Bitlocker: A little about the internals and what changed in Windows 8

Dan Rosendorf

S.ICZ
Na hřebenech II 1718/10
14700 Praha 4

dan.rosendorf@i.cz

October 1, 2013

BitLocker origins

Birth of BitLocker in 2006

1. Microsoft decides they want a FDE system for Windows Vista
2. Two main considerations: speed and security (and if it's proprietary it probably can't hurt)
3. Decision is made that none of the current algorithms is adequate
4. Niels Fergusson publishes a whitepaper [a] on BitLocker for Windows Vista

[a] Niels Ferguson, *AES-CBC + Elephant diffuser: A disk encryption algorithm for Windows Vista*, 2006

Original Goals:Attacker ability

Attacker

1. The attacker has many known (but not chosen) plaintext/ciphertext pairs for different sectors.
2. The attacker has the ciphertexts for a large number of chosen plaintexts for different sectors. The plaintexts are chosen before the attacker gets access to the laptop.
3. The attacker has access to a slow decryption function for some of the sectors.
4. The attacker gets several ciphertexts of plaintexts for the same sector with a known (but not chosen) difference.

Original Goals:Attacker success

The attacker succeeds if he can modify a ciphertext such that the corresponding plaintext change has some non-random property.

While this is more limited then the usual ideal FDE attacker it is a reasonable attacker for more real-life applications.

The other goal of BitLocker stated in the whitepaper was to provide what Niels Fergusson termed "poor man's authentication".

Algorithm overview

Algorithm

BitLocker as described in the original whitepaper is made up of two key building blocks

1. AES in CBC mode
2. Elephant diffuser - there is a detailed discription of this part of BitLocker in the whitepaper

Algorithm overview

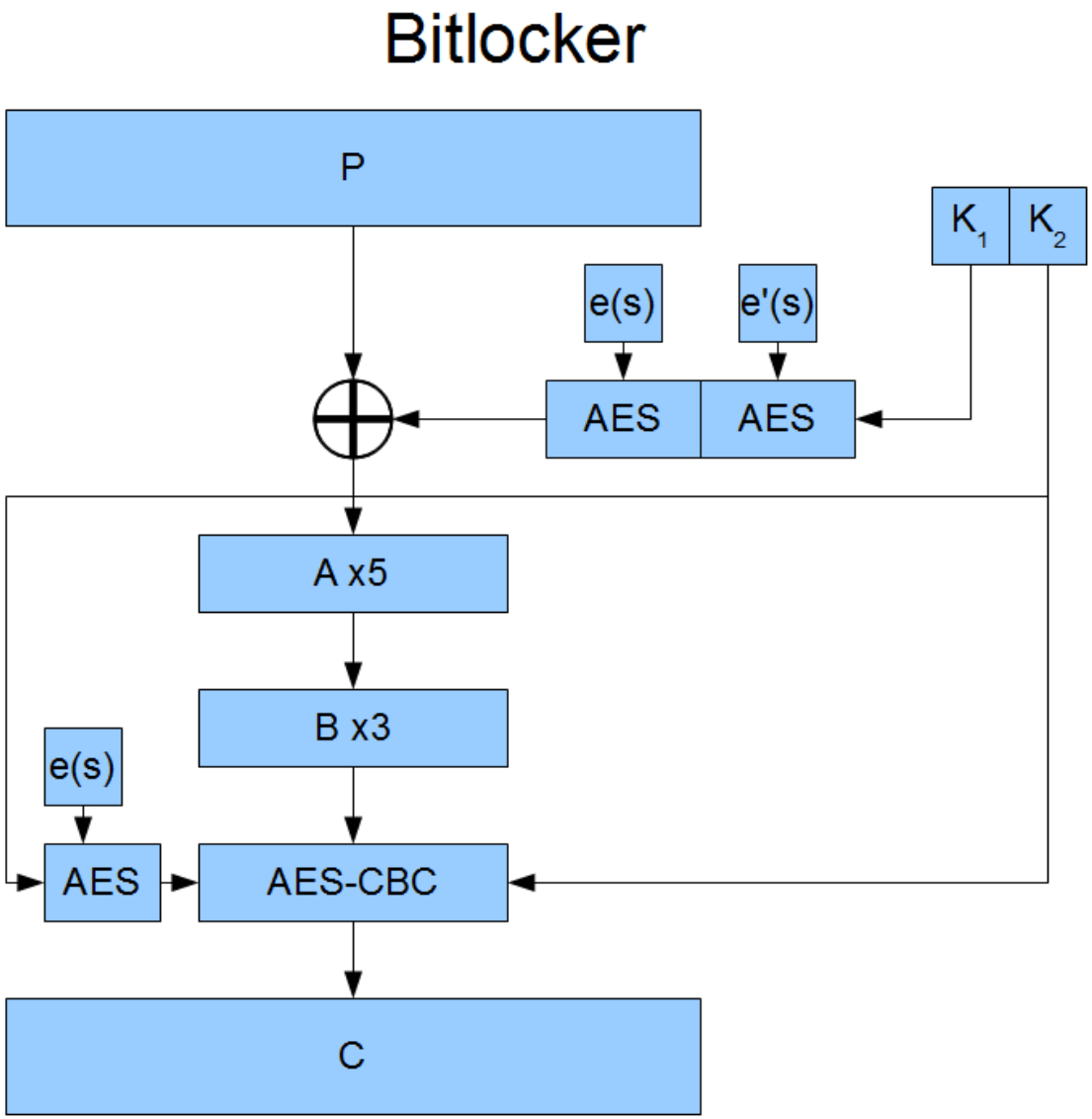
Algorithm

BitLocker as described in the original whitepaper is made up of two key building blocks

1. AES in CBC mode
2. Elephant diffuser - there is a detailed discription of this part of BitLocker in the whitepaper

These two parts are independently keyed by two distinct 128b (or 256b) keys. This ensures that the basic security of AES in CBC mode is not compromised if a vulnerability is found in Elephant diffuser .

Bitlocker algorithm



Usage

What you encrypt

1. Removable drives (called Bitlocker to Go)
2. Non system disk encryption
3. System disk encryption

How you encrypt it

1. Choice of key length 128b or 256b
2. Choice of with or without Elephant diffuser

Keys

Basic types of Keys

1. FVEK - full volume encryption key
2. VMK - volume master key
3. KEK - key encryption key

Keys

Basic types of Keys

1. FVEK - full volume encryption key
2. VMK - volume master key
3. KEK - key encryption key

Types of KEK's

1. TPM key
2. USB key
3. Recovery key
4. Recovery Password (128 bits of real entropy)
5. Password derived key
6. Auto unlock key
7. Certificate

Bitlocker implementation

What Microsoft provides:

1. Enough information to implement the actual algorithm

Bitlocker implementation

What Microsoft provides:

1. Enough information to implement the actual algorithm
2. No information about key generation

Bitlocker implementation

What Microsoft provides:

1. Enough information to implement the actual algorithm
2. No information about key generation
3. No information about key storage

Bitlocker implementation

What Microsoft provides:

1. Enough information to implement the actual algorithm
2. No information about key generation
3. No information about key storage
4. No information about key retrieval/metadata management

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

1. Quite a bit about key storage and metadata from a paper by J. Kornblum[1]

[1] Jesse Kornblum, *"Implementing BitLocker Drive Encryption for Forensic Analysis"*,Digital Investigation, 2009

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

1. Quite a bit about key storage and metadata from a paper by J. Kornblum[1]
2. More information on key storage and a working implementation from a program by R. Coltel[2]

[1] Jesse Kornblum, *"Implementing BitLocker Drive Encryption for Forensic Analysis"*,Digital Investigation, 2009

[2] Romain Coltel,*HSC Tools Dislocker*, http://www.hsc.fr/ressources/outils/dislocker/index.html.en, 2013

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

1. Quite a bit about key storage and metadata from a paper by J. Kornblum[1]
2. More information on key storage and a working implementation from a program by R. Coltel[2]

What we still don't know:

[1] Jesse Kornblum, *"Implementing BitLocker Drive Encryption for Forensic Analysis"*,Digital Investigation, 2009

[2] Romain Coltel,*HSC Tools Dislocker*, http://www.hsc.fr/ressources/outils/dislocker/index.html.en, 2013

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

1. Quite a bit about key storage and metadata from a paper by J. Kornblum[1]
2. More information on key storage and a working implementation from a program by R. Coltel[2]

What we still don't know:

1. How is key generation accomplished?

[1] Jesse Kornblum, *"Implementing BitLocker Drive Encryption for Forensic Analysis"*,Digital Investigation, 2009

[2] Romain Coltel,*HSC Tools Dislocker*, http://www.hsc.fr/ressources/outils/dislocker/index.html.en, 2013

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

1. Quite a bit about key storage and metadata from a paper by J. Kornblum[1]
2. More information on key storage and a working implementation from a program by R. Coltel[2]

What we still don't know:

1. How is key generation accomplished?
2. What happens when key storage fails in unexpected ways?

---

[1] Jesse Kornblum, *"Implementing BitLocker Drive Encryption for Forensic Analysis"*,Digital Investigation, 2009

[2] Romain Coltel,*HSC Tools Dislocker*, http://www.hsc.fr/ressources/outils/dislocker/index.html.en, 2013

Bitlocker implementation

When Microsoft doesn't provide details often the "white/black hat" community steps up. So what we actually mostly know:

1. Quite a bit about key storage and metadata from a paper by J. Kornblum[1]
2. More information on key storage and a working implementation from a program by R. Coltel[2]

What we still don't know:

1. How is key generation accomplished?
2. What happens when key storage fails in unexpected ways?
3. Some key storage information for specific types of keys (e.g. keys generated from password)

---

[1] Jesse Kornblum, *"Implementing BitLocker Drive Encryption for Forensic Analysis"*,Digital Investigation, 2009

[2] Romain Coltel,*HSC Tools Dislocker*, http://www.hsc.fr/ressources/outils/dislocker/index.html.en, 2013

Metadata

Metadata regions

1. There are 4 regions of metadata on a BitLocker encrypted partition/disk
2. The header data: this takes the place of the standard partition header and is 512 bytes long
3. Three key metadata blocks at various offsets. These blocks should all be the same

Metadata

Header block

1. Starts with `-FVE-FS-` or `2D 46 56 45 2D 46 53 2D` in hex
2. Contains version info, 1 for Windows Vista and 2 for Windows 7 and 8
3. Sector size (512 pretty much always it seems)
4. Offsets of the 3 key metadata blocks

Metadata

Header block

1. Starts with `-FVE-FS-` or `2D 46 56 45 2D 46 53 2D` in hex
2. Contains version info, 1 for Windows Vista and 2 for Windows 7 and 8
3. Sector size (512 pretty much always it seems)
4. Offsets of the 3 key metadata blocks

Key metadata block

1. Starts with `-FVE-FS-` or `2D 46 56 45 2D 46 53 2D` in hex
2. Contains offsets for all 3 key metadata blocks
3. Contains keys wrapped in protectors (VMK wrapped in KEK's and FVEK wrapped in VMK)
4. Contains a CRC32 checksum

Key protectors

Key protector

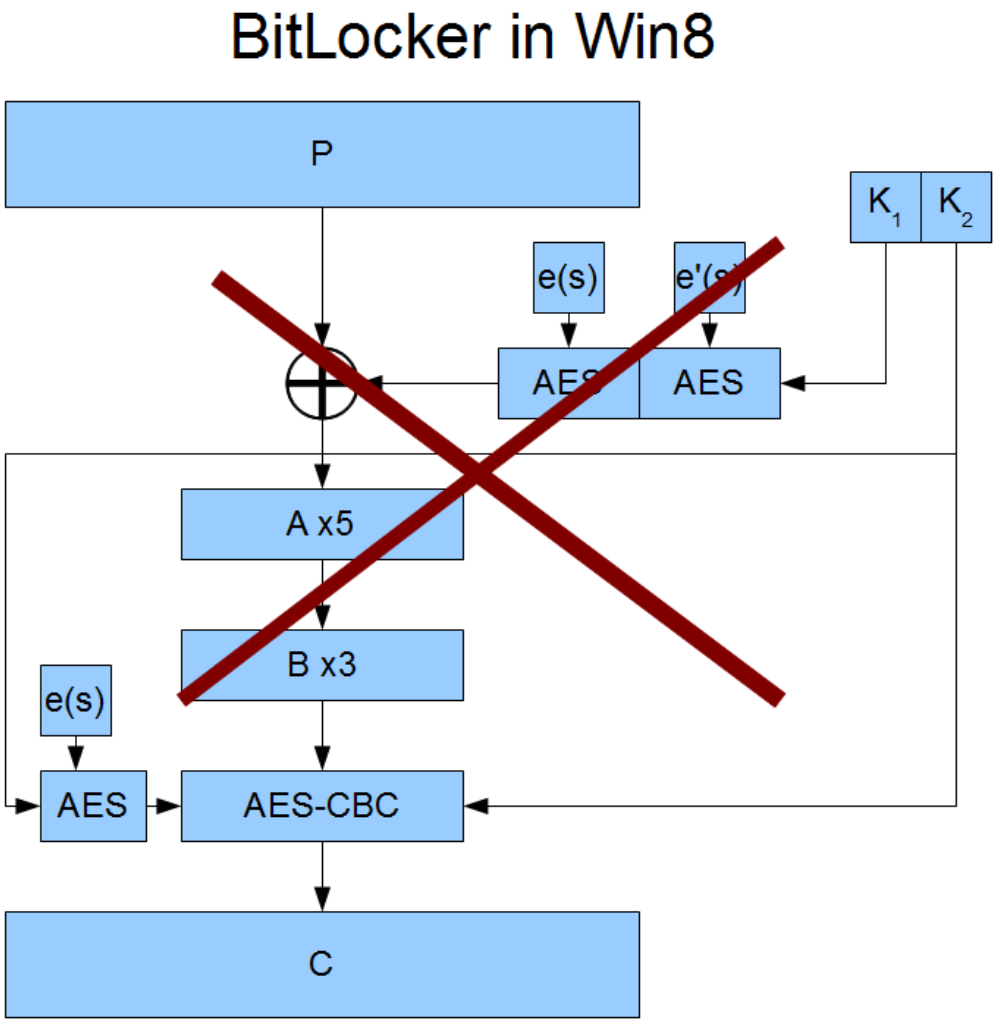
This is a term for the structure in which a key is stored on the disk. It contains the key (usually encrypted) along with metadata such as key usage policies, type of key, key encryption mechanism (if key is encrypted) and hash value.

1. Can be removed and added at leisure
2. Contain timestamp
3. Contain nonce which is nondecreasing and seems to be linear in the number of protectors
4. Contain information about key being wrapped and key used for wrapping
5. Contain wrapped key encrypted using AES-CCM by wrapping key
6. Contain hash of decrypted version of protector
7. Contain wrapping key encrypted by wrapped key (at least in some versions)

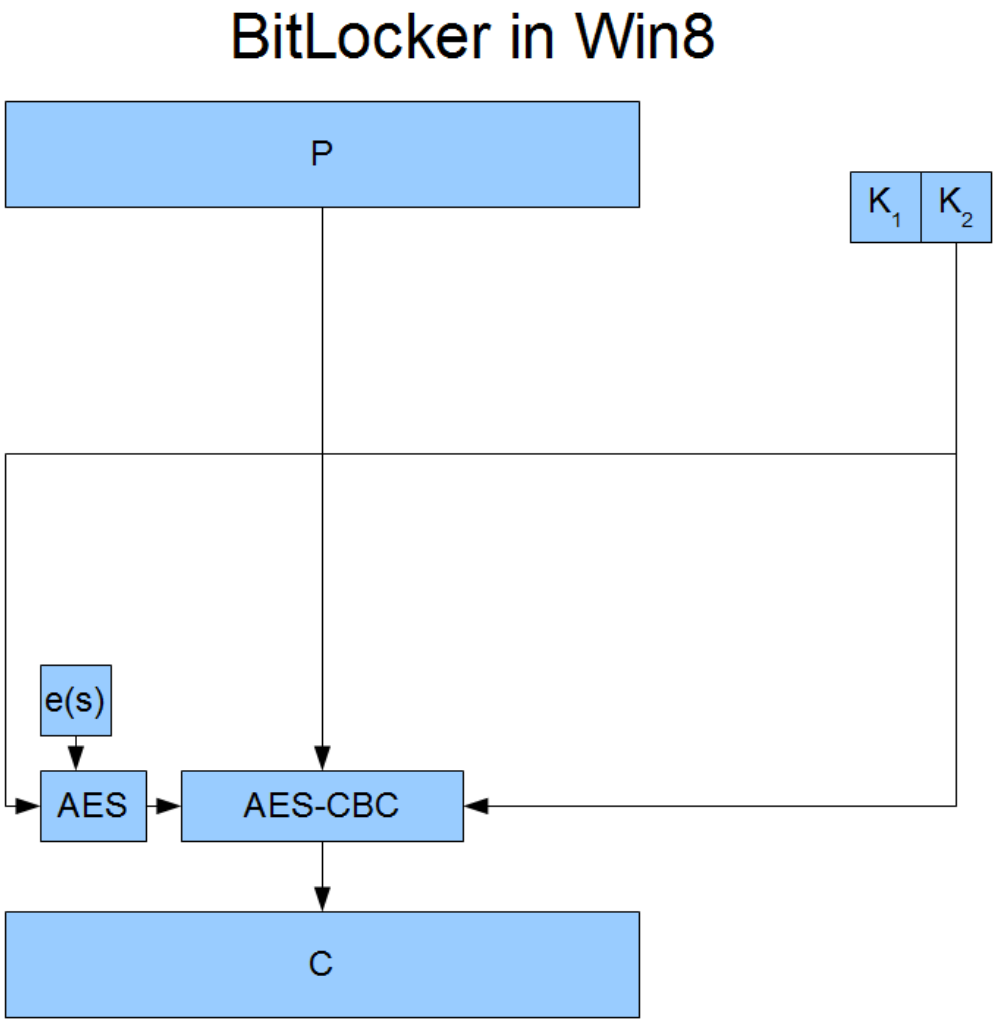
Changes in BitLocker in Windows 8

In Windows 8 a very sparsely documented change to BitLocker has been made. The original BitLocker schema has been changed to no longer include the Elephant Diffuser. This has also had the side effect of no longer having a sector key so the only change between sectors is now given by the IV used for CBC.

Bitlocker in Windows 8

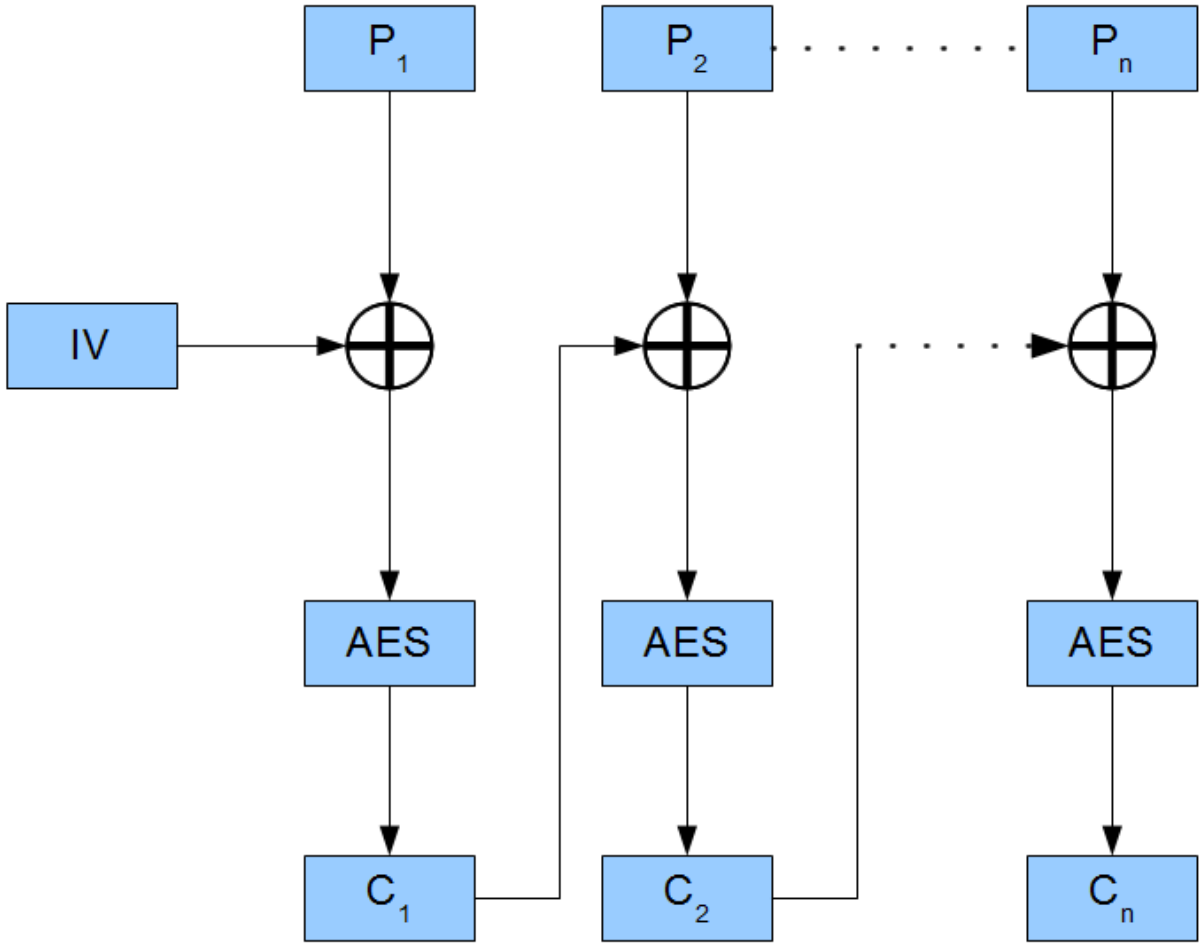


Bitlocker in Windows 8

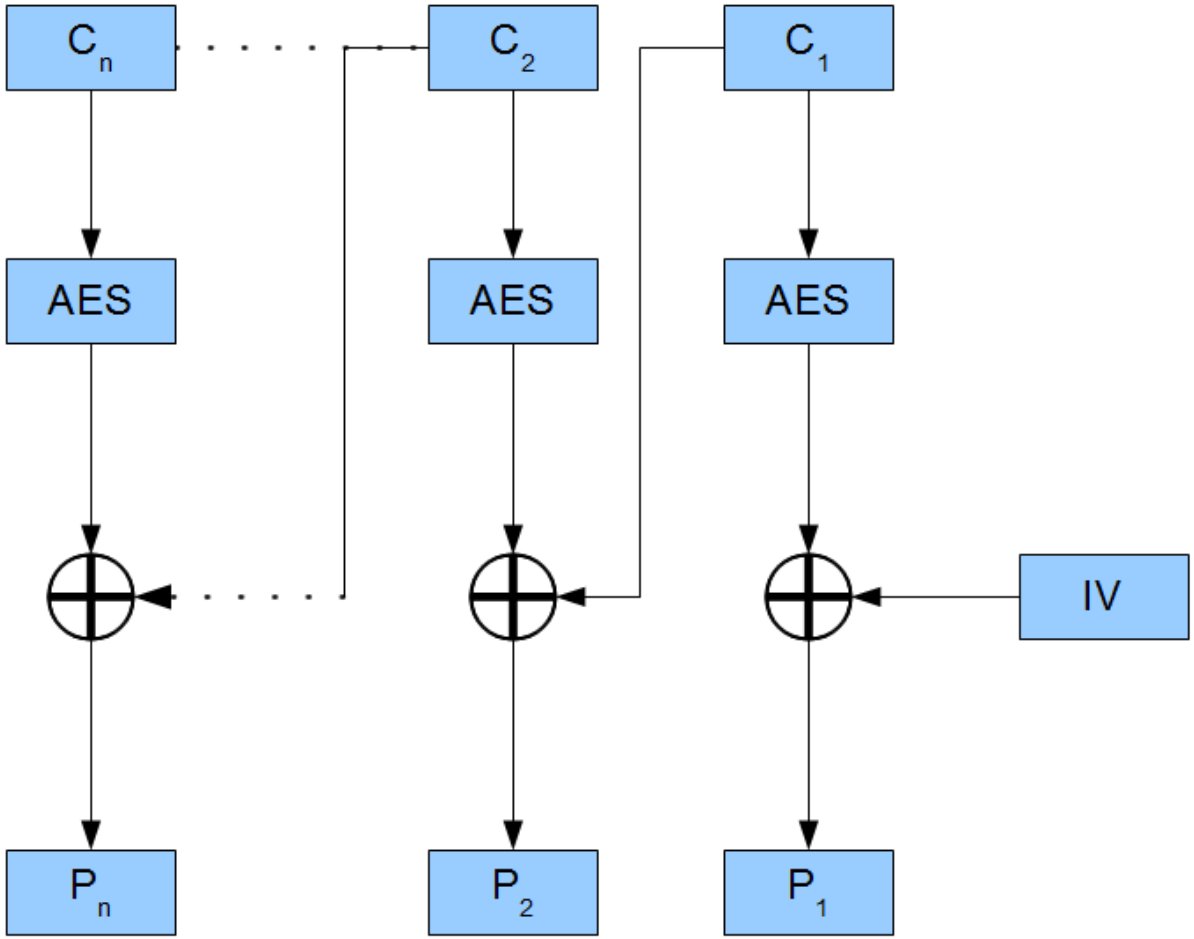


CBC

Bitlocker Win 8= AES v CBC-mode



Dešifrování v CBC módu

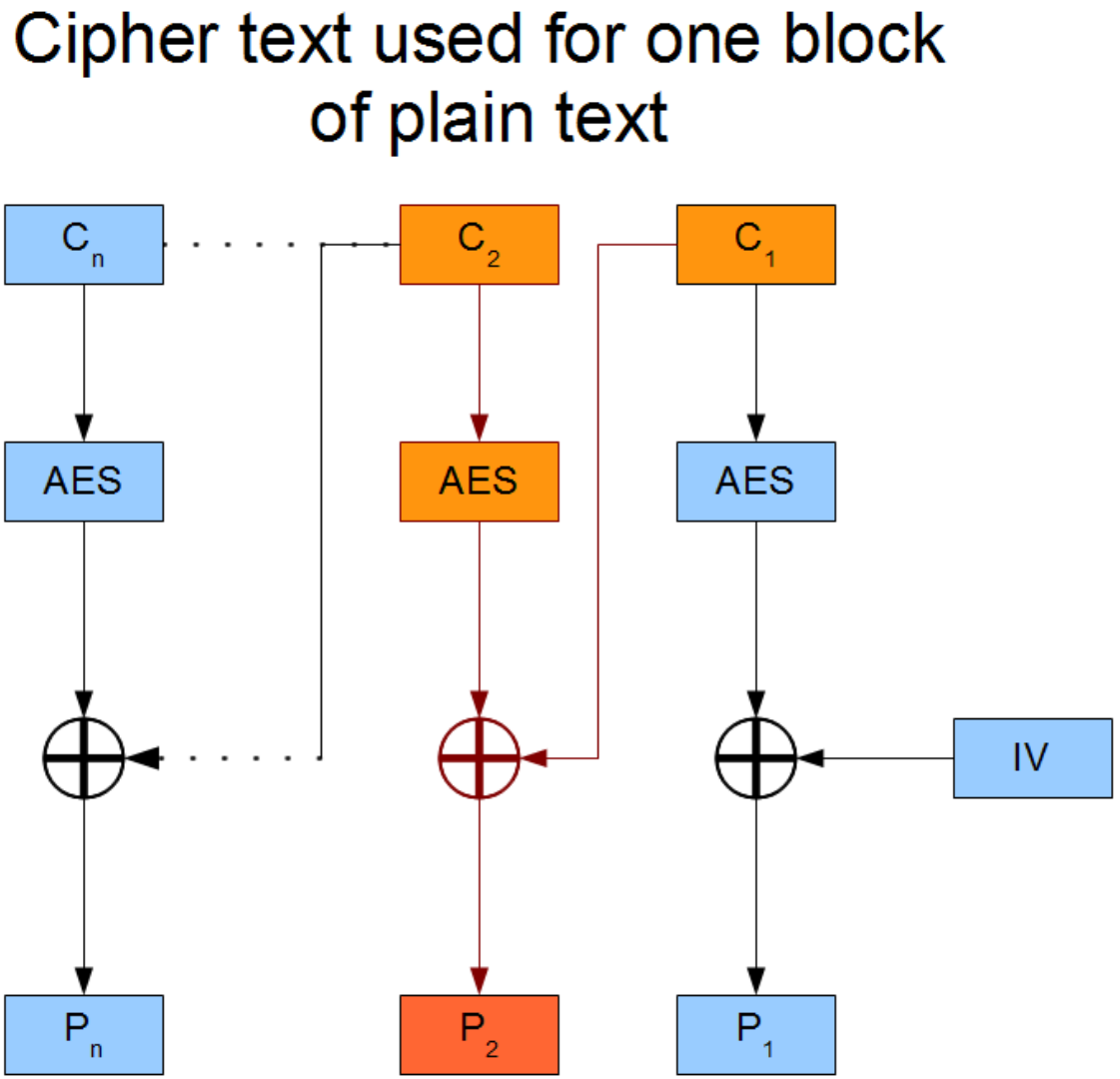


Attacking Bitlocker in Windows 8

All the attacks presented just stem from the use of AES-CBC. It is well known that this method is vulnerable to bit flipping and block exchanging attacks.
In the case of FDE we can use this for two different purposes.

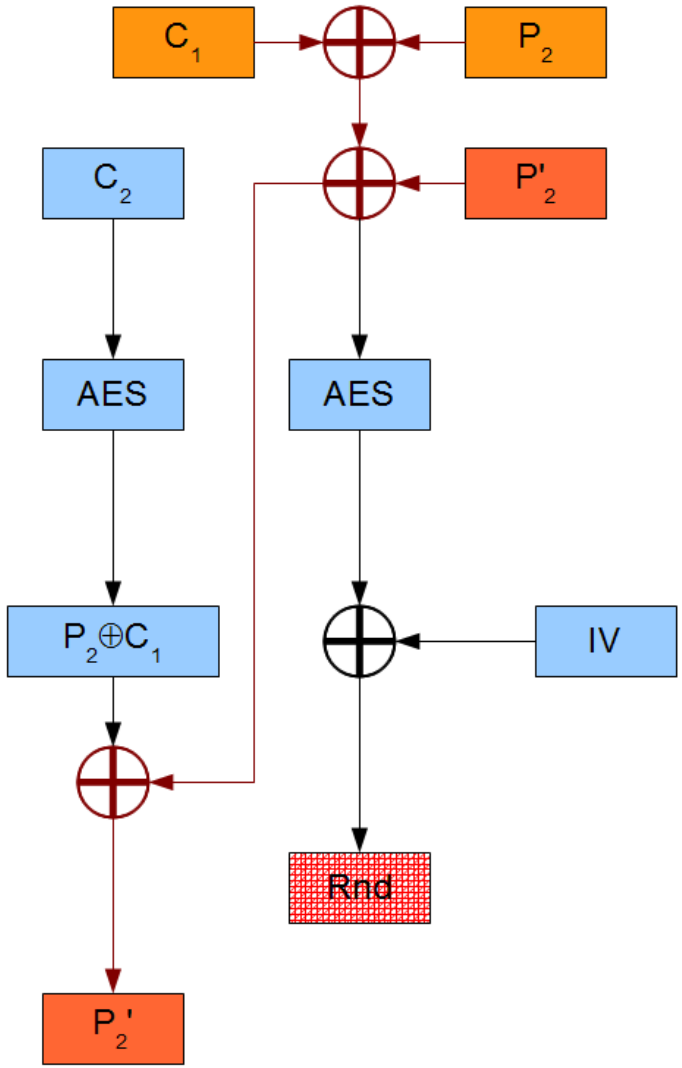
1. Make targeted changes to specific 16 byte blocks
2. Move sectors around without losing too much information after decryption

Decryption dependency



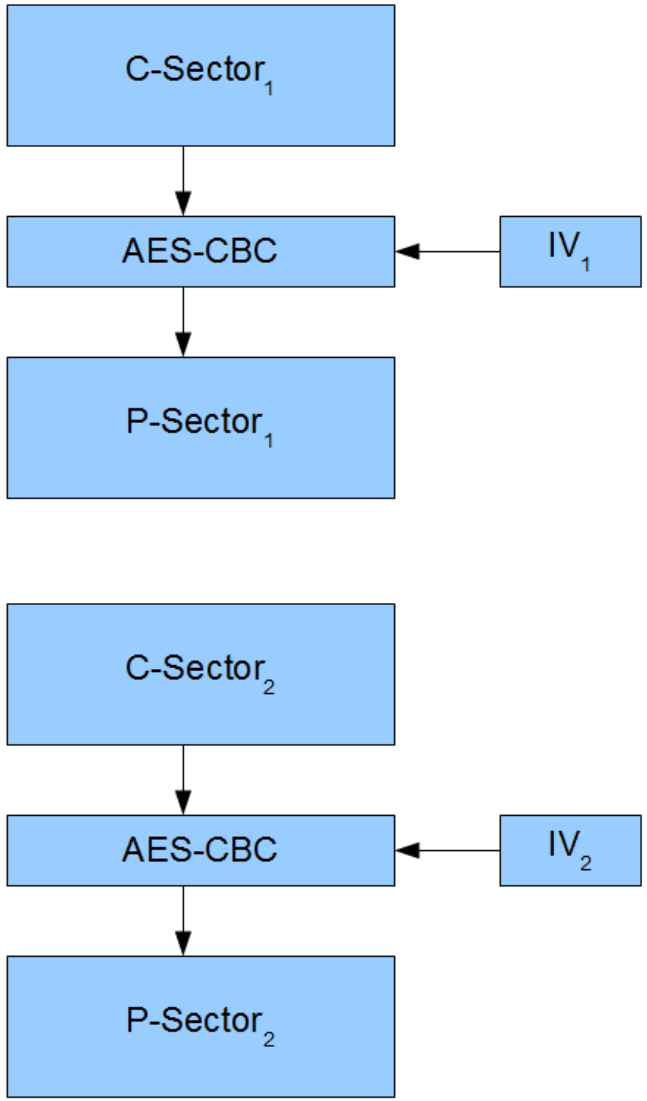
Bit flipping attack

16-byte targeted attack



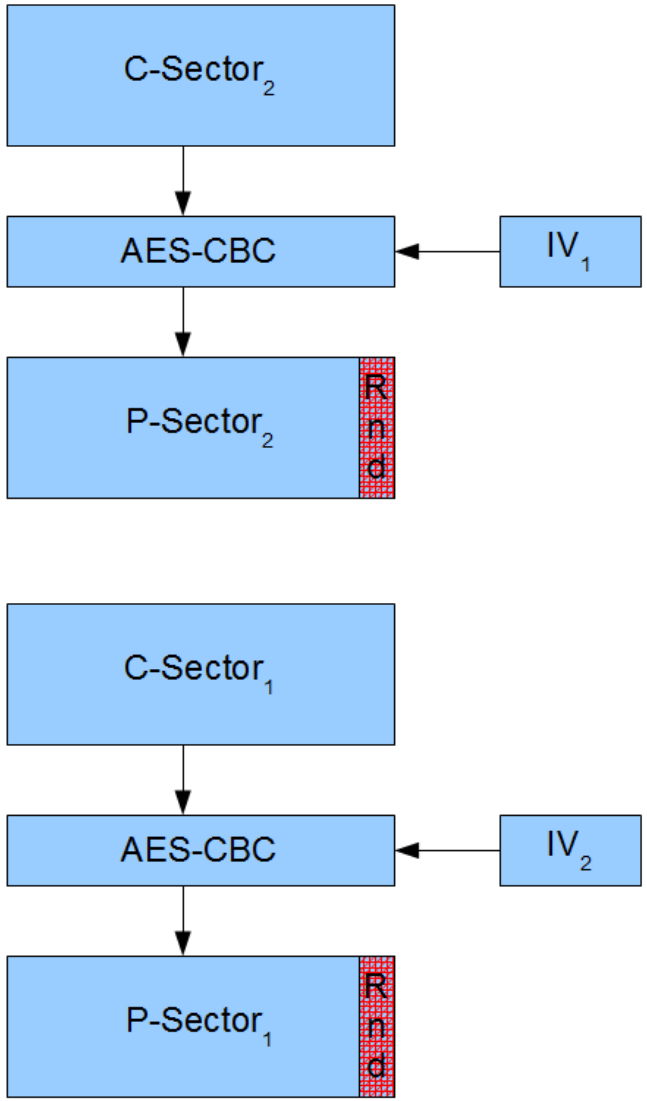
Sector move attack

Full sector viewpoint

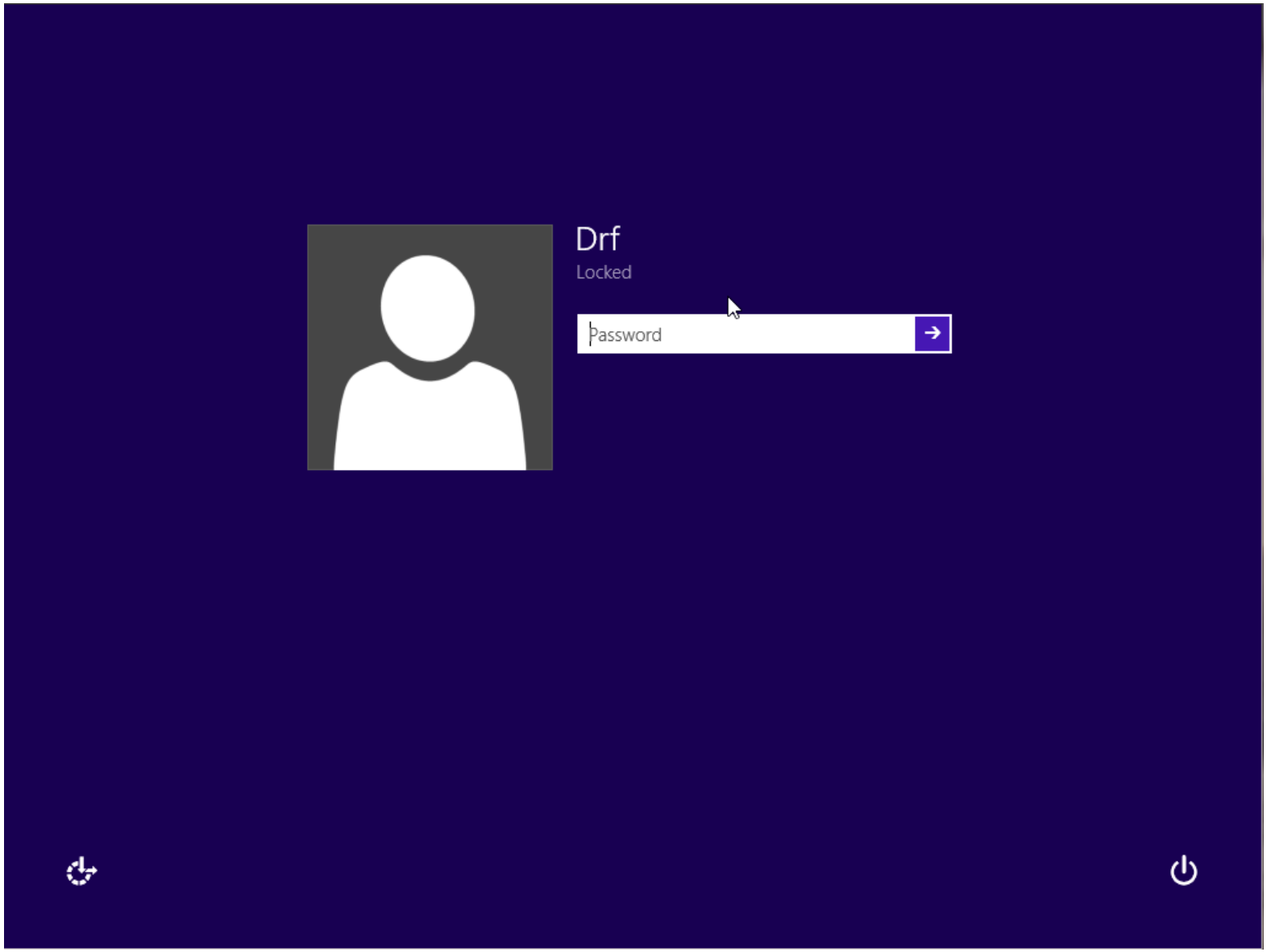


Sector move attack

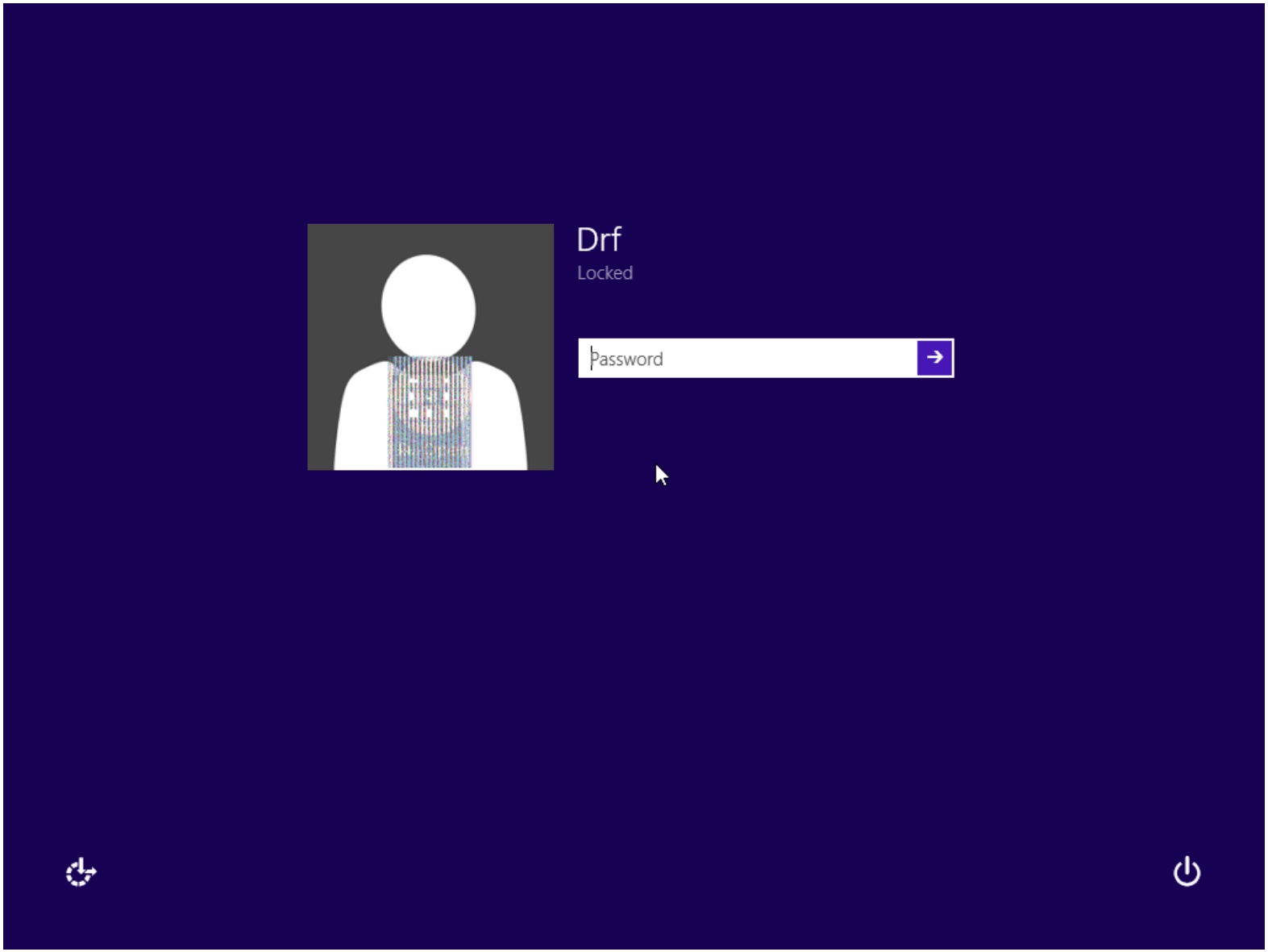
Moving a whole sector



Proof of concept



Proof of concept



Important assumptions

It is important to realise that in order for these attacks to be succesful some prerequisite assumptions must be met. We need in particular

1. Knowledge of at least a partial layout of the disc (where the files we wish to attack are)
2. For the targeted change we need the plaintext
3. Stay undetected by the user
4. Data changed can't be overly redundant or must not have integrity checking implemented

Conclusions

The decision to remove Elephant Diffuser from BitLocker in Windows 8, has had an undeniably negative impact on the security of this FDE solution. On the other hand Microsoft never claimed that BitLocker should protect data from a targeted attack, rather its use should be to protect data from an opportunistic attack. With a properly configured BitLocker installation this protection is still intact.

Thank you for your attention!
The End