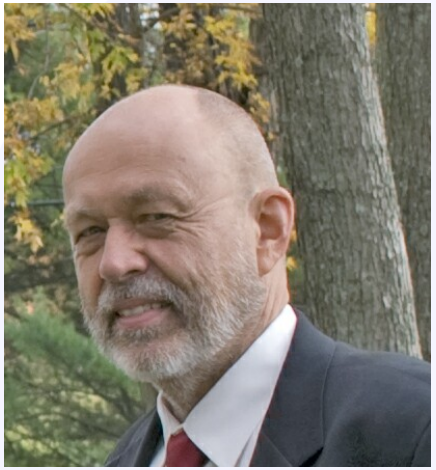
How to Secure Your Data - Informix Column Encryption and Security

by Lester Knutsen

November 10, 2014

Lester Knutsen



Lester Knutsen is President of Advanced DataTools Corporation, and has been building large Data Warehouse and Business Systems using Informix Database software since 1983. Lester focuses on large database performance tuning, training and consulting. Lester is a member of the IBM Gold Consultant program and was presented with one of the Inaugural IBM Data Champion awards by IBM. Lester was one of the founders of the International Informix Users Group and the Washington Area Informix User Group.

lester@advancedatatools.com
www.advancedatatools.com
703-256-0267 x102

How to Secure your Data - Informix Column Encryption and Security

Agenda

- Informix Column Encryption
- Demo – Implementing Encryption with Minimal Impact
- Managing Users with Roles

Requirements

- Need to Protect - **"Personally identifiable information"** (**PII**)
  - *Information which can be used to distinguish or trace an individual's identity, such as their name, social security number, biometric records, etc. alone, or when combined with other personal or identifying information which is linked or linkable to a specific individual, such as date and place of birth, mother's maiden name, etc.*
- Need to Secure Financial Data
- Many different rules (local, state, national, financial) - You need to find out what yours are!

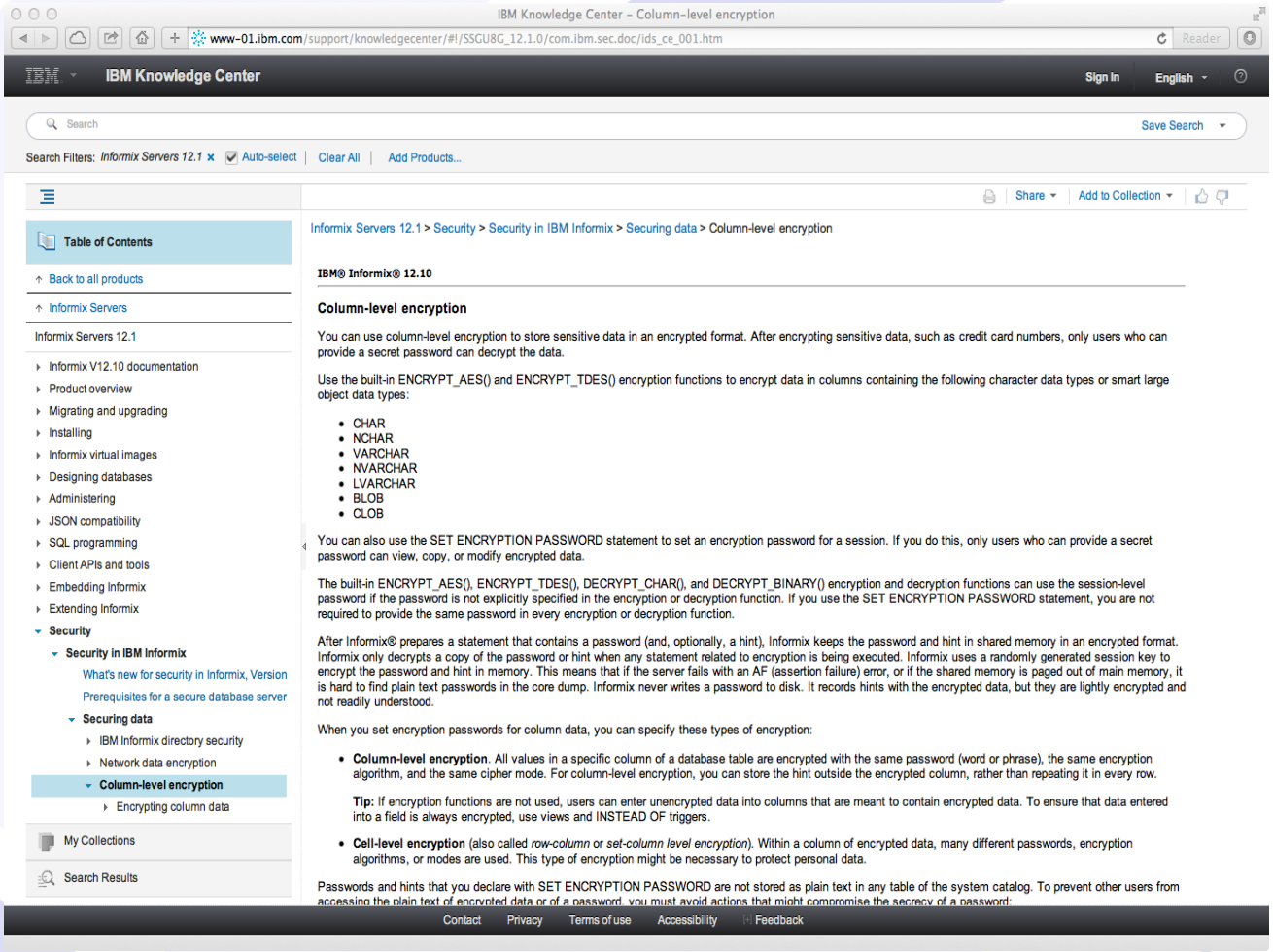
Common Encryption Requirements

- Credit Card Info
- Bank Info
- Social Security Number

Problem

- How to you add encryption to an exiting database and application?
- With minimal change to the application
- With minimal impact to users?

IBM Knowledge Center on Column Encryption



Informix Encryption Functions

- SET ENCRYPTION PASSWORD to set an encryption password or encryption key for a session
- ENCRYPT_AES() and ENCRYPT_TDES() functions to encrypt data in columns
- DECRYPT_CHAR(), and DECRYPT_BINARY() decrypt functions

Encryption Data Types

- CHAR
- NCHAR
- VARCHAR
- NVARCHAR
- LVARCHAR
- BLOB
- CLOB

Encrypt Virtual Processor (oninit)

- Started at startup if defined in the ONCONFIG
  - VPCLASS encrypt,num=1
- First time an encryption function is called if not defined in the ONCONFIG
- Start from the command line
  - onmode –p 1 encrypt

Encrypt Virtual Processor (oninit)

- Informix keeps the password and hint in shared memory in an encrypted format
- Informix uses a randomly generated session key to encrypt the password and hint in memory.
- **Not stored on disk or in plain text in memory or in AF dumps**

Encrypt Function

- ENCRYPT_AES()
  - Advanced Encryption Standard
- ENCRYPT_TDES()
  - Triple Data Encryption Standard, which is sometimes also called DES3 algorithm
  - Slower but considered more secure

Encrypt Function Arguments

- Data to be encrypted (required)
- Encryption password or Key (required when session password is not set)
- Encryption Hint (optional)
  - Use GETHINT to retrieve the Hint
- Example
  - set encryption password (abc)
  - encrypt_ssn = encrypt_tdes (ssn)

Encryption Column Size

- The encrypted data will be larger than the original unencrypted data
- The size of the encrypted column must be large enough to store the full encrypted data
- If column is too small, data will be truncated and cannot be decrypted

Determine Encryption Column Size

```
select length
    ( encrypt_tdes  ("1234567890123456", "password", “hint"))
from sys tables where tabid = 1;
```

- Or without a hint

```
select length
    ( encrypt_tdes  ("1234567890123456", "password”,))
from sys tables where tabid = 1;
```

Manage Encryption Password

- The key to encrypt or decrypt data
- All data will be lost if key is lost
- Where do you keep your password?
  - Encrypted table in database which only the user Informix or designated user
  - Compiled in a C UDR

Demo and Examples

How to take an Informix database and change it to use encrypted columns with little impact to your front end applications using triggers, functions, and Informix encryption.

Example – Base Table

```
create table new_customer
  (
    customer_num serial not null ,
    fname char(15),
    lname char(15),
    company char(20),
    address1 char(20),
    address2 char(20),
    city char(15),
    state char(2),
    zipcode char(5),
    phone char(18),
    ssn char(9),
    tin char(10),
    primary key (customer_num)
  );

---- Load data into the New Customer table
insert into new_customer
select c.*,
       phone[1,3] ||phone[5,6] || phone[9,12], -- Demo SSN
       zipcode || phone[10,12] || phone[2,3]     -- Demo TIN
from customer c
;
```

Example – Add Encrypted Fields

----- Add 2 fields to hold the Encrypted Data

alter table new_customer add (encrypt_ssn varchar(50) before tin);

alter table new_customer add (encrypt_tin varchar(50));

Example – Batch Encrypt Data

```
-- Mass Encrypt the SSN and TIN in the new customer table
set encryption password (( select trim(password) from
encrypt_key ));

update new_customer set  encrypt_ssn = encrypt_tdes (ssn)
          where 1=1;
update new_customer set encrypt_tin = encrypt_tdes (tin)
          where 1=1;
-- hide old data
update new_customer set ssn = "*****" ll ssn[6,9] where 1=1;
update new_customer set tin = "******" ll tin[7,10] where 1=1;
```

Example – Create Trigger to Encrypt New Rows

```
create trigger ins_cust insert on new_customer
referencing new as post

for each row
(
        execute procedure sp_do_encryption('new_customer',
        post.customer_num),
         execute function trunc_to_4('new_customer',
        post.customer_num ) INTO ssn, tin
);
```

Example – Encryption SPL

```
--- Create a Procedure to Encrypt the SSN and TIN
CREATE PROCEDURE sp_do_encryption( tabname varchar(60), in_val integer)
    Define esql, eisam smallint;
    ON EXCEPTION SET esql, eisam
    END EXCEPTION
set lock mode to wait;
set encryption password (select password from encrypt_key);
if tabname = "new_customer" then
   update new_customer set
            encrypt_ssn = (case when ssn[1] != "*" then encrypt_tdes(trim(ssn)) else " end),
            encrypt_tin = (case when tin[1] != "*" then encrypt_tdes(trim(tin)) else "
end)
    where customer_num = in_val ;
end if
END PROCEDURE;
```

Example – Hide Old Data SPL

```
-- Create a Procedure to Hide the Old SSN and TIN
create function trunc_to_4( tabname char(60), in_val integer )
  returning char(10), char(10);
  Define ln1, ln2 smallint;
  Define l_ssn, l_tin char(60);
  Define val1, val2, val3 char(10);

  if tabname = 'new_customer' then
    select length(ssn), ssn, length(tin), tin
      into ln1, l_ssn, ln2, l_tin from new_customer
     where customer_num = in_val;

                if l_ssn[1] != "*" then
                        let val1 = "*****" ||  l_ssn[6,9];
                else
                        let val1 = l_ssn;
                end if
                if l_tin[1] != "*" then
                        let val2 = "******" || l_tin[7,10];
                else
                        let val2 = l_tin;
                end if
  end if
  return val1, val2;
end function;
```

Example – Decrypt Rows

```
select
        customer_num,
        lname,
        ssn,
        decrypt_ssn(customer_num) unencrypt_ssn,
        tin,
        decrypt_tin(customer_num) unencrypt_tin,
        encrypt_ssn,
        encrypt_tin
from new_customer;
```

Example – Decrypt SPL

```
create function decrypt_ssn ( in_val integer )
  returning  char(10);
  Define val1 char(10);
            -- Check if user is allowed to see data
            if USER  = "informix"  then
                        set encryption password (select password from encrypt_key);
                        select ( decrypt_char(encrypt_ssn)::char(10) ) into val1
                        from new_customer
                        where customer_num = in_val;
            else
                        let val1 = NULL;
            end if
  return val1;
end function;
```

To Add Encryption to an Existing Informix Database Application

Example Summary

Example Summary

1. Create the columns to hold the encrypted data
   - In separate shadow tables or
   - At the end of the current table
2. Create an SPL function to encrypt and hide the existing data
3. Create trigger to encrypt and hide all data on insert

Example Summary

4. Grant permissions on the SPL – allow everyone to encrypt the data
5. Create a function to decrypt the data
6. Grant permissions on the decrypt SPL – only authorized users can see the data
7. ***The only applications that need to change are the ones that need to show the decrypted data!***

Managing Users with Roles

- First article on Roles
  - **Roles - A New Security Feature in INFORMIX OnLine 7.10.UD1**

    by Lester Knutsen

    October 1995

- http://advancedatatools.com/TechInfo/WAIUG/iugnew54.htm#roles

Roles

- Database privileges are granted to Roles, rather than individual users
- Users are added to Roles
- Roles can be set on login with the sysdbopen() function

Roles

- Reduce the complexity of managing thousands of users to:
  - Create and manage a few roles
  - Assign users to roles

Roles - Syntax

- Create Role statement
- Grant Role to User statement
- Grant Privileges to Role Statement
- Set Role statement

Questions?

Send follow-up questions to
lester@advancedatatools.com

Informix Training in 2015

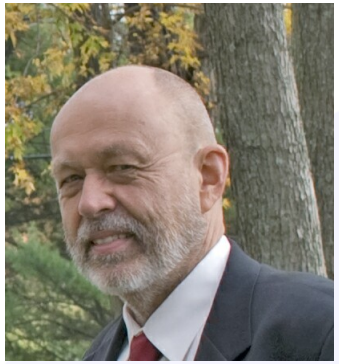
January 26-29, 2015 - Informix for Database Administrators
March 16-19, 2015 - Informix Enterprise Replication (New Course)
July 20-23, 2015 - Advanced Informix Performance Tuning
October 12-15, 2015 - Informix for Database Administrators

- All courses can be taken online over the web from your desk, or at our training center in Virginia.
- We guarantee to *NEVER* cancel a course and will teach a course as long as one student is registered!

Next Webcast

- Date: TBD January 2015
- Time: 2:00pm EST

- Fastest Informix DBA Contest 2015
- Informix and the Internet of Things:
  - Informix and embedded Devices/ARM and Informix – Tom Beebe
  - Connecting your database to the outside world – Mike Walker



Thank You

Lester Knutsen

Advanced DataTools Corporation

lester@advancedatatools.com

For more information:

http://www.advancedatatools.com