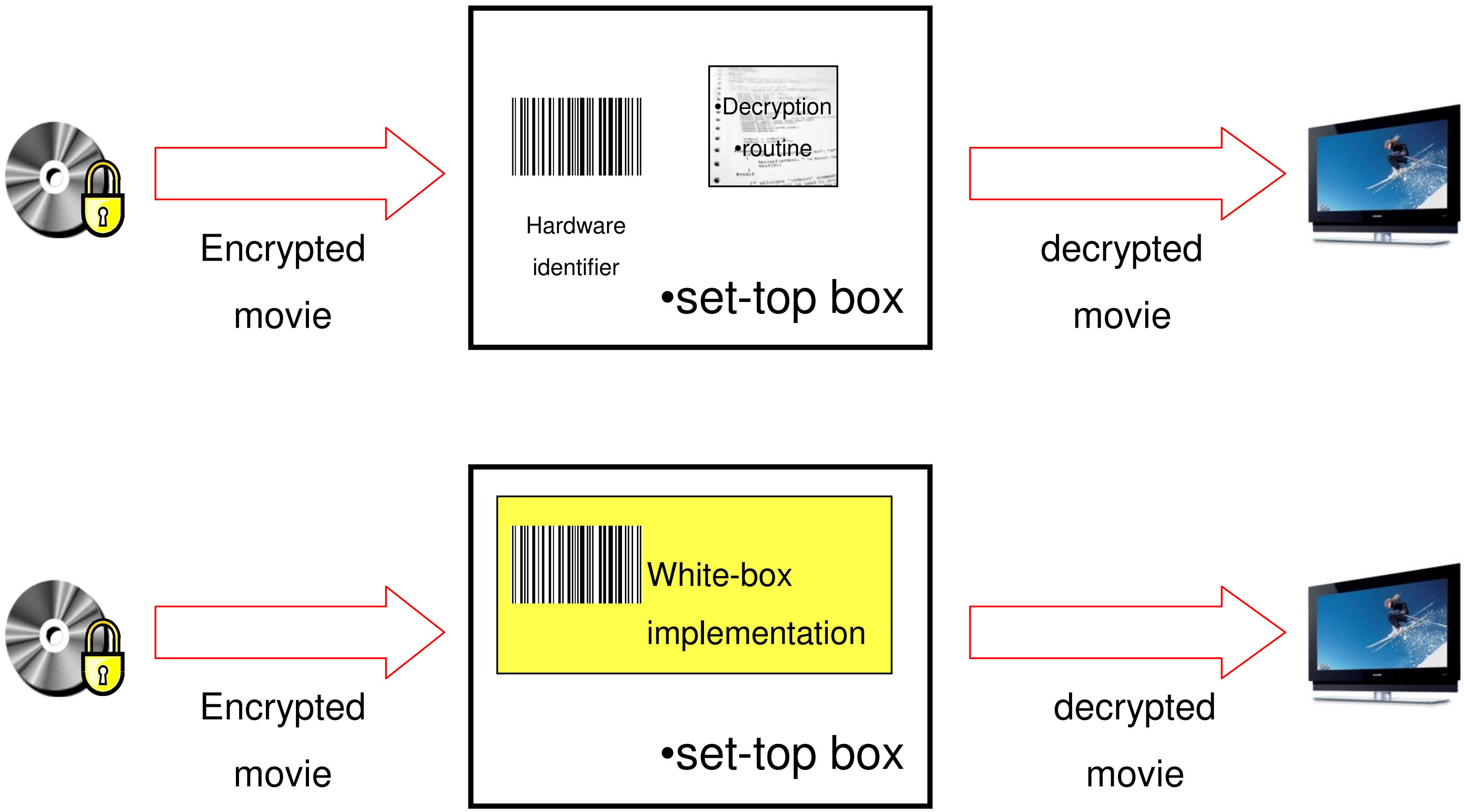
- All white-box implementations have the same cryptographic functionality.
- Include a user identifier in the white-box implementation.
- Each user gets a traceable white-box implementation.

PHILIPS

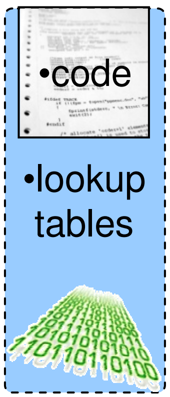
Node locking

- Include a hardware identifier in the white-box implementation.
- Give a user a white-box implementation in which the hardware identifier is omitted.
- The white-box implementation only works on a system with the correct hardware identifier.

Node locking



Software tamper resistance



- Include a binary software image in the white-box implementation.
- Software gets a dual interpretation.
- Changing the code

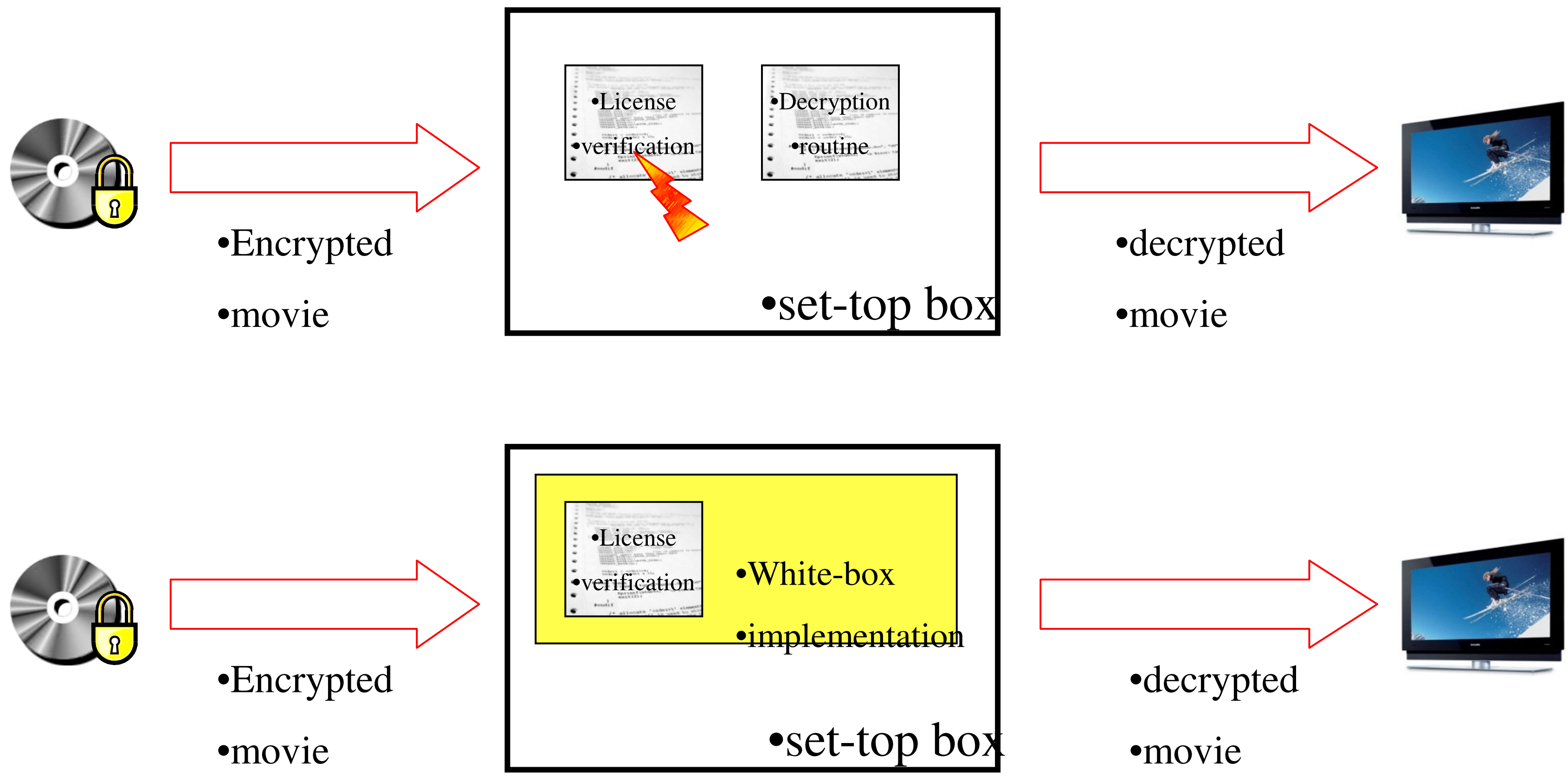
  ⇒

  changing the white-box implementation (key)

  ⇒

  cryptographic operation disabled

Software tamper resistance



PHILIPS

Outline

- Introduction
  - Attack models
- White-box cryptography
  - How it is done
  - Interesting properties
  - State of the art
- Conclusion

State of the art (1) – known white-box methods

- We do not know any publication describing white-box implementations of RSA or ECC
- Several companies offer white-box implementations of RSA and ECC:
  - Cloakware http://www.cloakware.com/
  - Arxan http://www.arxan.com
  - Syncrosoft http://www.syncrosoft.com

State of the art (2) – known white-box methods

- White-box implementations for DES[1] and AES[2] have been published.
- Several companies offer white-box implementations of AES and DES
  - Cloakware http://www.cloakware.com/
  - Syncrosoft http://www.syncrosoft.com

1) Chow, S., Eisen, P., Johnson, H., van Oorschot, P.C.: A White-Box DES Implementation for DRM Applications. Proceedings of the 2nd ACM Workshop on Digital Rights Management, 1-15, 2002.
2) Chow, S., Eisen, P., Johnson, H., van Oorschot, P.C.: White-Box Cryptography and an AES Implementation. Proceedings of the 9th Annual Workshop on Selected Areas in Cryptography, 250-270, 2002.

State of the art (3) - attacks on white-box crypto

- The published white-box implementations of AES and DES have been broken
  - The classic key can be found in 2^{30} time for AES[1] and in 2^{14} for DES[2]
- Philips has shown that standard methods of symmetric cipher construction have fundamental weaknesses for a strong white-box implementation[3]
  - AES and DES are not suitable for applications that need secure white-box implementations
  - New ciphers, or new white-box techniques, are needed to allow secure white-box implementations

1) Billet, O., Gilbert, H., Ech-Chatbi, C.: Cryptanalysis of a White-Box AES Implementation. Proceedings of the 11th Annual Workshop on Selected Areas in Cryptography, 227--240, 2004.

2) Wyseur, B., Michiels, W., Gorissen, P., Preneel, B.: Cryptanalysis of White-Box DES Implementations with Arbitrary External Encodings. Proceedings of the 14th Annual Workshop on Selected Areas in Cryptography, 264--277, 2007.

3) Michiels, W., Gorissen, P., and Hollmann, H.D.L.: Cryptanalysis of a Generic Class of White-Box Implementations, Proceedings of the 15th Annual Workshop on Selected Areas in Cryptography (SAC 2008), 392-406, 2008

State of the art (4) - beyond parlor tricks and obfuscation

- Is it impossible to achieve real security with white-box crypto?
  - It is quite clear that AES and DES cannot be securely white-boxed

- It is possible to construct symmetric ciphers that have the right characteristics to resist the known attacks on white-box methods:
  - Enlarge the key-dependent operations of the cipher by making the diffusion matrix and/or S-box variable and/or key-dependent

  - Use a diffusion operator other than matrix multiplication
    - MDS matrices, for example, should be avoided.

PHILIPS

Outline

- Introduction
  - Attack models
- White-box cryptography
  - How it is done
  - Interesting properties
  - State of the art
- Conclusion

PHILIPS

Conclusions

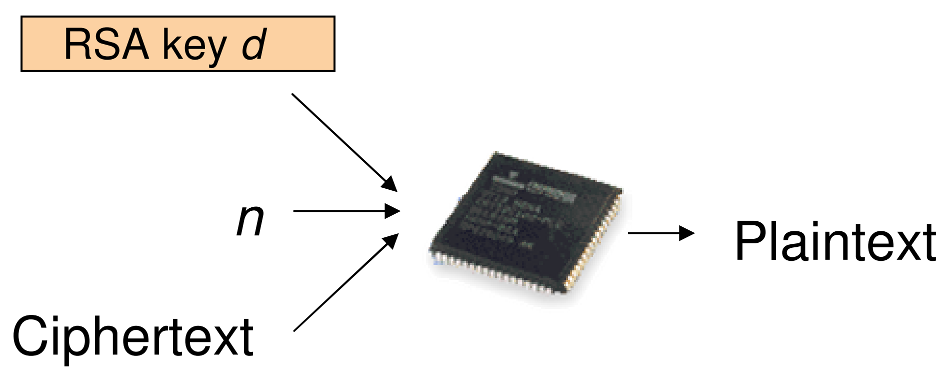
White-box implementations can have useful properties, but beware…

different white-box methods have different security properties

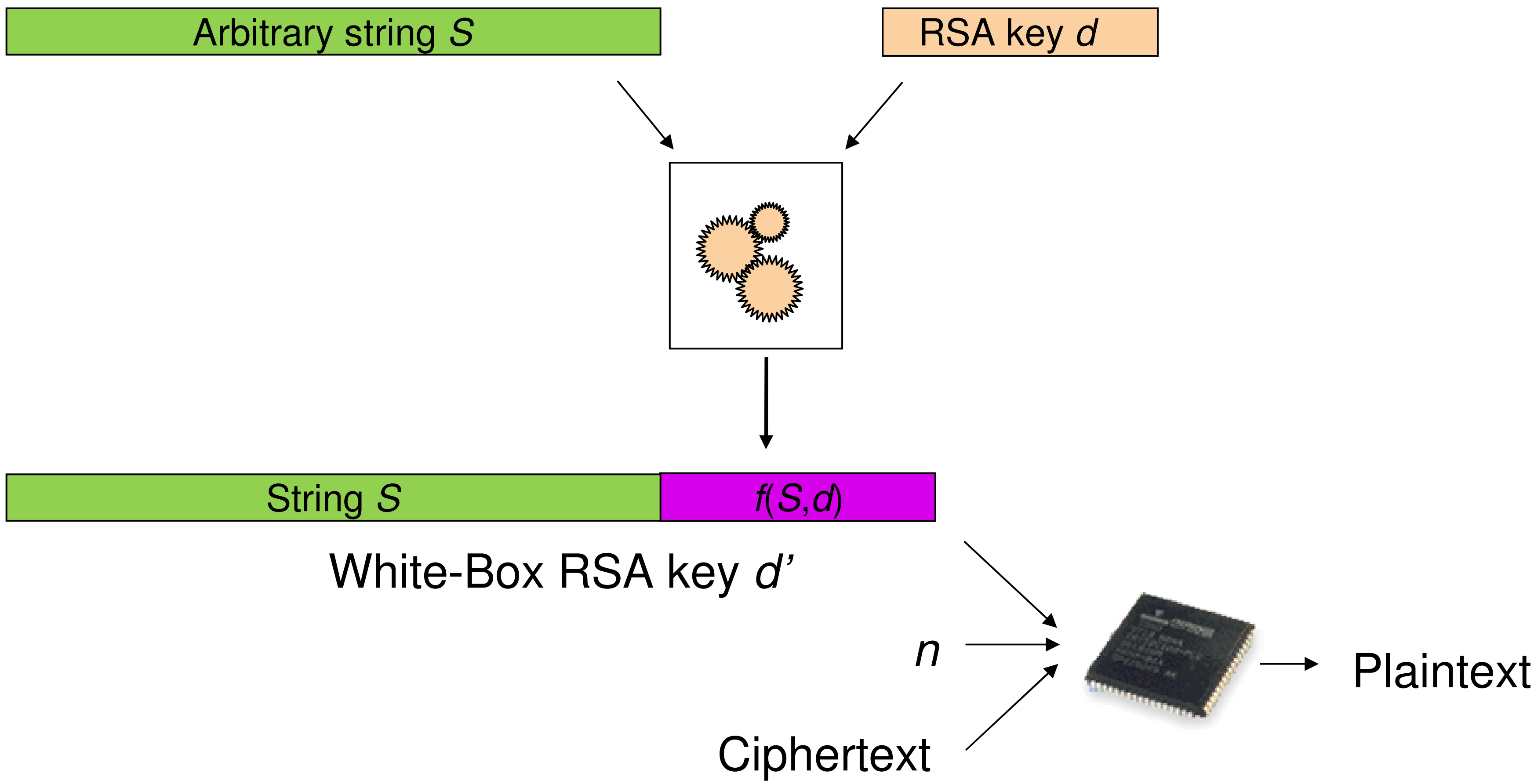
PHILIPS

PHILIPS

RSA Decryption



White-box RSA – Philips' method (1)



White-box RSA: Philips' method (2)

- RSA is a matter of exponents:
  - Choose prime p and q and create $n = p*q$
  - Choose public key e
  - Create private key d
  - Encrypt: ciphertext = $\text{message}^{e} \bmod n$
  - Decrypt: message = $\text{ciphertext}^{d} \bmod n$
- Expand the key d by creating an equivalent larger key
  - New key $d' = d + b * \phi$
    - $\phi = (p-1)*(q-1)$ is the Euler function
  - By a proper choice of b, we can include any bit string S in the binary representation of d'.

White-box RSA: Summary (1)

- RSA key size can be enlarged from k bits (typically 1024-2048 bits) to an arbitrary number of bits
- Linear processing speed reduction
  - Bit length = 10*1024; new speed = old speed/10
- Arbitrary bit strings can be included
  - Maximum size of included bit strings will be on the order of thousands of bits due to practical performance degradation limits