Breaking the keychain from digital forensic perspective

Chain Breaker - Keychain Analysis & Data Extractor

forensic.nofate.com

Who?

- I'm with Agency for Defense Development
- Co-Developer for volafox a.k.a Mac OS X Memory Forensic Toolkit
- Member of F-INSIGHT
- I'm not a Mac Nerd ;p

http://volafox.tumblr.com

http://forensicinsight.org

Contents

- Introduction
- Related Works
- Procedure for Analysis
- Show Time
- Conclusion

Introduction

Students and their notebooks at The Missouri School of Journalism

reference : http://osxdaily.com/wp-content/uploads/2010/08/macs-at-college.jpg

Introduction

- Status of Mac OS X Forensics
  - Early stage research ;-(
    - Information gathering using API
    - has some high-level forensic tool
      - sleuthkit, volafox ...
    - Most are not suitable for Mac OS X

Introduction

http://1.bp.blogspot.com/_pAFfga-55tg/S9ZHb8L1ZoI/AAAAAAAAADE/RIB9p2DPqtU/s1600/Man+on+Mountain.jpg

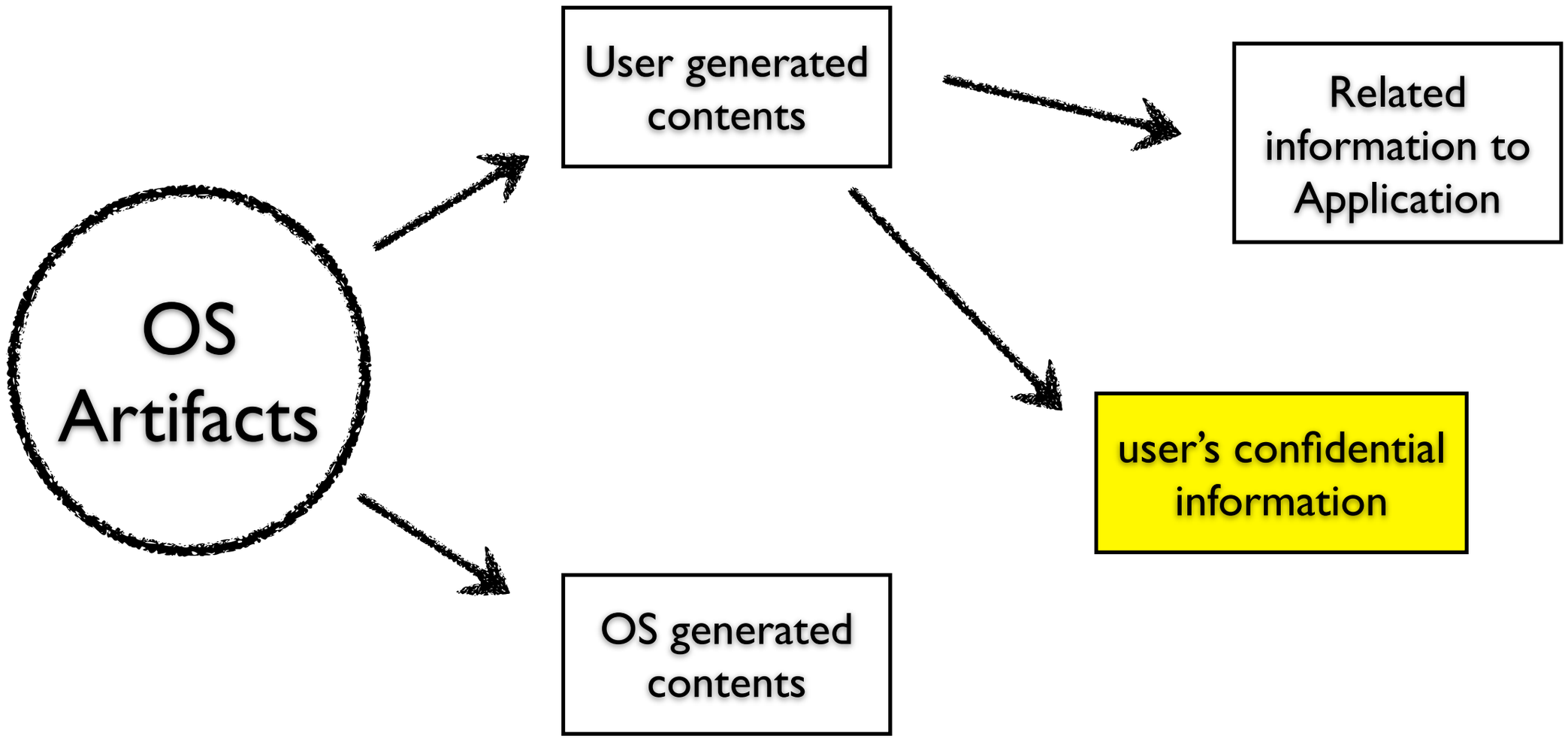
- We are going to the top of a mountain :)

Introduction

Memory Forensics	Disk Forensics	OS Artifacts
 VOLATILITY	EnCase® AccessData®	  MacResponse Forensics

OK?

Introduction



Introduction

- User Account is most important things
  - Mac OS X has Integrated password management system
  - It is named the ‘Keychain’

Keychain (Wikipedia)

- Password management system in Mac OS
- It contain various types of data
  - passwords(Websites, FTP, SSH, network shares, wireless networks, groupware applications, encrypted disk images)
  - and private keys, certificates, secure notes

Keychain (Wikipedia)

- Storage and access
  - ~/Library/Keychains
  - /Library/Keychains/
  - /Network/Library/Keychains/
- GUI Application : Keychain Access GUI
- CUI Application : /usr/bin/security

Reference : http://www.lawyerswithdepression.com/wp-content/uploads/2012/04/stress-management-technique.gif

Related Works

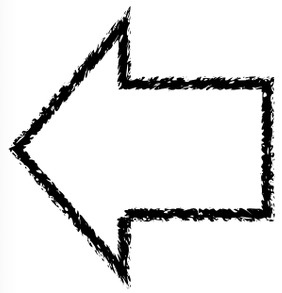
Related Works

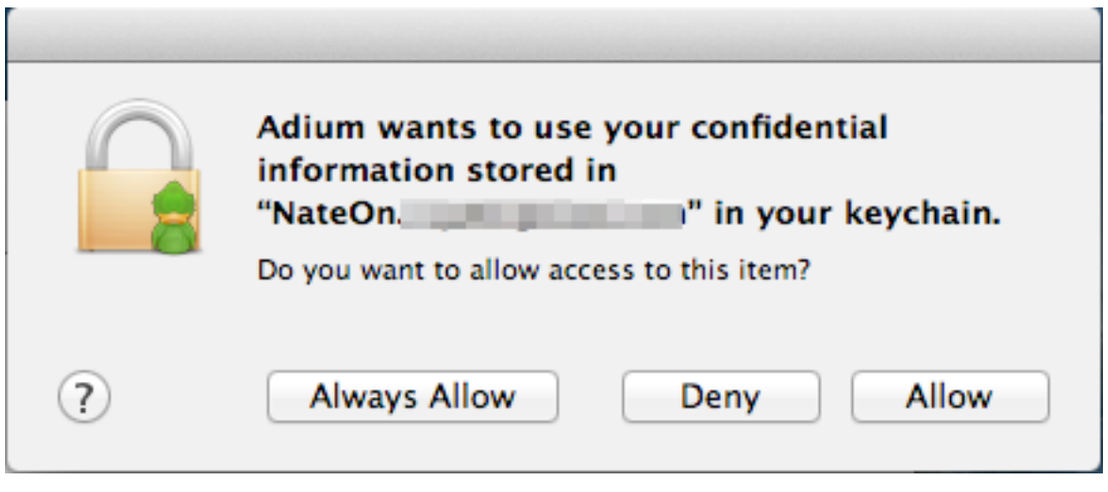
- 2004, Matt Johnston release tool named ‘extractkeychain’
- It is based on Security-177 Source Code
- extract data using ‘security dump-keychain’ & decrypt data through the master key generated by user password

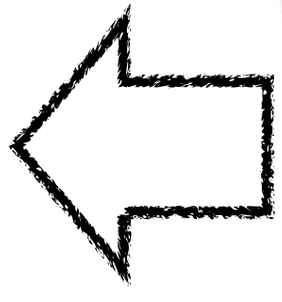
Related Works

- 2011 Juuso Salonen wrotes PoC Code named ‘keychaindump’
  - It executes a ‘vmmap’ command to find MALLOC_TINY area in securityd process and extract a master key candidates
  - Signature-based data recovery

Related Works







Limitations

- root privileges

- running it on live system

- Credential search based on signature

Limitations?

- How?

  ⇨ Memory & Disk Imaging!

- Integrity?

  ⇨ Memory dump (e.g. Inception)!

- Reliability?

  ⇨ keychain structure analysis!

Reference : http://www.richardgjonesjr.com/storage/research2books.jpg?__SQUARESPACE_CACHEVERSION=1274733532270

Procedure for Analysis

Method

- If we have memory and disk image,
  - Extracting the master key from memory image
  - keychain file format analysis (reversing & source code analysis)
    - finally, we decrypts user's confidential information. :-)

Getting the master key

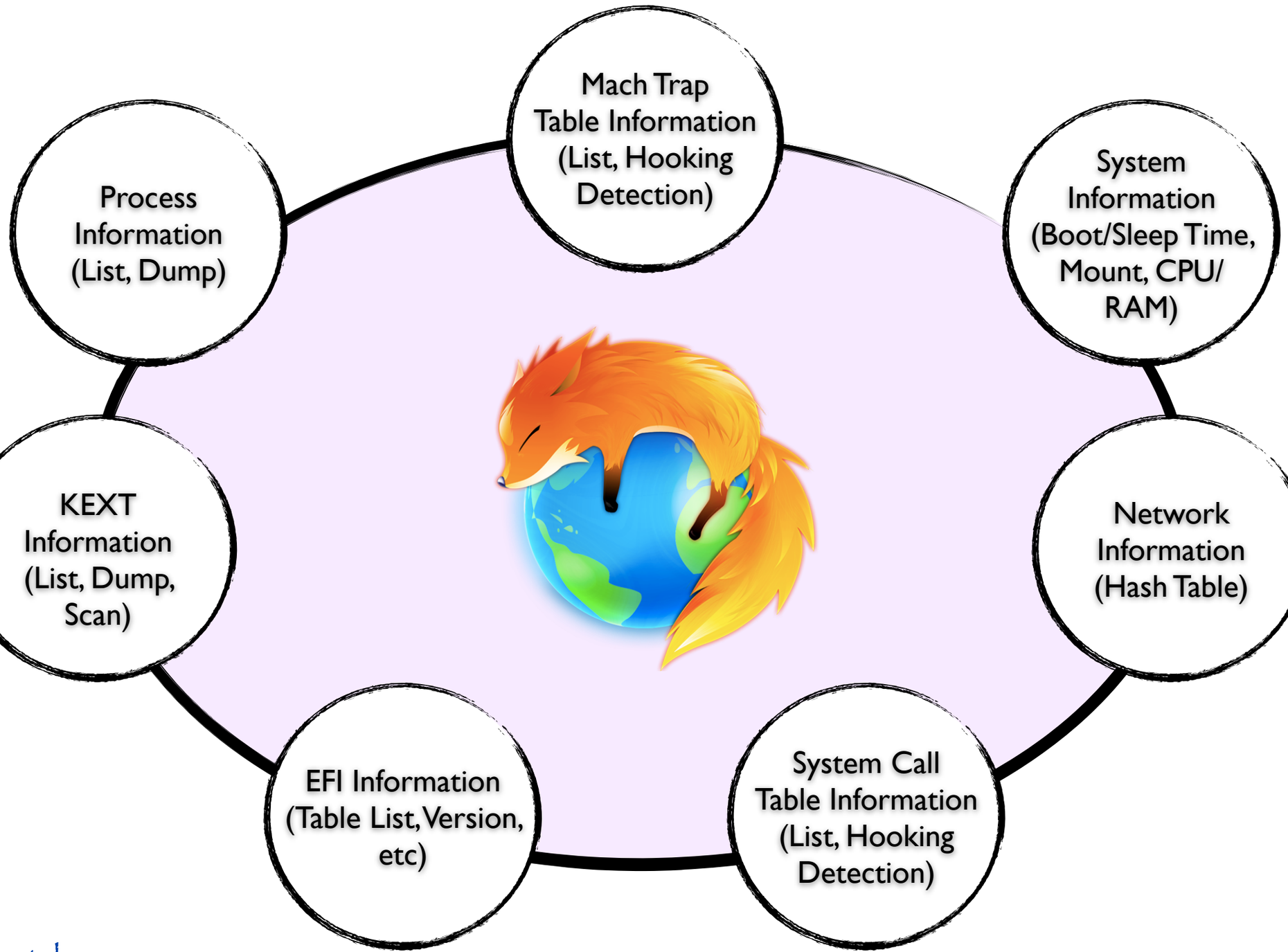
- We have to find virtual memory space of ‘Security Server’ process
- How to solve this problem?
  - I have selected the memory forensic tool and developed a keychaindump module.

volafox project

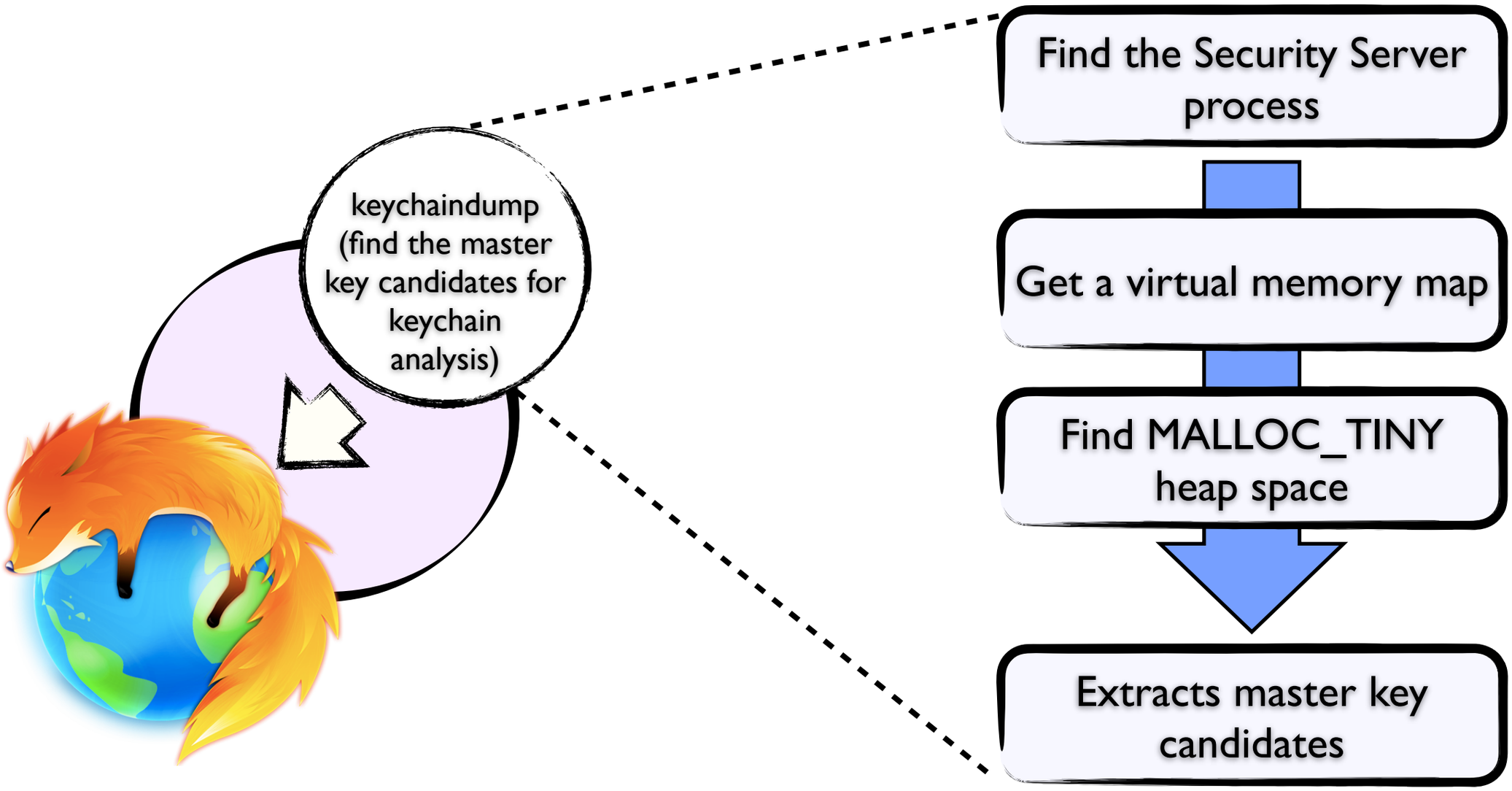
Icon Reference : http://www.kaishinlab.com

- volafox is most famous memory forensic toolkit for Mac OS X

volafox project



volafox project



Keychain file format

- File Format : Apple Database
- OS X security APIs are built on the open source Common Data Security Architecture(CDSA) and its programming interface, Common Security Services Manager(CSSM)
- Site: http://www.opengroup.org/security/

Keychain file format

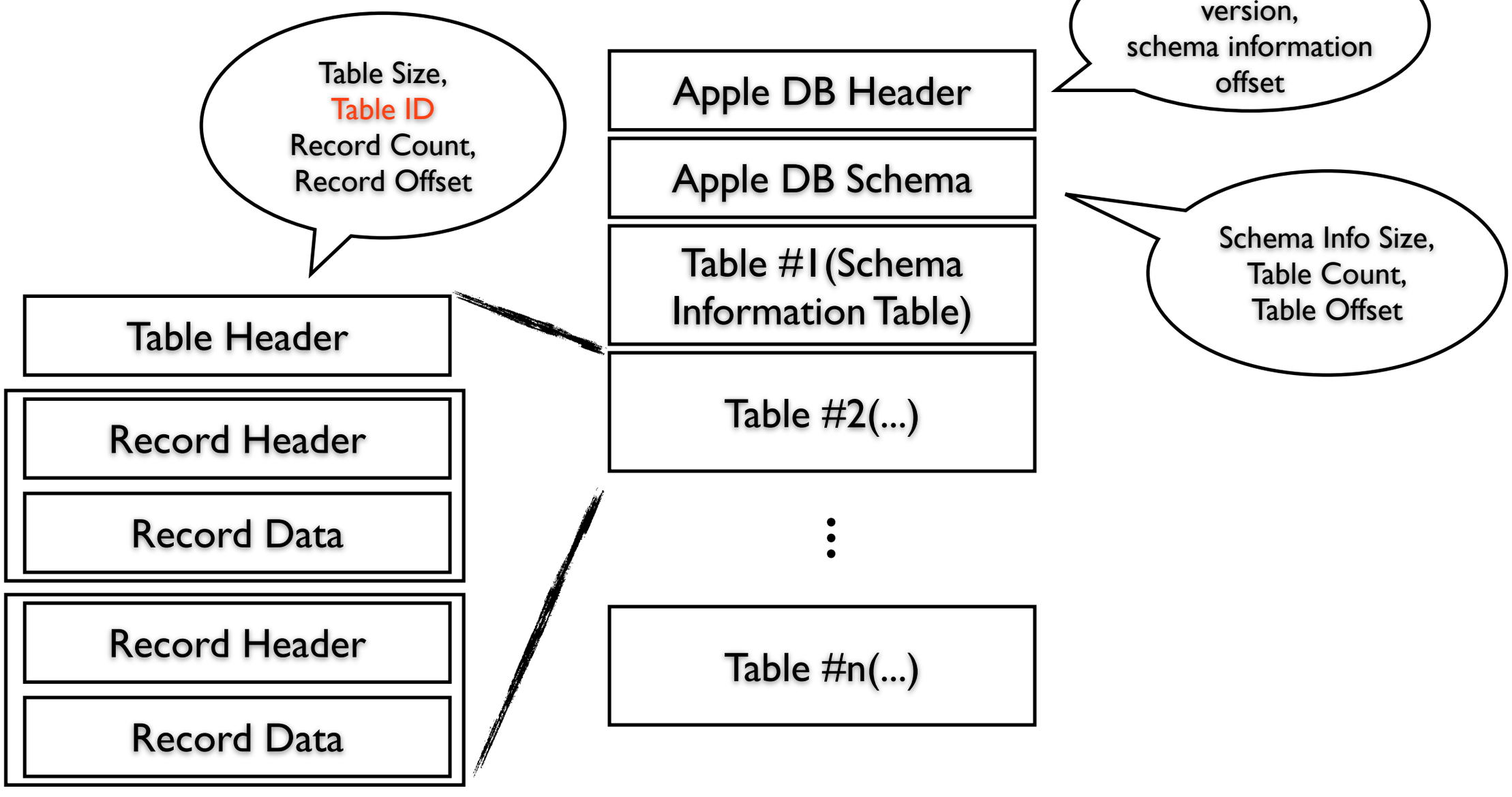


Table ID

```
typedef uint32 CSSM_DB_RECORDTYPE;
enum {
        /* Schema Management Name Space Range Definition*/
        CSSM_DB_RECORDTYPE_SCHEMA_START =                       0x00000000,
        CSSM_DB_RECORDTYPE_SCHEMA_END =                         CSSM_DB_RECORDTYPE_SCHEMA_START + 4,
        /* Open Group Application Name Space Range Definition*/
        CSSM_DB_RECORDTYPE_OPEN_GROUP_START =           0x0000000A,
        CSSM_DB_RECORDTYPE_OPEN_GROUP_END =                     CSSM_DB_RECORDTYPE_OPEN_GROUP_START + 8,
        /* Industry At Large Application Name Space Range Definition */
        CSSM_DB_RECORDTYPE_APP_DEFINED_START =          0x80000000,
        CSSM_DB_RECORDTYPE_APP_DEFINED_END =            0xffffffff,
        /* Record Types defined in the Schema Management Name Space */
        CSSM_DL_DB_SCHEMA_INFO =                                        CSSM_DB_RECORDTYPE_SCHEMA_START + 0,
        CSSM_DL_DB_SCHEMA_INDEXES =                                     CSSM_DB_RECORDTYPE_SCHEMA_START + 1,
        CSSM_DL_DB_SCHEMA_ATTRIBUTES =                          CSSM_DB_RECORDTYPE_SCHEMA_START + 2,
        CSSM_DL_DB_SCHEMA_PARSING_MODULE =                      CSSM_DB_RECORDTYPE_SCHEMA_START + 3,
        /* Record Types defined in the Open Group Application Name Space */
        CSSM_DL_DB_RECORD_ANY =                                         CSSM_DB_RECORDTYPE_OPEN_GROUP_START
        CSSM_DL_DB_RECORD_CERT =                                        CSSM_DB_RECORDTYPE_OPEN_GROUP_START
        CSSM_DL_DB_RECORD_CRL =                                         CSSM_DB_RECORDTYPE_OPEN_GROUP_START
        CSSM_DL_DB_RECORD_POLICY =                                      CSSM_DB_RECORDTYPE_OPEN_GROUP_START
        CSSM_DL_DB_RECORD_GENERIC =                                     CSSM_DB_RECORDTYPE_OPEN_GROUP_START
        CSSM_DL_DB_RECORD_PUBLIC_KEY =                  CSSM_DB_RECORDTYPE_OPEN_GROUP_START + 5,
        CSSM_DL_DB_RECORD_PRIVATE_KEY =                 CSSM_DB_RECORDTYPE_OPEN_GROUP_START + 6,
        CSSM_DL_DB_RECORD_SYMMETRIC_KEY =               CSSM_DB_RECORDTYPE_OPEN_GROUP_START + 7,
        CSSM_DL_DB_RECORD_ALL_KEYS =                    CSSM_DB_RECORDTYPE_OPEN_GROUP_START + 8
};
```

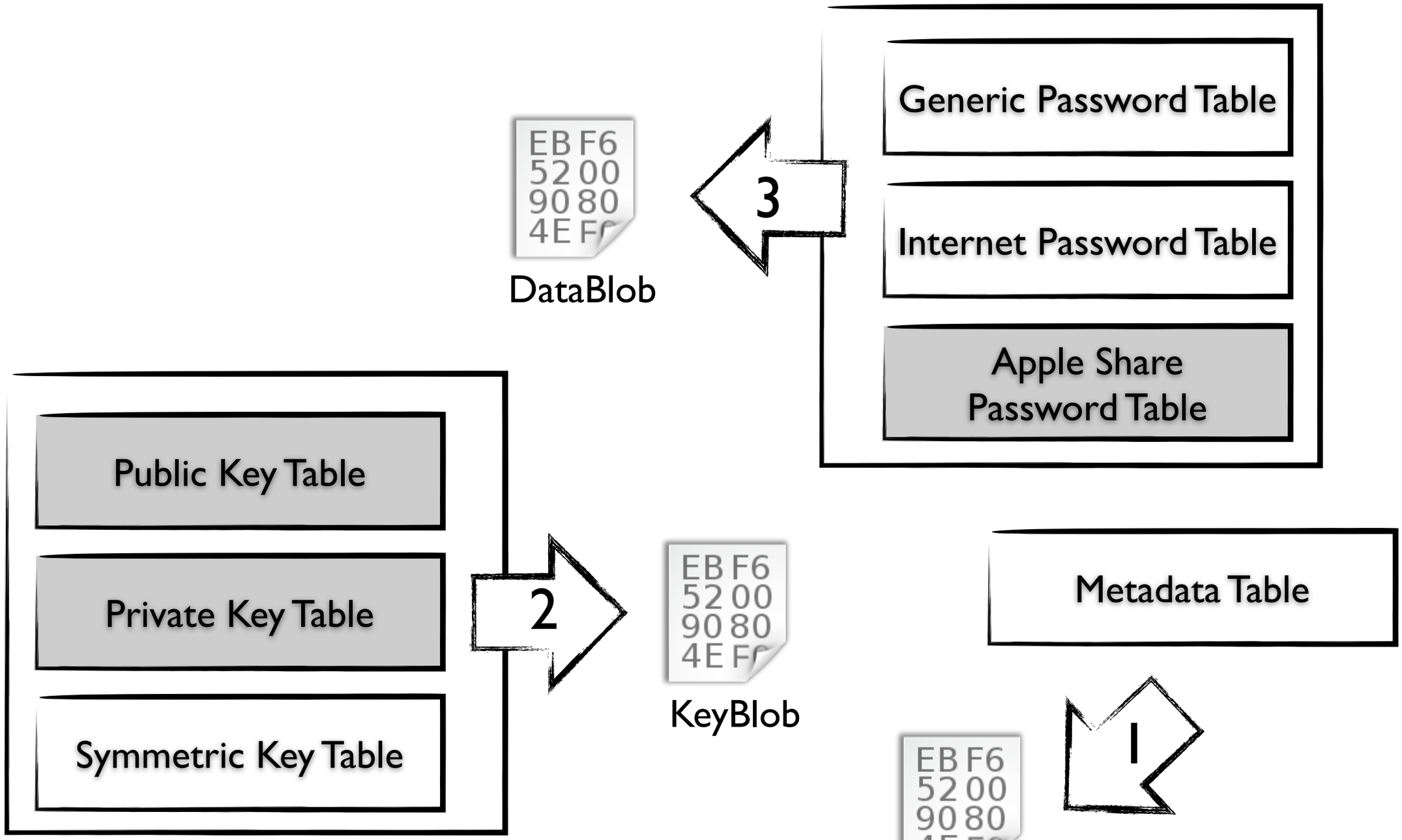
http://www.opensource.apple.com/source/libsecurity_cssm/libsecurity_cssm-6/lib/cssmtype.h

```
/* AppleFileDL record types. */
enum
{
    CSSM_DL_DB_RECORD_GENERIC_PASSWORD = CSSM_DB_RECORDTYPE_APP_DEFINED_START + 0,
    CSSM_DL_DB_RECORD_INTERNET_PASSWORD = CSSM_DB_RECORDTYPE_APP_DEFINED_START + 1,
    CSSM_DL_DB_RECORD_APPLESHARE_PASSWORD = CSSM_DB_RECORDTYPE_APP_DEFINED_START + 2,

    CSSM_DL_DB_RECORD_X509_CERTIFICATE = CSSM_DB_RECORDTYPE_APP_DEFINED_START + 0x1000,
        CSSM_DL_DB_RECORD_USER_TRUST,
        CSSM_DL_DB_RECORD_X509_CRL,
        CSSM_DL_DB_RECORD_UNLOCK_REFERRAL,
    CSSM_DL_DB_RECORD_METADATA = CSSM_DB_RECORDTYPE_APP_DEFINED_START + 0x8000
};
```

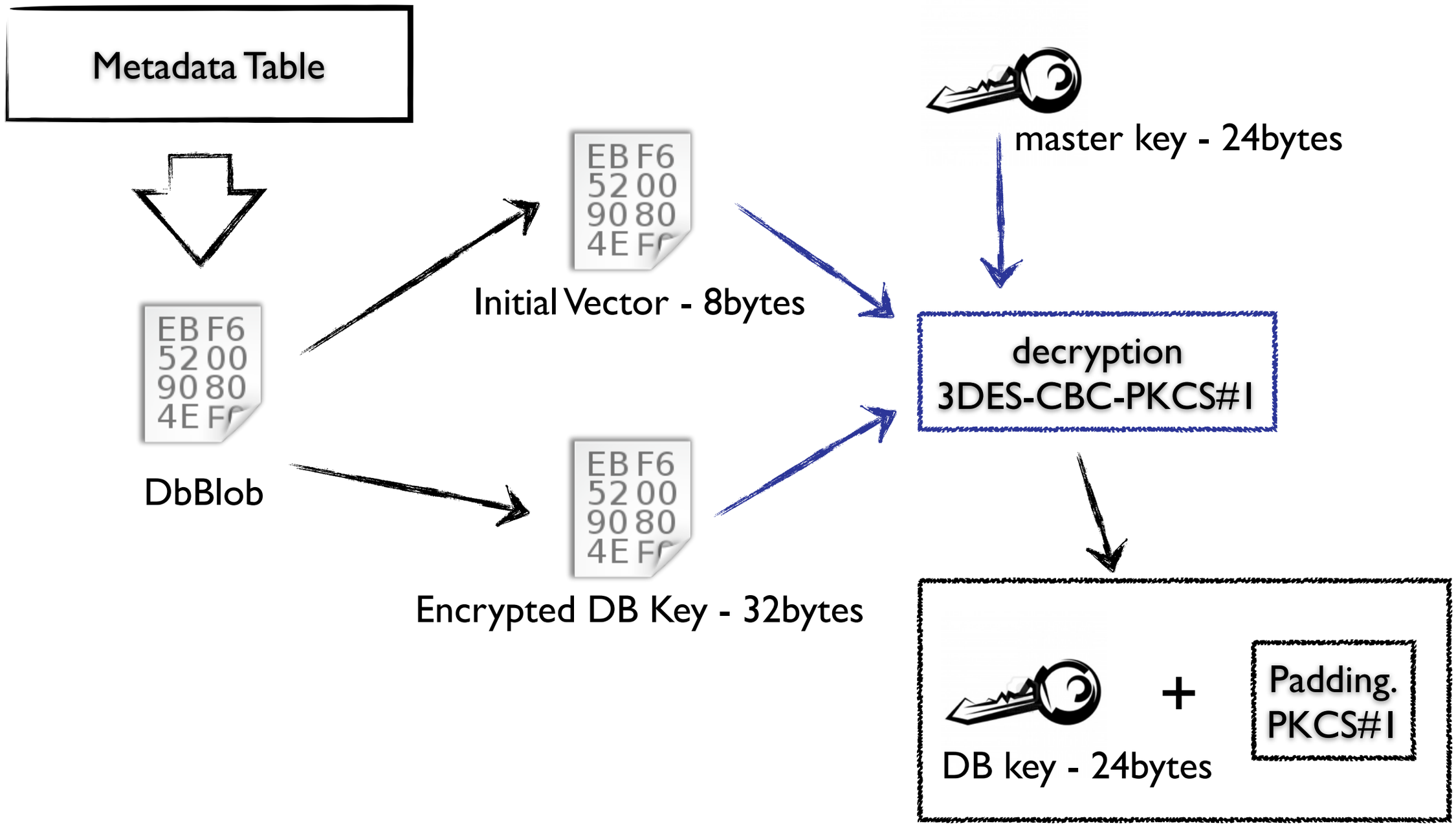
http://www.opensource.apple.com/source/libsecurity_cssm/libsecurity_cssm-36064/lib/cssmapple.h

Extract *-Blob



Icon : http://icdn.pro/images/en/h/e/hex-icone-7579-128.png

Decrypt a DB Key

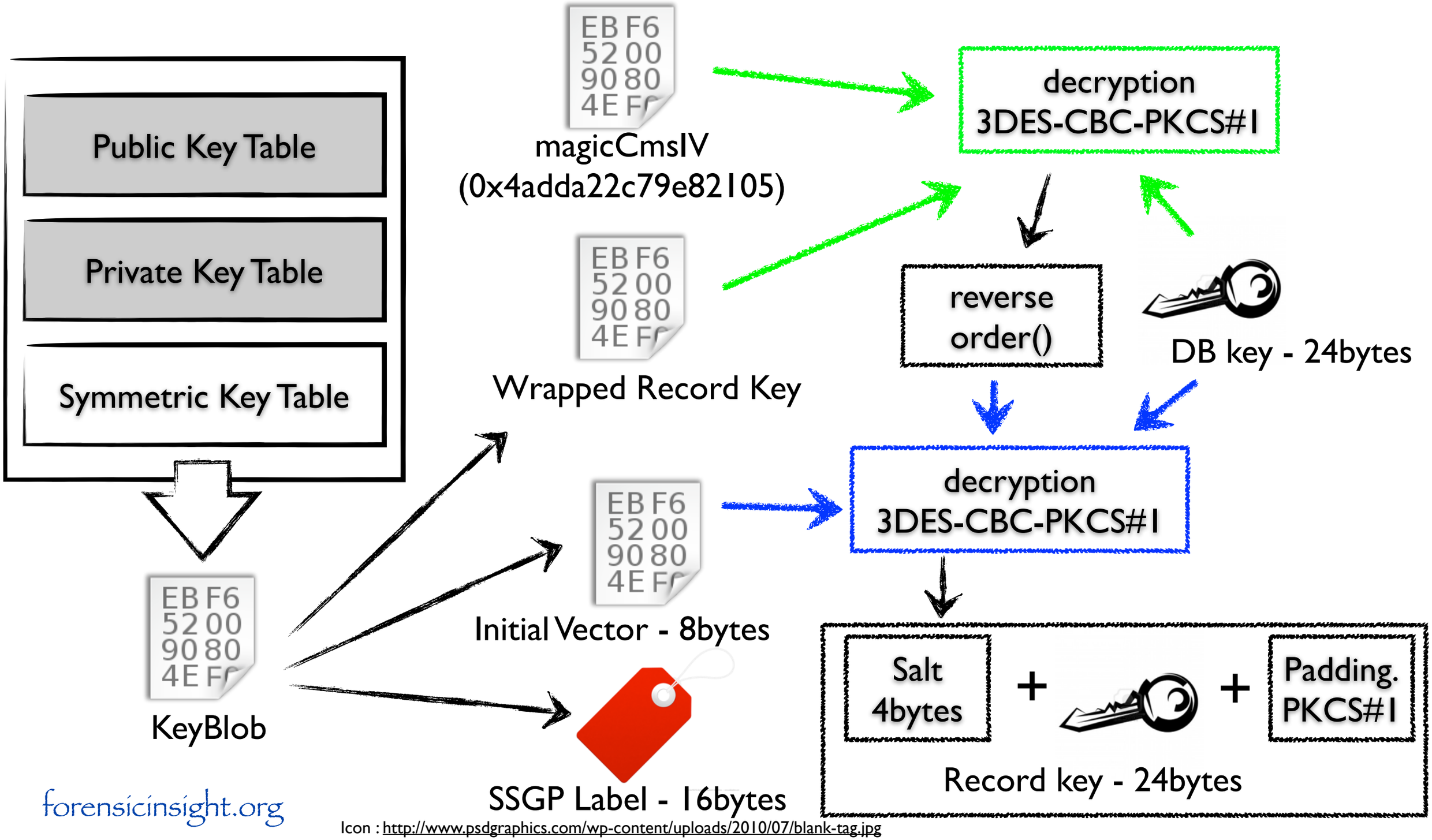


Decrypt a KeyBlob

```
void AppleCSPSession::unwrapKeyCms(
        CSSM_CC_HANDLE CCHandle,
        const Context &Context,
        const CssmKey &WrappedKey,
        const CSSM_RESOURCE_CONTROL_CONTEXT *CredAndAclEntry,
        CssmKey &UnwrappedKey,
        CssmData &DescriptiveData,
        CSSM_PRIVILEGE Privilege,
        cspKeyStorage keyStorage)
{
    /*
     * In reverse order, the steps from wrap...
     *
 * 5. Encrypt TEMP3 using DEK with an IV of 0x4adda22c79e82105 in CBC mode
     *    with PKCS1 padding call the result TEMP4.
     *
     *    TEMP4 is wrappedKey.KeyData.
     */
    const CssmData &wrappedBlob = CssmData::overlay(WrappedKey.KeyData);
    dumpBuf("unwrap inBlob", &wrappedBlob, 64);
    CssmData &IV1 = Context.get<CssmData>(CSSM_ATTRIBUTE_INIT_VECTOR,
                            CSSMERR_CSP_MISSING_ATTR_INIT_VECTOR);
    uint8 *savedIV = IV1.Data;
    uint32 savedIvLen = IV1.Length;
    IV1.Data = (uint8 *)magicCmsIv;
    IV1.Length = 8;
    CssmData TEMP3;
    uint32 bytesDecrypted;
```

http://www.opensource.apple.com/source/Security/Security-28/AppleCSP/AppleCSP/wrapKeyCms.cpp

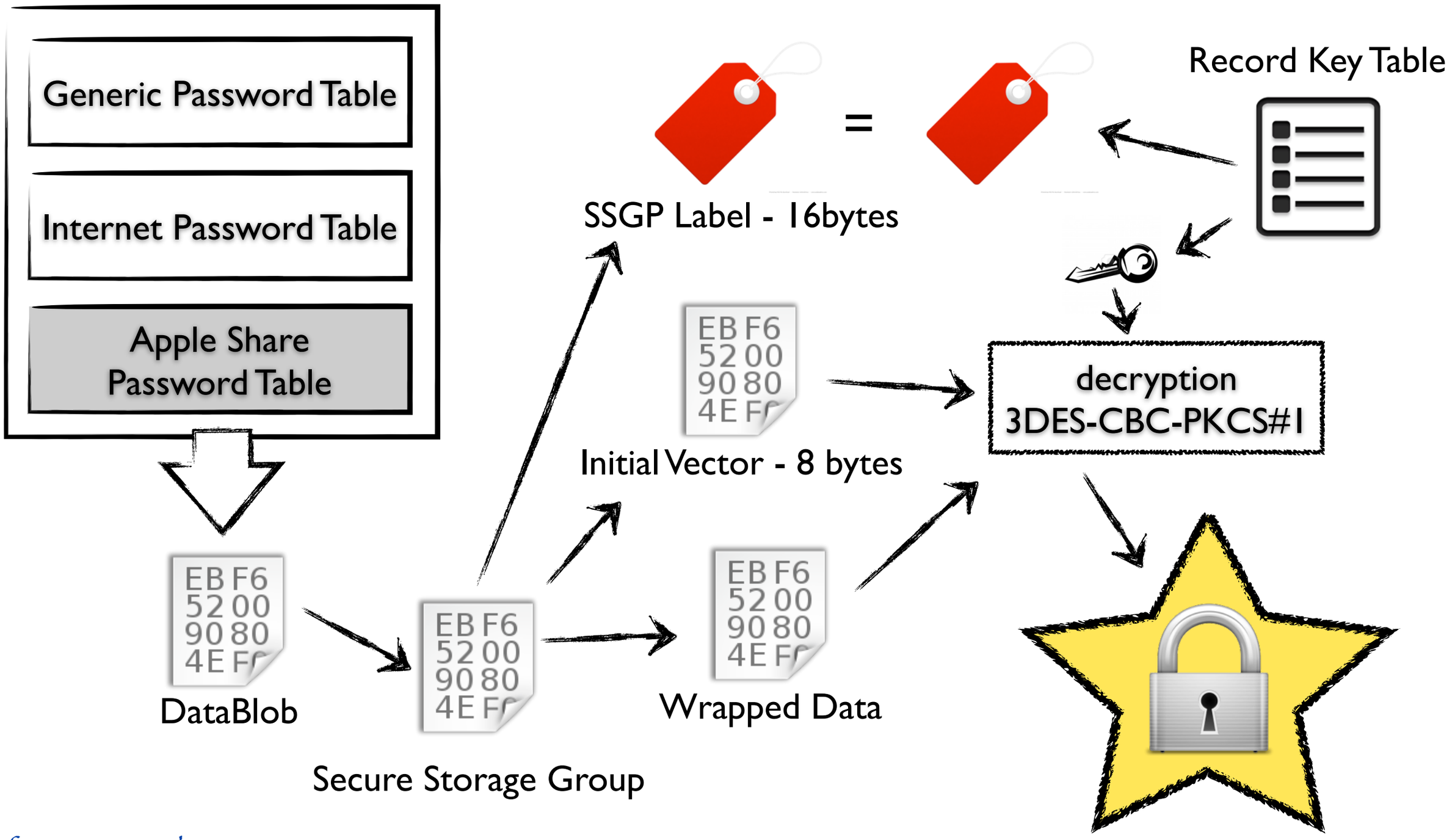
Decrypt a KeyBlob



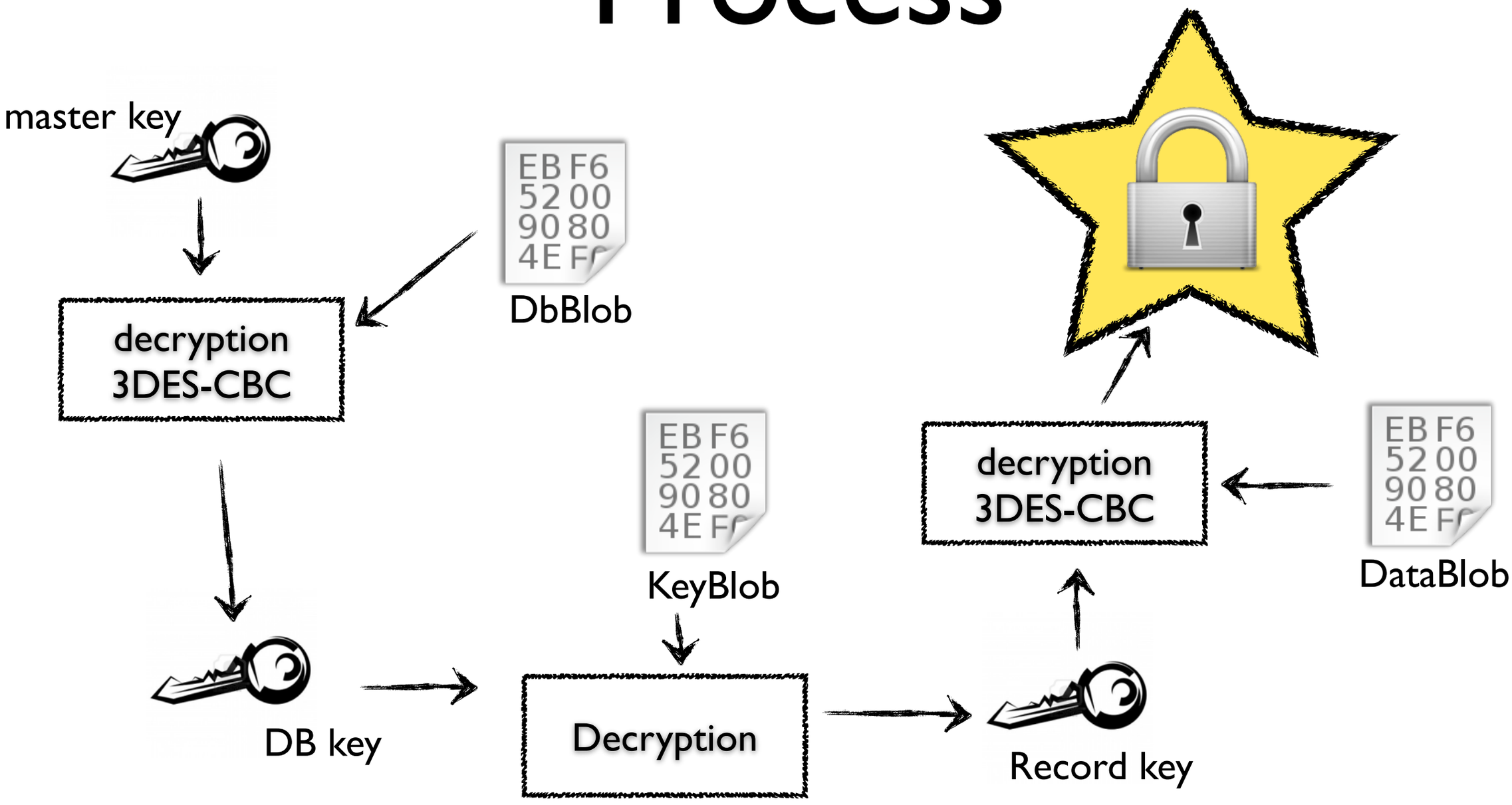
forensicinsight.org

Icon : http://www.psdgraphics.com/wp-content/uploads/2010/07/blank-tag.jpg

Decrypt a DataBlob



Data Decryption Process



Icon Reference : http://us.123rf.com/400wm/400/400/sooolnce/sooolnce1210/sooolnce12100021/15596242-house-key-black-and-white-vector.jpg

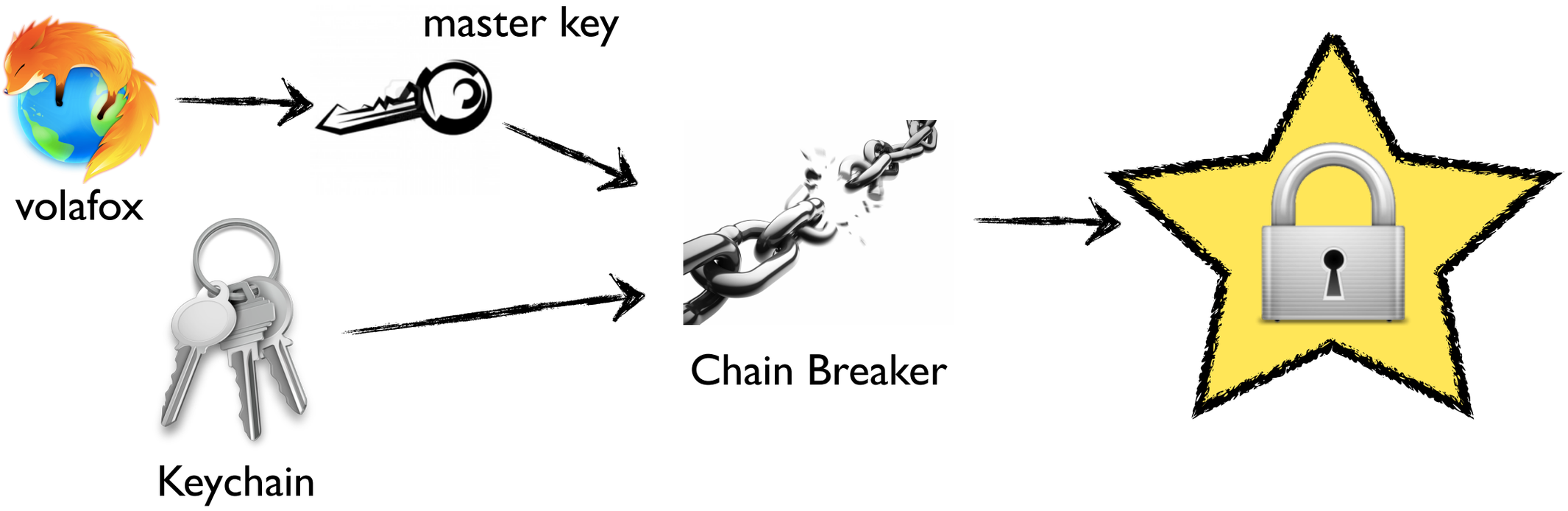
Icon Reference : http://files.softicons.com/download/system-icons/human-o2-icons-by-oliver-scholtz/png/128x128/mimetypes/text-x-hex.png

Chain Breaker

- Keychain analysis tool for digital investigator

Icon Source : http://unisonmeetings.files.wordpress.com/2012/08/the-broken-chain1.jpeg

Chain Breaker



- Language : Python 2.x
- Target OS : Mac OS X Lion & Mountain Lion

Show Time

Conclusion

- volafox keychain module + chainbreaker
  - Don’t require root privileges
    - has Integrity (use memory image)
    - has reliability (as we know it)
- It is very useful for digital investigator and *Forgetful Guy* ;-)

Q & A

nofate@nofate.com