Analysis
of
Window Vista
Bitlocker Drive Encryption

This is an **INCOMPLETE draft version.**
Visit **www.nvlabs.in** for updates

Nitin Kumar
Independent Security Researcher

Vipin Kumar
Independent Security Researcher

What we do ?

Analyzing malware
Custom Development of S/W
Code Reviewing
Network PenTests
and anything that seems interesting !

Presentation Outline

- Bitlocker Introduction
- Modes of Operation
- Available algorithms
- Structure of Bitlocker Volume
- Different Keys used in Bitlocker
- Key Generation
- Key Storage
- Key Usage
- Data Encryption
- In non-diffuser mode
- In diffuser mode
- References
- Questionaire

Bitlocker introduction

BitLocker Drive Encryption is a full disk encryption feature included with Microsoft's Windows Vista and Windows Server 2008 operating systems designed to protect data by providing encryption for entire volumes.

However, BitLocker is only available in the Enterprise and Ultimate editions of Windows Vista.

Modes of Operation

Bitlocker operates in one or more modes for every volume. Available modes are:-

Basic

- TPM only :- all keys are stored within TPM

Advanced

- USB:- Key is stored on an external device
- TPM + PIN:- TPM stores key with a user specific PIN
- TPM + USB:- TPM stores ½ key and USB stores another ½ half.
- TPM + USB + PIN (available in Vista SP1):- TPM stores ½ key, USB stores another ½ half, together with a user specific PIN.

Available Algorithms

User can select encryption algorithm at the time of enabling bitlocker.
Algorithm can be selected per volume.
And it cannot be changed during reseal.
To change algorithm,turn off bitlocker & then turn it on.

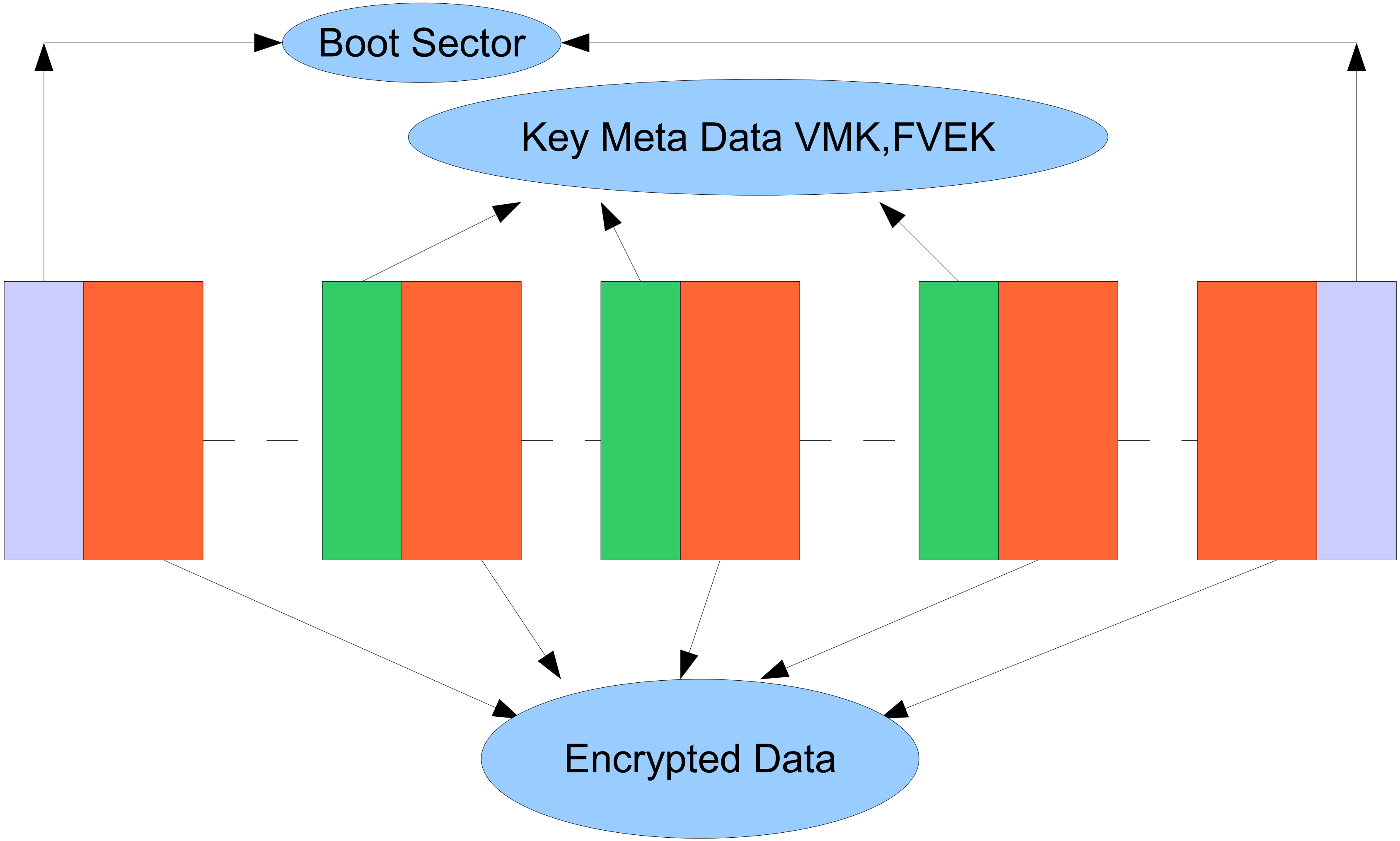
Available algorithms are

- **AES 128 bit**
- **AES 256 bit**
- **AES 128 bit + Diffuser (Elephant)   Default**
- **AES 256 bit + Diffuser (Elephant)**

Bitlocker Volume Structure

Structure of Bitlocker Volume

Bitlocker volume has almost all it sectors encrypted except a few which contain metadata.



Different Keys used in Bitlocker

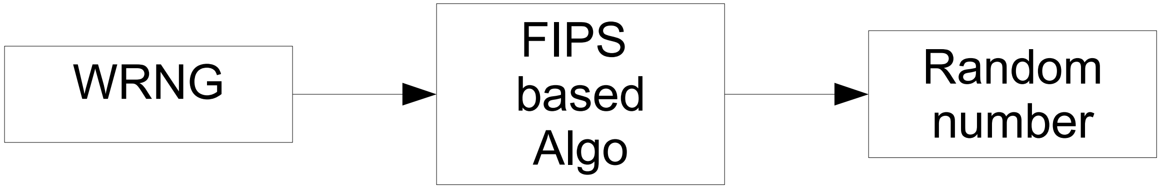
Bitlocker uses a total of 5 different types of keys which are as follows:-

- VMK unlockers(These keys decrypt VMK)
- VMK (Volume Master Key is used to decrypt FVEK)
- FVEK (Full Volume Encryption Key decrypts DATA)
- TWEAK Key (Generates Sector Key)
- SECTOR Key (decrypts DATA)

Each of these will be detailed in the subsequent slides

Key Generation

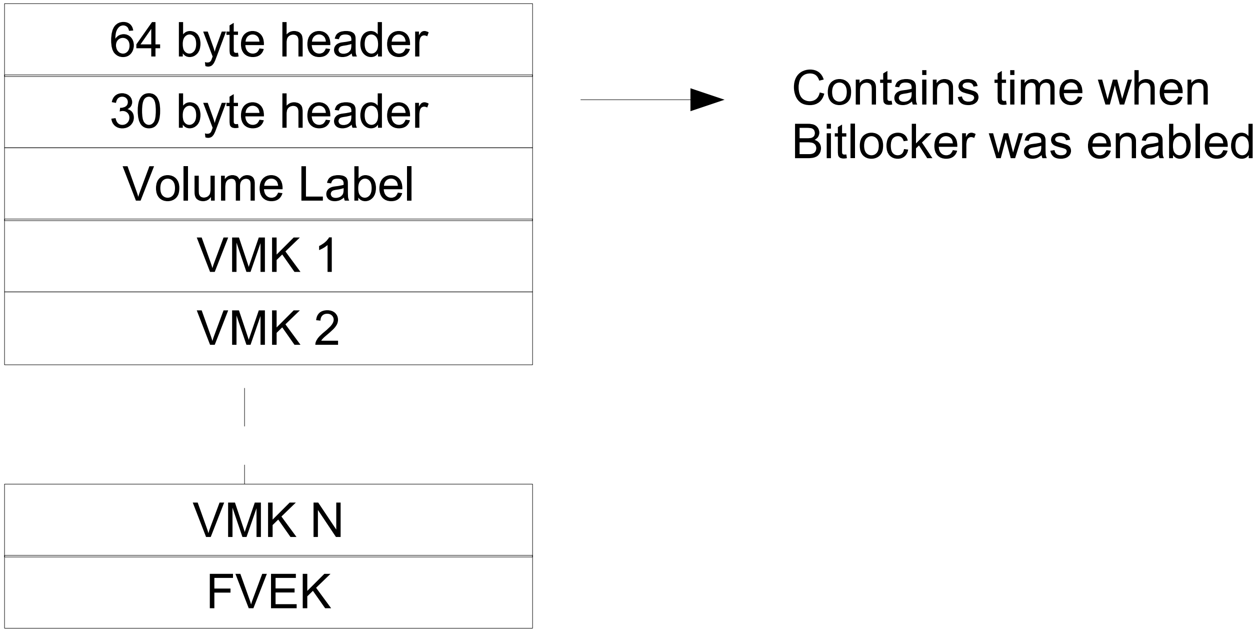
Whole encryption chain depends on keys, so keys should be derived in as random as possible method.



The above method is employed to generate all keys except Sector Key

Key Storage

The keys are stored in the meta data of the Bitlocker Volume. Total number of meta data blocks is 3.



64 byte header

30 byte header

Volume Label

VMK 1

VMK 2

VMK N

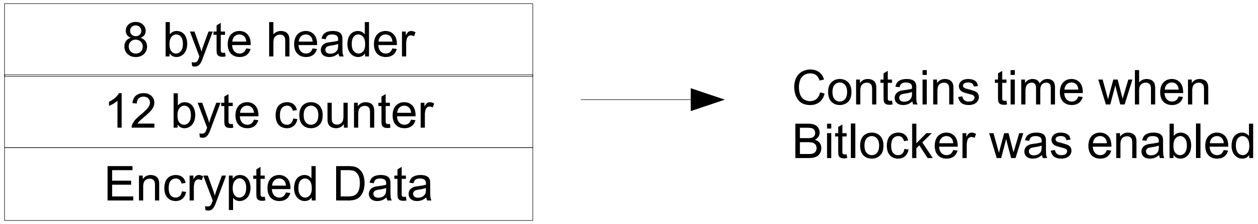
FVEK

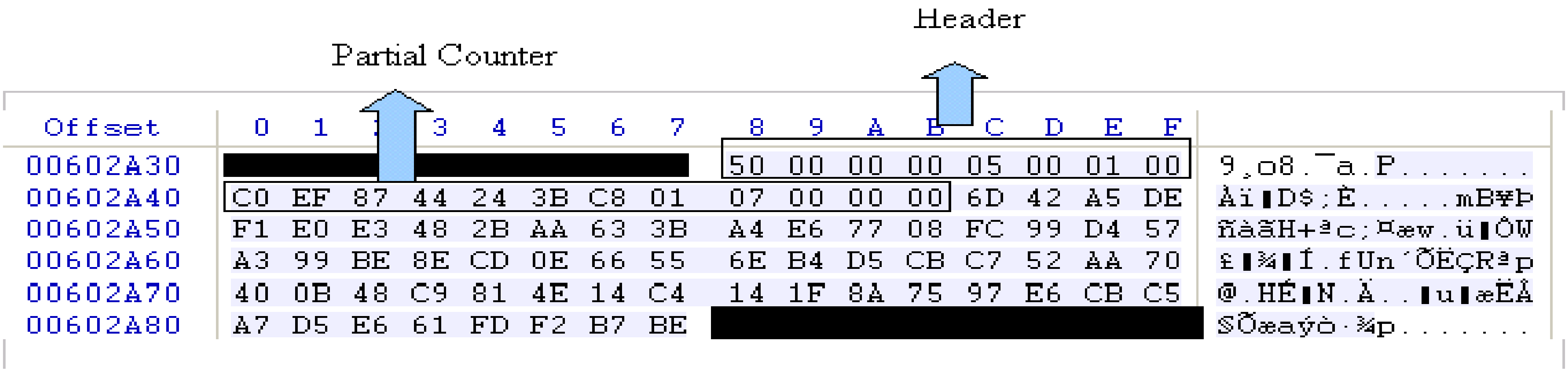
Contains time when Bitlocker was enabled

Key storage meta data structure as stored in Bitlocker volume

Encrypted Key Storage

The header contains size of encrypted data





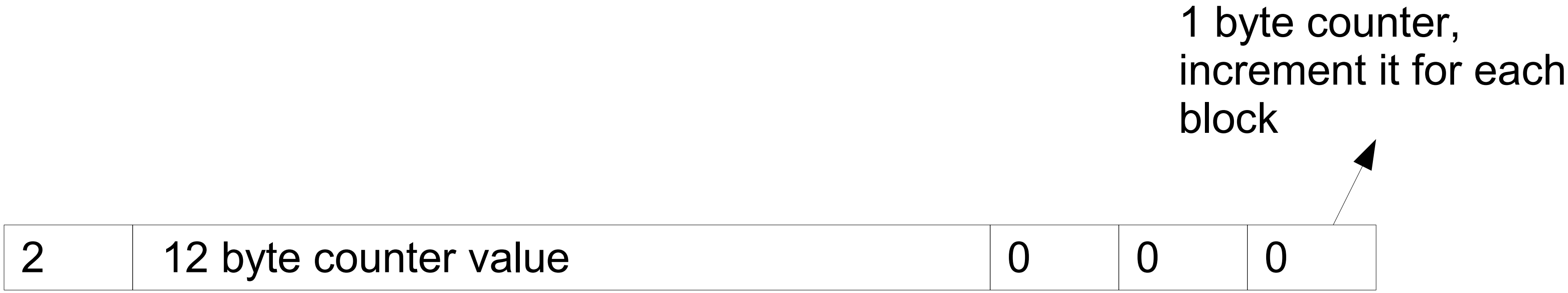
```
Offset     0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F

00602A30   [redacted]               50 00 00 00 05 00 01 00   9¸o8.¯a.P.......
00602A40   C0 EF 87 44 24 3B C8 01  07 00 00 00 6D 42 A5 DE   Àï▮D$;È.....mB¥Þ
00602A50   F1 E0 E3 48 2B AA 63 3B  A4 E6 77 08 FC 99 D4 57   ñàãH+ªc;¤æw.ü▮ÔW
00602A60   A3 99 BE 8E CD 0E 66 55  6E B4 D5 CB C7 52 AA 70   £▮¾▮Í.fUn´ÕËÇRªp
00602A70   40 0B 48 C9 81 4E 14 C4  14 1F 8A 75 97 E6 CB C5   @.HÉ▮N.Ä..▮u▮æËÅ
00602A80   A7 D5 E6 61 FD F2 B7 BE  [redacted]                §Õæaýò·¾p.......
```

Sample Encrypted Key

Key Encryption

The keys are encrypted either using RSA 2048 bit key or AES 256 bit.AES mode used is AES-CCM (AES–Counter with CBC-MAC)

In AES, 12 byte Counter is expanded as given below to 16 bytes



1 byte counter, increment it for each block

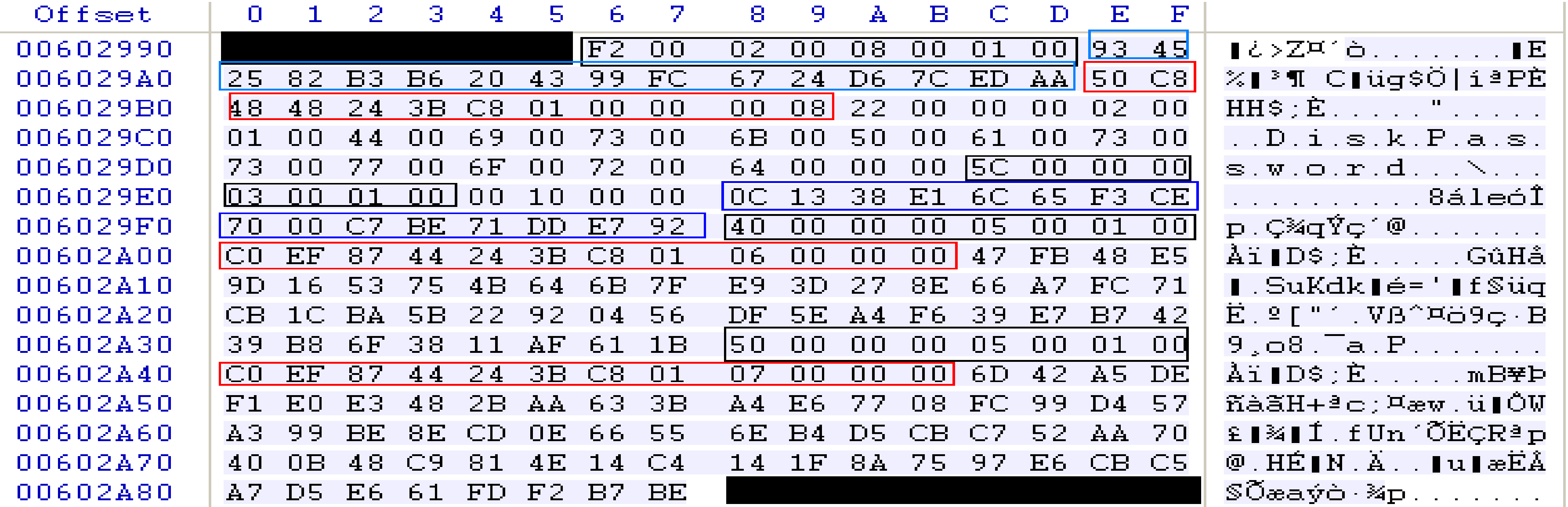
2	12 byte counter value	0	0	0

Expansion of Partial Counter to 16 byte Initialization Vector

Storage of VMK

N number of VMKs can be stored. Each one having a similiar structure.

8 byte header	Key type Label	Key encrypted using itself	VMK encrypted using key



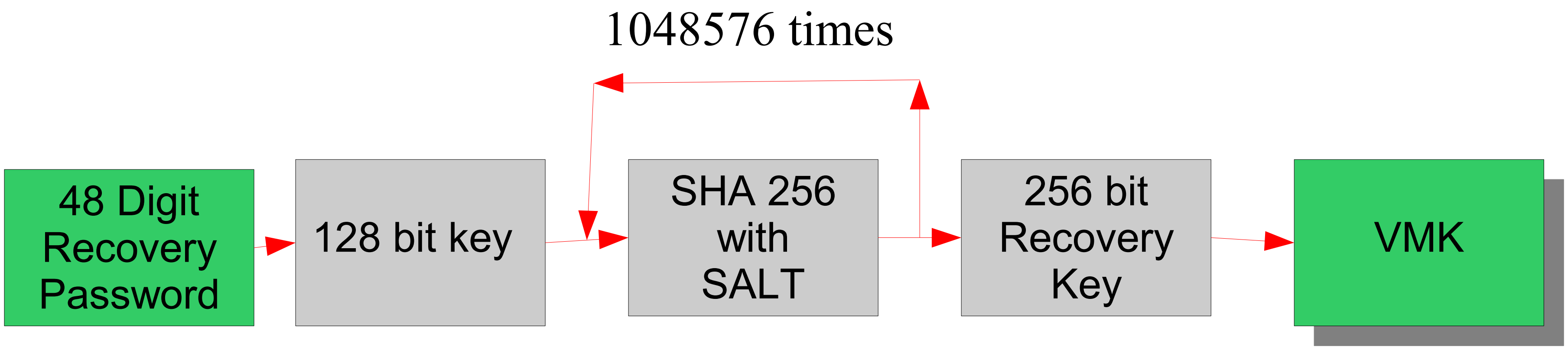
```
Offset     0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
00602990   [redacted]        F2 00  02 00 08 00 01 00 93 45   ▮¿>Z¤´ò........▮E
006029A0   25 82 B3 B6 20 43 99 FC  67 24 D6 7C ED AA 50 C8   %▮³¶ C▮üg$Ö|íªPÈ
006029B0   48 48 24 3B C8 01 00 00  00 08 22 00 00 00 02 00   HH$;È.....".....
006029C0   01 00 44 00 69 00 73 00  6B 00 50 00 61 00 73 00   ..D.i.s.k.P.a.s.
006029D0   73 00 77 00 6F 00 72 00  64 00 00 00 5C 00 00 00   s.w.o.r.d...\...
006029E0   03 00 01 00 00 10 00 00  0C 13 38 E1 6C 65 F3 CE   ..........8áleóÎ
006029F0   70 00 C7 BE 71 DD E7 92  40 00 00 00 05 00 01 00   p.Ç¾qÝç´@.......
00602A00   C0 EF 87 44 24 3B C8 01  06 00 00 00 47 FB 48 E5   Àï▮D$;È.....GûHå
00602A10   9D 16 53 75 4B 64 6B 7F  E9 3D 27 8E 66 A7 FC 71   ▮.SuKdk▮é='▮f§üq
00602A20   CB 1C BA 5B 22 92 04 56  DF 5E A4 F6 39 E7 B7 42   Ë.º["´.Vß^¤ö9ç·B
00602A30   39 B8 6F 38 11 AF 61 1B  50 00 00 00 05 00 01 00   9¸o8.¯a.P.......
00602A40   C0 EF 87 44 24 3B C8 01  07 00 00 00 6D 42 A5 DE   Àï▮D$;È.....mB¥Þ
00602A50   F1 E0 E3 48 2B AA 63 3B  A4 E6 77 08 FC 99 D4 57   ñàãH+ªc;¤æw.ü▮ÔW
00602A60   A3 99 BE 8E CD 0E 66 55  6E B4 D5 CB C7 52 AA 70   £▮¾▮Í.fUn´ÕËÇRªp
00602A70   40 0B 48 C9 81 4E 14 C4  14 1F 8A 75 97 E6 CB C5   @.HÉ▮N.Ä..▮u▮æËÅ
00602A80   A7 D5 E6 61 FD F2 B7 BE  [redacted]                §Õæaýò·¾p.......
```

Generating Recovery Key from Recovery Password

In case of system modification, user is asked to type a 48 digit key which will unlock the volume. Pseudocode given below

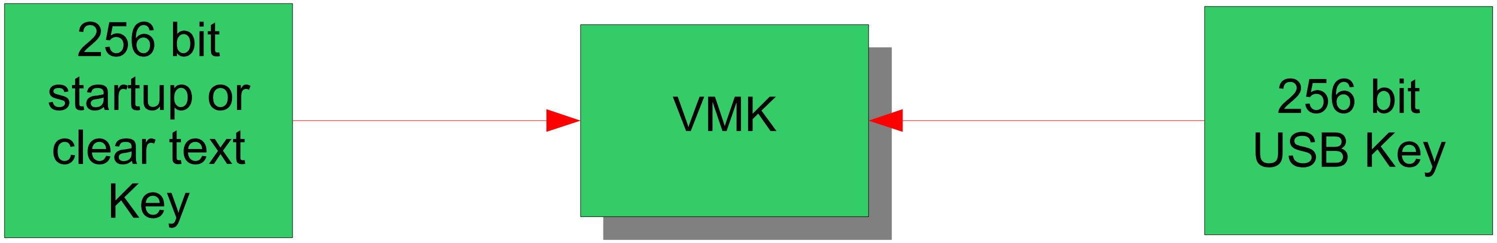
```
1.Divide each block by 11, if the remainder not 0 in all cases the key is not valid
2.collect the quotients, and concatenate them to obtain a 128 bit key.
3.Take a 88 byte buffer  and zero it. The structure of the buffer is as follows
 struct { unsigned char sha_current[32];
unsigned char sha_password[32];
unsigned char salt[16];
int64 hash_count;    };
4.Take SHA256 of the key and place it in the above structure in sha_password
5.The salt is place in the salt field of the above structure
6.Now run a loop  0x100000  ( 1048576)  times
7.Find SHA256 of the entire structure and place it in sha_current field
8.increment hash_count field counter in the structure
9.repeat steps 6 through 9 , till the loop is over
10. Take the first 32 bytes of the structure as the 256 bit key which can be used
to decrypt the VMK corresponding to this key
```

Generating Recovery Key from Recovery Password



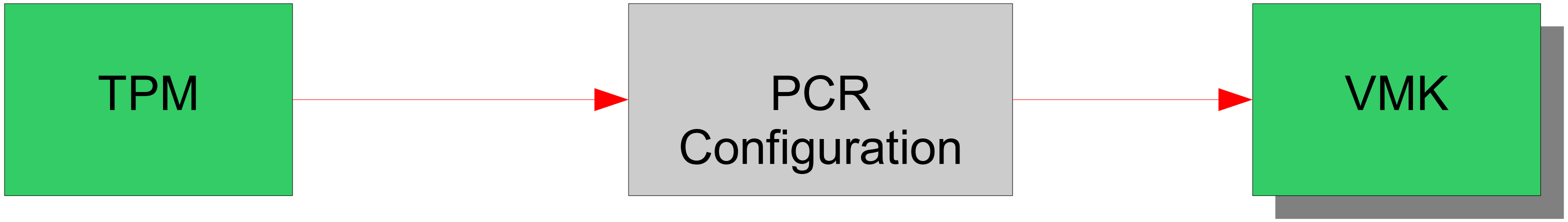
Block Diagram showing conversion from Recovery Password to Recovery Key

Startup Key and/or USB Key



Block Diagram showing usage of Startup Key and USB Key

TPM



Block Diagram showing usage of Startup Key and USB Key

Full volume Encryption Key (FVEK)

FVEK

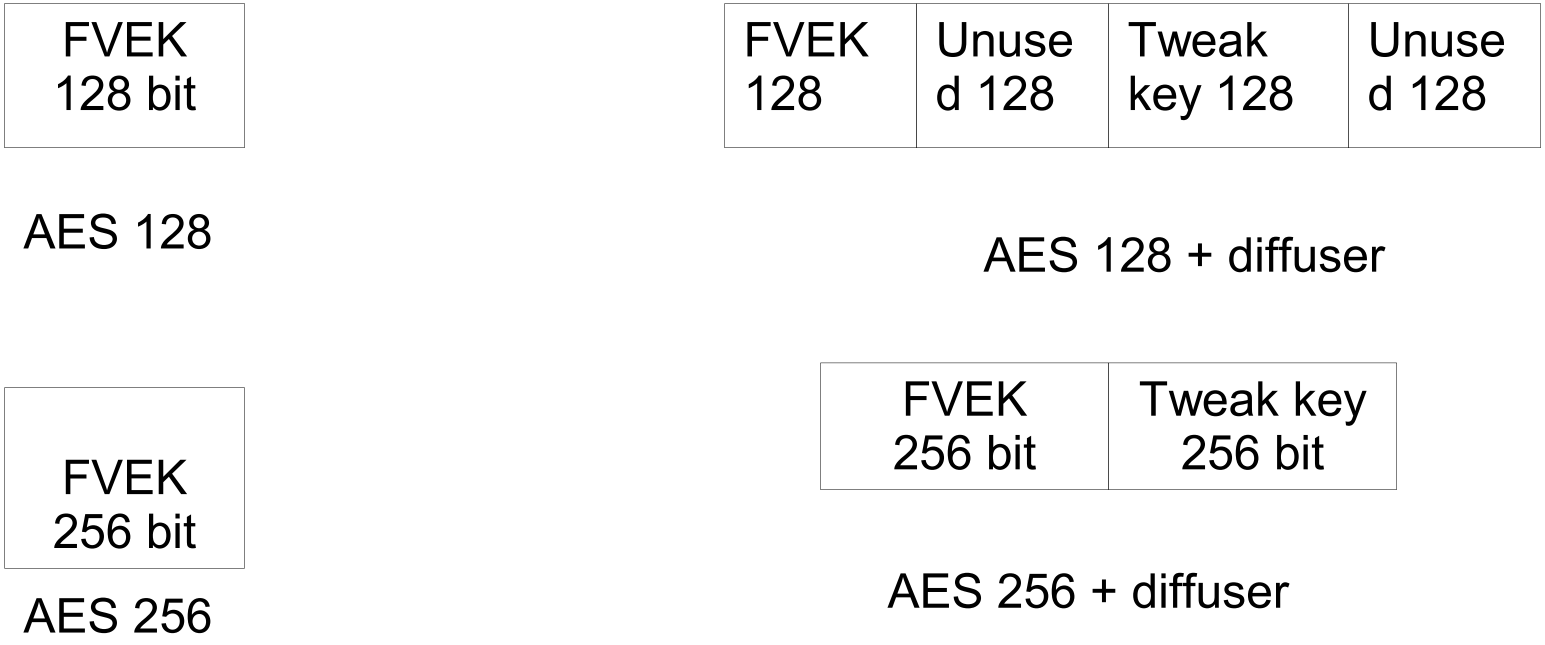
FVEK is used to data stored ion the volume.

It's size is different according to

AES 128 bit	size 128 bits
AES 256 bit	size 256 bits
AES 128 + diffuser	size 512 bits (half of the bits are unused)
AES 256 + diffuser	size 512 bits

FVEK Structure

FVEK is broken into two parts if larger that 256 bits



FVEK 128 bit

AES 128

FVEK 128	Unused 128	Tweak key 128	Unused 128

AES 128 + diffuser

FVEK 256 bit

AES 256

FVEK 256 bit	Tweak key 256 bit

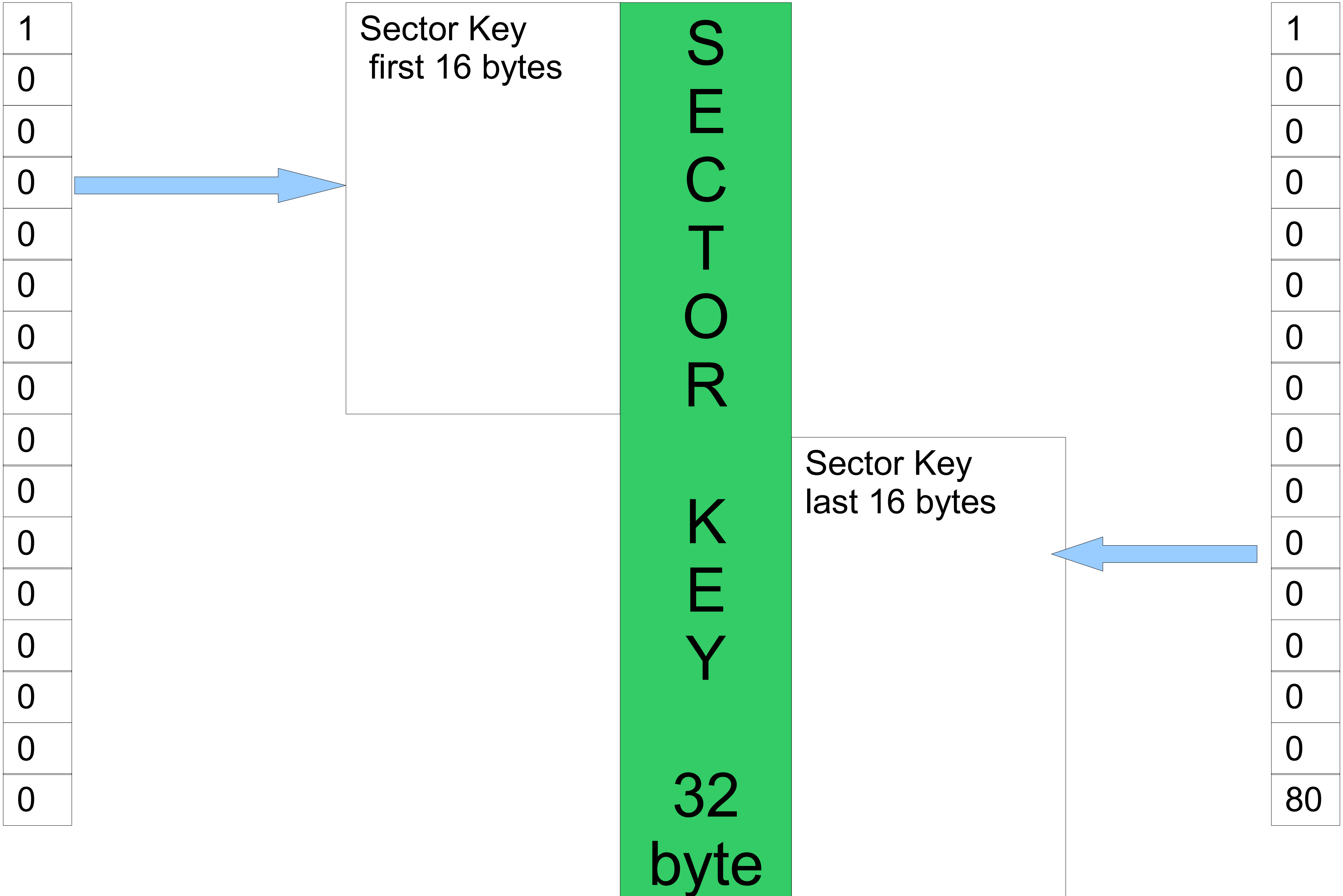
AES 256 + diffuser

Sector key from TWEAK key

Pseudocode

- Take a buffer of 16 bytes, zero it.
- Now copy the Sector Number in little endian format and encrypt it with TWEAK key to obtain first 16 bytes of Sector key.
- Take a buffer of 16 bytes, zero it.
- Now copy the Sector Number in little endian format and make the 16th byte as 128 or 0x80,now encrypt it with TWEAK key to obtain remaining 16 bytes of Sector Key.
- Concatenate both part to obtain full 32 byte or 512 bit Sector Key

Sector key from TWEAK key



1
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0

Sector Key
first 16 bytes

SECTOR KEY
32 byte

Sector Key
last 16 bytes

1
0
0
0
0
0
0
0
0
0
0
0
0
0
0
80

Diffusers A & B

The Diffusers just diffuse the data ie they mingle up the bits
Bitlocker has 2 diffusers called Diffuser A and Diffuser B

Diffuser doesn't need any keys and thus doesn't need to be broken to defeat bitlocker.

It's just based on XOR and mod operation

Diffuser B

Diffuser B in decryption direction
It's represented by

```
for i = 0, 1, 2, ,; n
d[i] = d[i] + (d[i+2]  XOR( d[i+5] <<< Rb[n mod 4])
```

where Rb = [0 ,10 ,0,25]

To obtain encryption function, just change first + to -

NOTE:- data is processed in 32 bit blocks
            <<< is left rotate operation

Diffuser A

Diffuser A in decryption direction

It's represented by

```
for i = 0, 1, 2, ,; n
d[i] = d[i] + (d[i-2]  XOR( d[i-5] <<< Ra[n mod 4])
```

where Ra = [9,0 ,13 ,0]

To obtain encryption function, just change first + to -

NOTE:- data is processed in 32 bit blocks

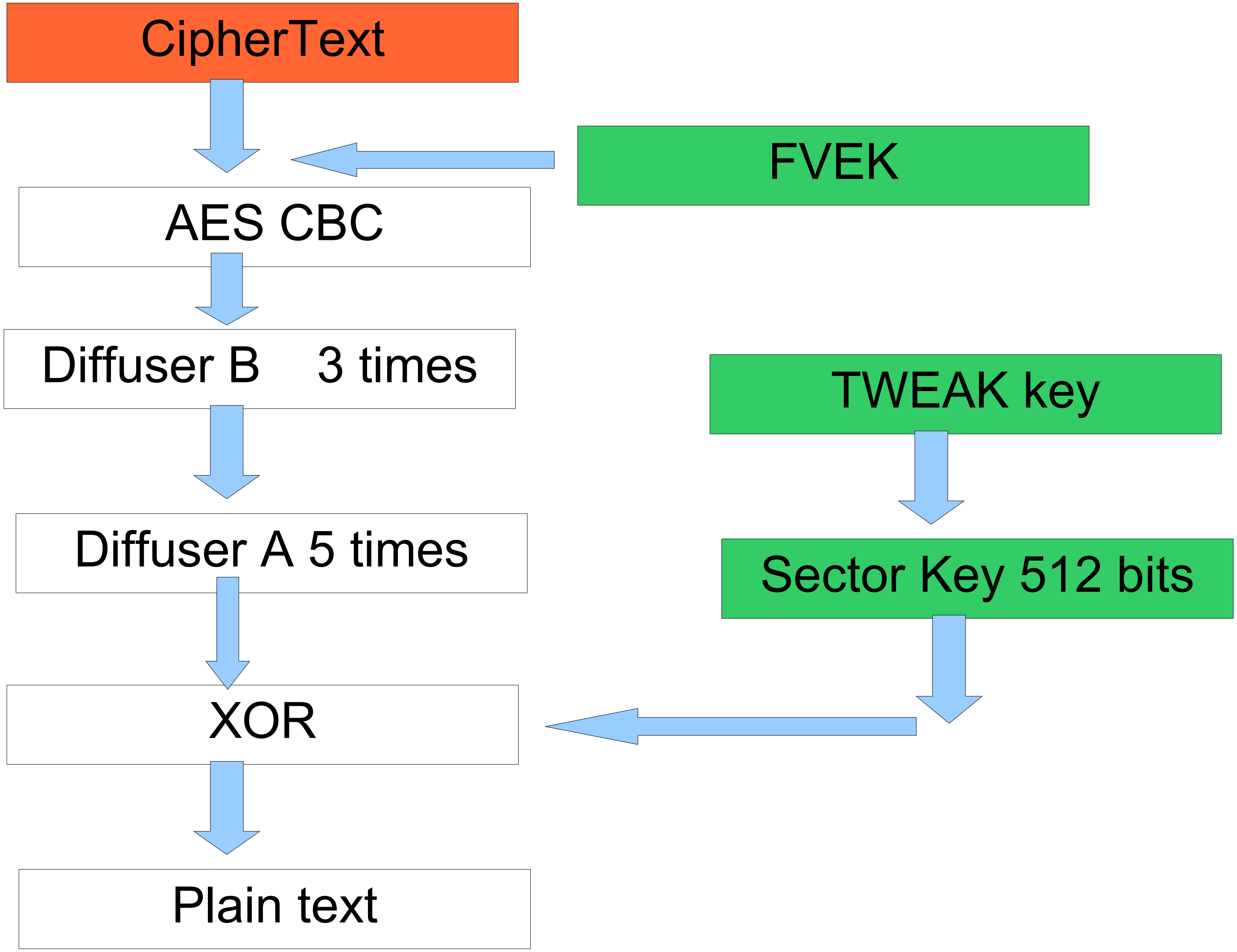
<<< is left rotate operation

Data Encryption

In AES 128 bit mode and AES 256 bit mode, AES-CBC mode is used with initialization vector (16 zero bytes)

However, if a diffuser capable mode is selected, then things turn out to be little bit more complex

Data decryption in diffuser capable mode



CipherText

FVEK

AES CBC

Diffuser B 3 times

TWEAK key

Diffuser A 5 times

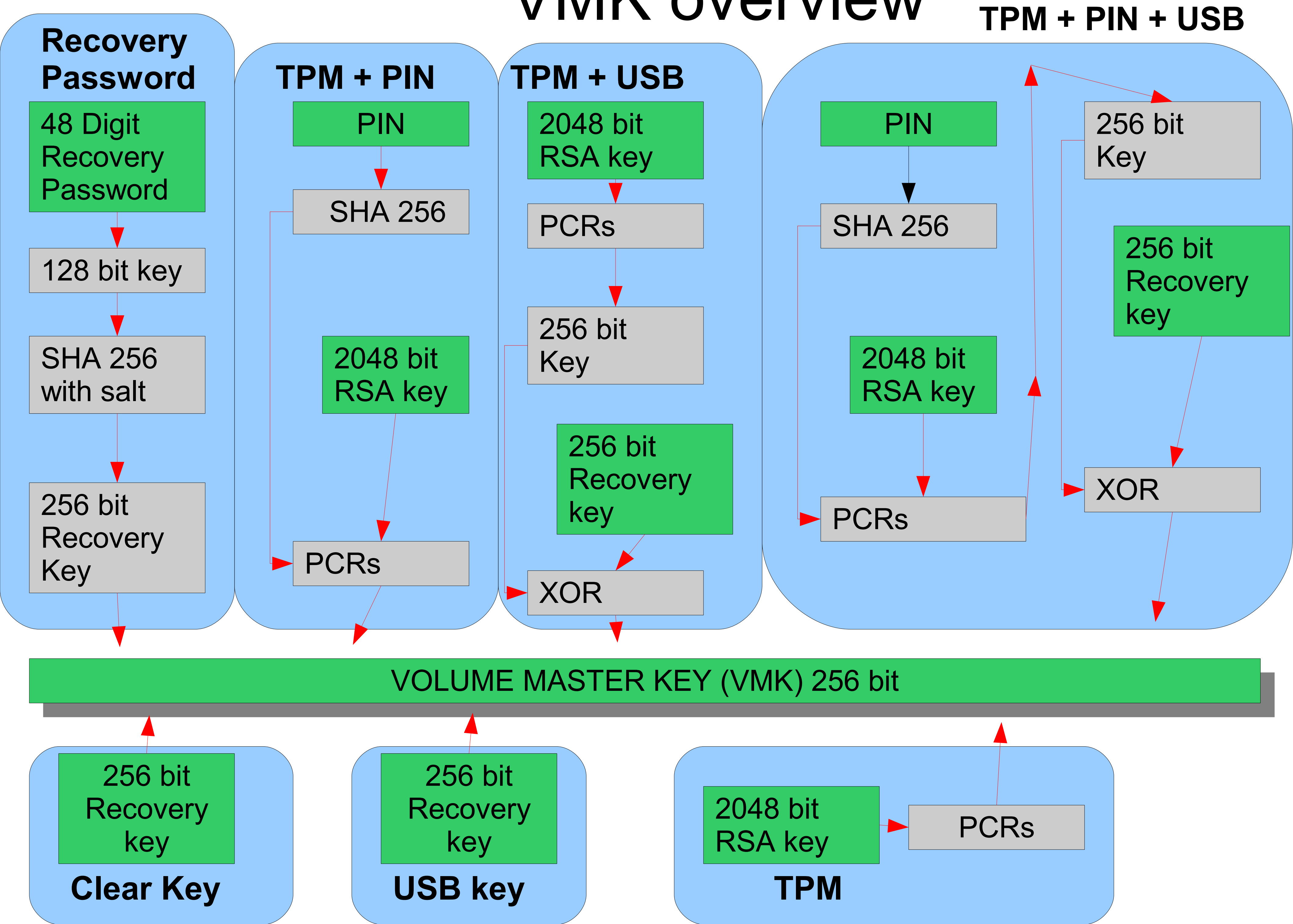
Sector Key 512 bits

XOR

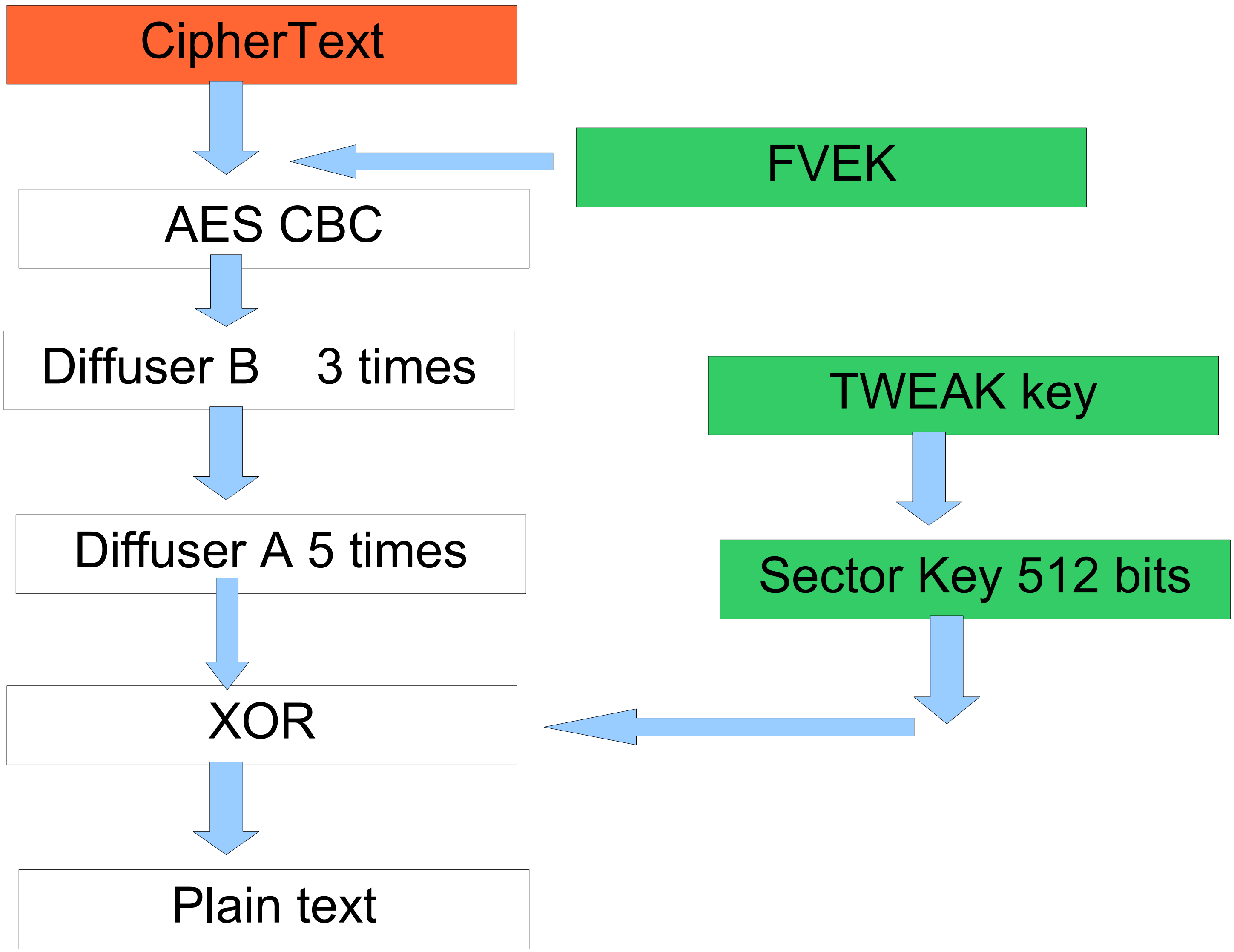
Plain text

Quick Rewind

VMK overview



Data decryption in diffuser capable mode



Tool Release

Tool features

- Transparent access to bitlocker volumes (if user supplies appropriate keys)
- 2 modes are supported(using Recovery Password/USB startup key)
- Currently provides only read only access but write access can be added
- Ability to process partition image files
- Ability to convert Bitlocker Volume to NTFS volumes permanently.

References

- Brown, Ralf. Ralf Brown’s Interrupt List. http://www.cs.cmu.edu/~ralf/files.html
- Nitin Kumar,Vipin Kumar Vbootkit:Compromising Windows Vista Security
- Randall Hyde ,Art of assembly Language
- M. Conover (2006, March). “Analysis of the Windows Vista Security Model,” http://www.symantec.com/avcenter/reference/Windows_Vista_Security_Model_Analysis.pdf

Questionaire ?

Questions

Comments

email us at

nitin@nvlabs.in
vipin@nvlabs.in

http://www.nvlabs.in

Thank you