BitLocker To Go

Jesse Kornblum
Outline

- Introduction
- While you'll see more BitLocker
- What hasn't changed
- Passwords
- Smart Cards
- Drive Layout
- BitLocker To Go
- Auto Unlock Keys
- Incident Response
- Forensics
- Operations
Introduction

• BitLocker Disk Encryption (BDE)
  – Full Volume Encryption
  – Introduced with Windows Vista
  – Protects data at rest
Introduction

• Not widely adopted
  – Only came with Enterprise or Ultimate edition
  – Pain to install
    • Required hard drive repartition
    • Easy to lose data
  – Pain to use
    • Easy to lose data
BitLocker version 2

- Introduced in Windows 7 (Fall 2009)
  - Still only in higher end versions
- Asks to upgrade existing BDE volumes
  - Still painful to use for fixed disks
Why You'll See More BitLocker

• Can encrypt removable media
  – BitLocker To Go (BTG)
  – USB drives
  – Encrypts in place
• As easy as right clicking
Why You'll See More BitLocker

User right-clicks drive icon
Why You'll See More BitLocker

• BitLocker policy options
  – Can be configured to only allow writes to BTG protected devices
• Force protection
• Ideal for deployed environments
  – USB sticks are lost, stolen
  – Prevent thief from accessing data
What Hasn't Changed

• Computationally infeasible to access protected volume without a key
  • There is no back door
  • Best hope is known plaintext attack against AES-256 (good luck!)

Image courtesy Flickr user shoothead and licensed under the Creative Commons
What Hasn't Changed

- Data is encrypted with Full Volume Master Key (FVEK)
- FVEK is stored on disk, encrypted with Volume Master Key (VMK)
- Many copies of VMK are stored on disk, encrypted with different keys
  - TPM, Recovery, External, etc.
What Hasn't Changed

• Logical access to a drive is not affected
  – Programs do not need to know about BitLocker to access the drive

• Physical access to volume
  – Use `\\C:` file
  – This file handle can be imaged

• Still using AES with Elephant diffuser

• BitLocker can be disabled but still encrypt the data
What Hasn't Changed

• For each particular key $k$
  – Metadata contains copy of the VMK encrypted with that key
  – $E(VMK, k)$
• Metadata also contains copy of $k$ encrypted with VMK
  – $E(k, VMK)$

• Having one key means you can get all the keys
  – Allows user to recreate an access device
  – Allows us to create an access device, too
What Hasn't Changed

• Let's suppose metadata contains:
  – $E(\text{VMK, recovery password})$
  – $E(\text{recovery password, VMK})$
  – $E(\text{VMK, external key})$
  – $E(\text{external key, VMK})$
  – $E(\text{VMK, TPM key})$
  – $E(\text{TPM key, VMK})$
Passwords

• Volumes can be protected with a Unicode password
  – Documentation used "password" and "passphrase" interchangeably
• Policy can control complexity requirements
  – By default, must be at least eight characters
• To get key from password:
  – Hash password using SHA-256
  – Chain hash the result, with salt, $2^{16}$ times
    • Repetitions makes brute force attacks more difficult
    • Salting impacts rainbow tables
Smart Cards

- Volume is protected using Smart Card and PIN
  - Something you have and something you know
  - Exact method of key generation is unknown
Command Line Interface

- Passwords and Smart Cards not normally used for fixed disks
- But can be added using CLI
- manage-bde.exe
Drive Layout

• Version one munged NTFS header
  • From NTFS to -FVE-FS-
  • Also set offset 0x38 to point to metadata
• Version two completely replaces NTFS header
  • Still has -FVE-FS- signature, but is not a valid disk header
• Contains BitLocker GUID
  • 4967D63B-2E29-4AD8-8399-F6A339E3D001
• Appears on disk as
  • 3b d6 67 49 29 2e d8 4a 83 99 f6 a3 39 e3 d0 01
Version 1 Drive Layout

NTFS

FVE-VEFS

Metadata Metadata Metadata
Version 2 Drive Layout
### Drive Layout

- **Metadata format changed slightly**
- **Offset** | **Size** | **Description**
- 0x00 | 8 | Signature FVE-FS-
- 0x08 | 2 | Unknown
- 0x0A | 2 | Version, must be 2
- 0x0C | 2 | Unknown
- 0x0E | 2 | Unknown
- 0x10 | 8 | Volume size, in bytes
- 0x18 | 4 | Unknown
- 0x1C | 4 | Size of volume header, in sectors
- 0x20 | 8 | First Metadata offset
- 0x28 | 8 | Second Metadata offset
- 0x30 | 8 | Third Metadata offset
- 0x38 | 8 | Offset of volume header
- 0x40 | 4 | Metadata length
- 0x44 | 4 | Unknown
- 0x48 | 4 | Unknown
- 0x4C | 4 | Metadata length 2
- 0x50 | 16 | Volume GUID
- 0x60 | 4 | Next nonce
- 0x64 | 4 | Volume algorithm
- 0x68 | 8 | Timestamp
BitLocker To Go

- Right-click easy for USB devices
- Can also be enabled from Control Panel and CLI
BitLocker To Go

- Default protection is password or smart card
  - Can add others using CLI
  - Only one password per device, sorry
- Encrypts in place
  - Doesn't disturb underlying FAT12, FAT32, NTFS or exFAT
  - User can work during encryption
- If space available, adds FAT32 file system with 'reader' application
  - More like a 'copy' application, really
  - Takes about 5MB
  - Works on Windows XP and Vista
BitLocker To Go Drive Layout

EXFAT

FAT32  Metadata  Metadata  Metadata  EXFAT
BitLocker To Go

- When using GUI, user **must** create a recovery key file
  - Series of eight groups of six digits
- Saved to a file on the disk
  - Default name is GUID of the recovery key
  - Default save location is user's home directory
  - BUT! Key must be on a removable device to be used
  - Can also be typed manually
Policy Settings

- Stored in Registry
- Control complexity, writing to unprotected devices
- HKCU\Software\Policies\Microsoft\FVE
Auto Unlock Keys

- User can configure system to automatically unlock a volume
  - BitLocker External Key stored in registry
- HKCU\Software\Microsoft\Windows\CurrentVersion\FveAutoUnlock\{GUID\}
- Key and metadata encrypted using CryptProtectData function
  - Uses login credentials and 3DES
  - Can be decrypted on the same machine
Auto Unlock Keys
Auto Unlock Keys

• Unfortunately GUIDs in the registry correspond to auto unlock keys, not the USB device itself
  – Can't correlate to list of USB devices seen by a machine
Incident Response

- Need the Full Volume Encryption Key
- Kept in RAM when drive unlocked
  - Capture RAM
- Search memory image for AES keys
  - See "Cold Boot" attack
  - Really searching for not AES key schedules

Image courtesy Flickr user mape_s and used under the Creative Commons
Incident Response

- Need a **physical** image of the drive
- If the drive is mounted, can image decrypted drive!
Incident Response

- Want the auto unlock keys in the registry
- Grab contents of
  - HKCU\Software\Microsoft\Windows\CurrentVersion\FveAutoUnlock
  - Decrypt using user credentials
Incident Response

- Look for written down/printed out recovery key
  - Eight groups of six digits
  - Each group is a multiple of 11
- Look for Recovery key files
  - Unicode text
  - Contain string "Full recovery key identification:"
- Look for External key files
  - Should be stored on USB devices
  - .BEK extension
Forensics

- Look for clear key
  - Indicates BitLocker has been disabled
Forensics

• Password cracking
  • About 0.8 seconds per guess
  • Time spent in chain hashing
  • Makes brute force generally infeasible

• Key guessing
  • Easiest attack is against FVEK
  • Known plaintext attack against AES-256
  • $2^{128}$ possible keys, all equally likely
    • Good luck!
Forensics

- Try it out!
- There's a BTG protected volume in the 2010 DC3 Forensics Challenge

Image courtesy U.S. Department of Defense and is public domain
Interrogation

(this slide intentionally left blank)
Forensics

- Look for evidence of use on XP and Vista systems
  - Prefetch files for BitLockerToGo.exe
Operations

• Can add an access device
  • Remember, only one password per device
  • But external keys or recovery keys are fair game
• Can recreate an access device
  • Use stored copies of the VMK
• Can copy auto unlock keys from one computer to another
  • Have to be decrypted and re-encrypted
Outline

• Introduction
• While you'll see more BitLocker
• What hasn't changed
• Passwords
• Smart Cards
• Drive Layout
• BitLocker To Go
• Auto Unlock Keys
• Incident Response
• Forensics
• Operations
Questions?

Jesse Kornblum
jesse.kornblum
@mantech.com

Image courtesy Flickr user ella_marie and licensed under the Creative Commons