
black hat
EUROPE 2018
DECEMBER 3-6, 2018
EXCEL LONDON / UNITED KINGDOM
Keeping Secrets:
Emerging Practice in Database Encryption
#BHEU / @BLACK HAT EVENTS

Keeping Secrets: Emerging Practice in Database Encryption

Kenneth White
@kennwhite

Goals

Highlight the gaps between real-world attack scenarios and the implicit security guarantees of most popular encrypted databases

Review recent advances & breaks in database encryption techniques

Look at emerging methods around data in-use & blind admin models

Provide architects and defenders with practical guidance for high-sensitivity workloads

A Brief History on Database Encryption...

A Brief History on Database Encryption…

- Transport

    SSL/TLS over native wire protocols

- Storage

    Volume encryption (FDE)

A Brief History on Database Encryption…

- Tables/tablespaces

    Transparent Data Encryption (TDE)/Encrypted Storage Engine (ESE)

        Oracle Server TDE

        SQLServer TDE

        MongoDB WiredTiger ESE

        MySQL Enterprise TDE

Current Market

- Microsoft/Azure
    - Transparent Data Encryption (TDE; server-side)
    - Always Encrypted engine (AE; client-side)
        - Deterministic
        - Randomized
    - SGX enclave encryption

Current Market

- CryptDB (Popa et al)
- Google
    - Encrypted BigQuery
        - CMKs - delegated
- Oracle
    - TDE with table- & column-level encryption

Current Market

- Postgres

    pgcrypto: DIY column-level

    PGP: home-brew AES constructions, etc.

Current Market

- MongoDB

    Wired Tiger ESE

        Atlas (BYOK w/ AWS KMS, Azure Vault, GCP KMS)

        Enterprise (native KMIP w/ HSM)

Current Market

- Amazon

    (this bullet will be obsolete in 3 months)

Broken Promises

Broken Promises

- Histograms & statistics views: DBA vs. DBA

- (some) format-preserving encryption

- (some) deterministic encryption

- Tokenization

- Cloud Access Brokers

Broken Promises

- Histograms & statistics views: DBA vs. DBA

Histograms & statistics views: DBA vs. DBA

Robert Lockard: An Oracle PoC

```
Important note: THIS IS ALL PSEUDO DATA, NOTING IS REAL.

-- the test customers table contains pseudo ssn's and cc numbers for demo purposes.
-- reality is, because cc_nbr and ssn are distinct, histograms should not be gathered,
-- however a "lazy" DBA may use the 'for all columns size skewonly' method_opt
-- therefore, by using the defaults you will get out 254 rows with data that should be encrypted.

create table t3 as select id, fname, lname, city, state, cc_nbr, ssn from customers;
alter table t3 modify (cc_nbr encrypt using 'AES256', SSN encrypt using 'AES256');

begin
  dbms_stats.gather_table_stats(null,'T3', method_opt=> 'for all columns size skewonly');
end;
/

desc t3
```

Source: Robert Lockard, https://web.archive.org/web/20180726160818/http://oraclewizard.com/Oraclewizard/2015/07/oracle-tde-dataleak-histograms/

```
RLOCKARD@pdev > desc t3
 Name Null?    Type
 ----------------------------------------------------------------
 ID NOT NULL NUMBER
 FNAME VARCHAR2(25)
 LNAME VARCHAR2(25)
 CITY VARCHAR2(25)
 STATE VARCHAR2(25)
 CC_NBR VARCHAR2(16) ENCRYPT
 SSN VARCHAR2(11) ENCRYPT
```

Source: Robert Lockard, https://web.archive.org/web/20180726160818/http://oraclewizard.com/Oraclewizard/2015/07/oracle-tde-dataleak-histograms/

```
  1  select
  2         endpoint_number,
  3         endpoint_actual_value
  4  from dba_tab_histograms
  5  where owner = 'RLOCKARD'
  6    and table_name = 'T3'
  7*   and column_name = 'SSN'
RLOCKARD@pdev > /

ENDPOINT_NUMBER ENDPOINT_ACTUAL_VALUE
--------------- --------------------------------
           4247 778294598
           4269 782777484
           4291 785731383
           4313 788768328
           4335 792928354
           4357 795685465
           4379 798987731
           4401 812732627
           4424 815857391
           4446 818188243
```

Source: Robert Lockard, https://web.archive.org/web/20180726160818/http://oraclewizard.com/Oraclewizard/2015/07/oracle-tde-dataleak-histograms/

```
> SELECT * FROM T3 WHERE SSN='778294598';

        ID FNAME LNAME                    CITY STATE                    CC_NBR
---------- ------------------------- ------------------------- ------------------------- --------------------
SSN
-----------
     41742 Monica Gaestel                 Strattanville Pennsylvania             3483712444144721
778294598


1 row selected.
```

Source: Robert Lockard, https://web.archive.org/web/20180726160818/http://oraclewizard.com/Oraclewizard/2015/07/oracle-tde-dataleak-histograms/

Broken Promises

- Histograms & statistics views: DBA vs. DBA

- (some) format-preserving encryption

PUBLICATIONS

SP 800-38G

Recommendation for Block Cipher Modes of Operation: Methods for Format-Preserving Encryption

Date Published: March 2016

Author(s)

Morris Dworkin (NIST)

Abstract

This Recommendation specifies two methods, called FF1 and FF3, for format-preserving encryption. Both of these methods are modes of operation for an underlying, approved symmetric-key block cipher algorithm.

Keywords

block cipher; confidentiality; encryption; FF1; FF3; format-preserving encryption; information security; mode of operation

DOCUMENTATION

Publication:
SP 800-38G (DOI)
Local Download

Supplemental Material:
Press Release (other)

TOPICS

Security and Privacy

UPDATES 2017

Recent Cryptanalysis of FF3

April 12, 2017

Two researchers, Betül Durak (Rutgers University) and Serge Vaudenay (Ecole Polytechnique Fédérale de Lausanne), have given NIST early notification of a cryptanalytic attack on the FF3 technique for format-preserving encryption (FPE). The researchers gave a presentation of their work at the ESC 2017 Conference in January, and the details of the attack are expected to be published in the coming year. FF3 is specified and approved in NIST Special Publication 800-38G as a mode of operation of the Advanced Encryption Standard (AES) block cipher algorithm. NIST has concluded that FF3 is no longer suitable as a general-purpose FPE method.

https://www.businesswire.com/news/home/20170530005151/en/Protegrity-Warns-NIST-Approved-Format-Preserving-Encryption-FPE-Standard

Protegrity Warns That NIST-Approved Format-Preserving Encryption (FPE) Standard May Leave Organizations Vulnerable to Attack

In a recent news alert, NIST described how two researchers performed a cryptanalytic attack on the FF3 technique for format-preserving encryption, demonstrating that FF3 clearly does not achieve the intended 128-bit security level, even for 9-digit decimal strings like Social Security numbers. “For any significantly smaller domains of confidential data--including the middle-six digits of credit card numbers, the format that FF3 was designed to encrypt--the level of computation for the attack might be practical for many attackers,” the alert stated.

NIST expects to revise Special Publication 800-38G later this year after the details of the attack are published, either to change the FF3 specification, or to withdraw the approval of FF3. NIST originally considered three FPE modes called FF1, FF2, and FF3. FF2 did not survive to publication and now FF3 has been broken by researchers.

“It’s unfortunate an attack vector was found in FF3 less than a year into being named a standard, but we will continue to monitor ongoing developments, and will support any future improvements to the FF3

Broken Promises

- Histograms & statistics views: DBA vs. DBA

- (some) format-preserving encryption

- (some) deterministic encryption

- Tokenization

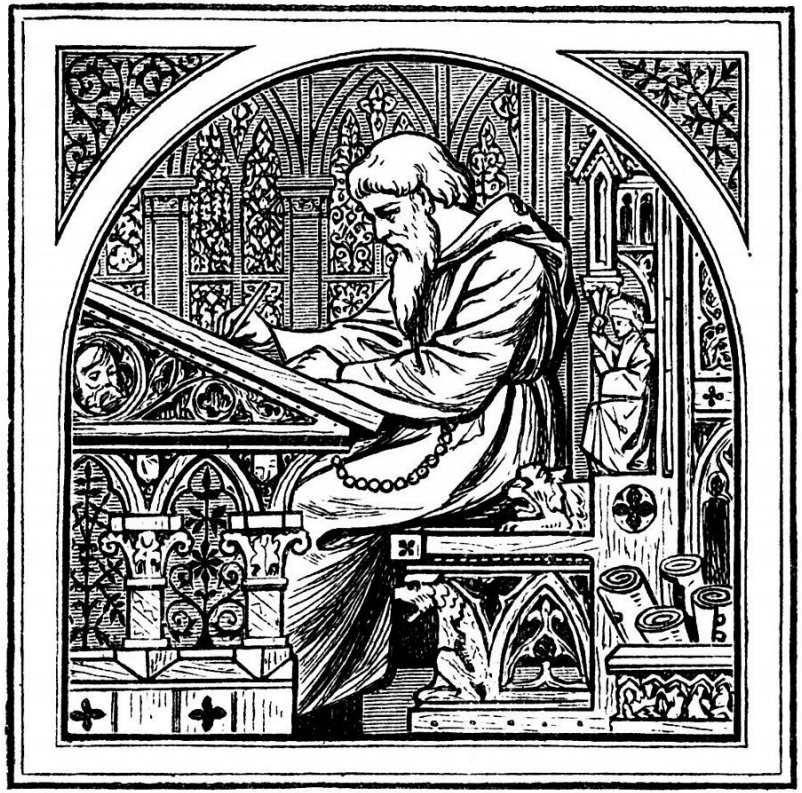
- Cloud Access Brokers

The threat model of most encrypted databases

Source: Imgur, author unknown

Your threat model is wrong, but your database is worse.

Breaking next-gen crypto in 2018 with 9th century frequency analysis



Source: Wikimedia CC

Your threat model is wrong, but your database is worse

- Breaking next-gen crypto in 2018 with 9th century frequency analysis

  Inference attacks on property-preserving encrypted databases

  Wright, Naveed, Kamara

- Logs, diagnostics, in-memory structures, oh my!

  Why your database is not secure

  Grubbs, Ristenpart, Shmatikov

Thinking beyond naive on/off key rotation lifecycle: Lessons from Google & Amazon scaling

AWS key management service (KMS): Handling cryptographic bounds for use of AES-GCM

Campagna & Gueron (Amazon)

Achieving high availability in the internal Google key management system

Kanagala, et al (Google)

First Principles

- Threat model-driven design
- My game over is not your game over
- RAM is the achilles heel of confidentiality
- Snapshot attackers will usually win, but you probably already lost
- Thinking through zero knowledge

First Principles

- Sane defenses
- Rate-limiting
- Segmentation
- Partial views/visibility (excellent use case for rational encryption)
- Real time anomaly detection & response

First Principles

- Savage key segregation

"*Of course you'd use sane key management & identity access policy.*"

— *Cryptographers*

"*We need to give all of Finance, Accounting, HR, and Helpdesk the key.*"

— *Senior Management*

"*This web app has* `[select * from *]` *& a hard-coded HSM API token.*"

— *Production Ops*

If your security sucks now without identity management,

you'll be pleasantly surprised

by the lack of change with encryption.

First Principles

Game out your own attacks before the bad guys do it for you

> *"You're on the Internet. You're already getting the pen test, just not the report"*
>
> — *Zane Lacke*

Emerging

- Secure enclave hardware
- Geo-attestation/location assurance
- Instance-based identity/temporary credentials
- Sane FDE & key management
- Homomorphic encryption
- Attribute-based (multi-party) encryption

Recommended Reading

- Microsoft Always Encrypted engine overview
- Oracle Column-Mode Transparent Data Encryption
- Deterministic & randomized encryption modes
- Guidelines for Using the CryptDB System Securely (Popa et al)
- Outsourcing the Decryption of ABE Ciphertexts
- Searchable Symmetric Encryption. Kamara & Moataz
- Inference Attacks on Property-Preserving Encrypted Databases (MSR)
- Adrian Colyer analysis on Grubbs et al
- Searchable Symmetric Encryption Implementation: Clusion (Kamara Lab)

Black Hat Sound Bytes

- Most encrypted database **security models are weak/underspecified**
    - Encrypted DB disks protect against eBay & Craigslist attacks, not Amazon, Microsoft, Google (and, only minimally, their customers)

- You ***may*** have to think about: court orders/discovery and motivated advanced attackers

- You ***do*** have to think about key surface/exposures, AppSec, SQLi, bearer tokens, API intercepts, backups, logs, sysadmins, DBAs…

Questions?

Kenneth White
@kennwhite