|  |
| --- |
| HỌC VIỆN KỸ THUẬT MẬT MÃ  **KHOA CÔNG NGHỆ THÔNG TIN**  Logo HvKTMM  **AN TOÀN CƠ SỞ DỮ LIỆU**  **Đề tài 15:**  **TÌM HIỂU CÁC CƠ CHẾ MÃ HÓA TRONG DB2 VÀ THỰC NGHIỆM**  Sinh viên thực hiện: ĐẶNG THỊ THU HÒA (AT150220)  ĐỖ VĂN NAM (AT150530)  NGUYỄN ĐÌNH LÂM (AT150230)  ĐẶNG TIẾN THÀNH (AT150252)  Giảng viên hướng dẫn: TS. TRẦN THỊ LƯỢNG    Hà Nội, 24-9-2021 |

# ****LỜI CẢM ƠN****

Mỗi đề tài là một thách thức mới đối với chúng em. Mỗi thách thức lại làm cho chúng em trở lên hoàn thiện hơn trong con đường học tập và nghiên cứu sau này. Chúng em xin chân thành cảm ơn TS. Trần Thị Lượng – Khoa An toàn Thông tin Học viện Kỹ thuật Mật mã đã có những đề tài hay cũng như tạo điều kiện tốt nhất để chúng em có thể hoàn thành đề tài này.

Chúng em xin chân thành cảm ơn.

MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc85066792)

[LỜI MỞ ĐẦU 4](#_Toc85066793)

[CHƯƠNG 1. GIỚI THIỆU SSL 5](#_Toc85066794)

[1.1 Khái niệm 5](#_Toc85066795)

[1.2 Lợi ích sử dụng 6](#_Toc85066796)

[1.3 Phương thức hoạt động trên DB2 6](#_Toc85066797)

[1.4 Thiết lập cấu hình 7](#_Toc85066798)

[CHƯƠNG 2. IBM Security Guardium: giải pháp bảo mật dữ liệu toàn diện 13](#_Toc85066799)

[2.1 Giải pháp giúp phân tích, bảo vệ và đáp ứng để bảo mật dữ liệu toàn diện 13](#_Toc85066800)

[2.2 Bảo mật dữ liệu toàn diện 14](#_Toc85066801)

[2.3 Phân tích các mối đe dọa đối với dữ liệu nhạy cảm 14](#_Toc85066802)

[2.4 Bảo vệ dữ liệu nhạy cảm 16](#_Toc85066803)

[2.5 Thích nghi với sự thay đổi 17](#_Toc85066804)

[2.6 Guardium mang tới giá trị cho một loạt các ngành công nghiệp 18](#_Toc85066805)

[2.7 Về giải pháp của IBM Security 19](#_Toc85066806)

[CHƯƠNG 3. Mã hóa dữ liệu dùng hệ thống file mã hóa AIX (AIX EFS) 19](#_Toc85066807)

[3.1 Giới thiệu về hệ thống file mã hóa AIX 19](#_Toc85066808)

[3.1.1 Khái niệm 19](#_Toc85066809)

[3.1.2 Điều kiện tiên quyết 19](#_Toc85066810)

[3.2 Vai trò 20](#_Toc85066811)

[3.3 Thiết lập 21](#_Toc85066812)

[3.3.1 Bật EFS trên hệ thống 21](#_Toc85066813)

[3.3.2 Kích hoạt thuộc tính V2 21](#_Toc85066814)

[3.3.3 Mã hóa file 24](#_Toc85066815)

[3.3.4 Giải mã một tệp tin 29](#_Toc85066816)

[3.3.5 Kế thừa mã hóa 29](#_Toc85066817)

[3.3.6 Cấp quyền truy cập cho người dùng khác 29](#_Toc85066818)

[CHƯƠNG 4. DB2 NATIVE ENCRYTION 31](#_Toc85066819)

[4.1 Giới thiệu về DB2 native encrytion 31](#_Toc85066820)

[4.1.1 Một số khái niệm về db2 native encrytion 31](#_Toc85066821)

[4.1.2 Vai trò 32](#_Toc85066822)

[4.1.3 Thiết lập cấu hình 32](#_Toc85066823)

[CHƯƠNG 5. Kết luận và tài liệu tham khảo 37](#_Toc85066824)

[5.1.1 Kết luận 37](#_Toc85066825)

[5.1.2 Tài liệu tham khảo 38](#_Toc85066826)

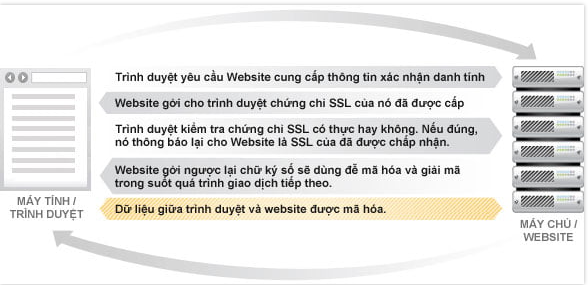
# LỜI MỞ ĐẦU

Hiện nay, thời đại công nghệ thông tin đã được toàn cầu hóa với sự trợ giúp của mạng Internet. Cùng với sự phát triển của công nghệ thông tin đã thúc đấy sự phát triển kinh tế, xã hội. Các hoạt động của các tổ chức xã hội như: Kinh tế, chính trị, văn hóa, quân sự…Ngoài những thông tin được chia sẻ của các tổ chức lên mạng Internet hay các trang mạng xã hội. Bên cạnh đó có những thông tin việc mất mát, rò rỉ thông tin có thể ảnh hưởng nghiêm trọng đến tài chính, sự tồn vong của tổ chức. Các phương thức tấn công thông qua mạng ngày càng tinh vi, phức tạp có thể dẫn đến mất mát thông tin, thậm chí có thể làm sụp đổ hoàn toàn hệ thống thông tin của tổ chức. Vì vậy an toàn và bảo mật thông tin là nhiệm vụ rất nặng nề và khó đoán trước được, nhưng tựu trung lại gồm ba hướng chính sau: Bảo đảm an toàn thông tin tại máy chủ; Bảo đảm an toàn cho phía máy trạm; Bảo mật thông tin trên đường truyền.   
 Vậy vấn đề đặt ra là dự liệu khi được lưu trữ trên may chủ, máy trạm và di chuyển trên đường truyền làm sao bảo về được dữ liệu và chống được các truy cập trái phép cơ sở dữ liệu. Nếu dữ liệu vẫn bị hacker tấn công và chiếm đọt quyền truy cập cơ sở dữ liệu, làm sao thông tin vẫn không bị rò rỉ, mất mát. Đó là dữ liệu trước khi lưu trữ ở máy trạm, máy chủ và truyền đi phải được mã hóa. Đây là nội dung chính trong tiểu luận của nhóm em về đề tài: Các cơ chế mã hóa trong DB2. Mặc dù đã rất cố gắng nhưng tiểu luận này không tránh khỏi những sai sót. Nhóm chúng em rất mong nhận được các ý kiến góp ý của cô và các bạn.

# GIỚI THIỆU SSL

## Khái niệm

**SSL là viết tắt của từ Secure Sockets Layer. SSL là tiêu chuẩn của công nghệ bảo mật, truyền thông mã hoá giữa máy chủ Web server và trình duyệt.** Tiêu chuẩn này hoạt động và đảm bảo rằng các dữ liệu truyền tải giữa máy chủ và trình duyệt của người dùng đều riêng tư và toàn vẹn. SSL hiện tại cũng là tiêu chuẩn bảo mật cho hàng triệu website trên toàn thế giới, nó bảo vệ dữ liệu truyền đi trên môi trường internet được an toàn.



*Giao thức bảo mật SSL*

## Lợi ích sử dụng

* Xác thực website, giao dịch.
* Nâng cao hình ảnh, thương hiệu và uy tín doanh nghiệp.
* Bảo mật các giao dịch giữa khách hàng và doanh nghiệp, các dịch vụ truy nhập hệ thống.
* Bảo mật webmail và các ứng dụng như Outlook Web Access, Exchange, và Office Communication Server.
* Bảo mật các ứng dụng ảo hó như Citrix Delivery Platform hoặc các ứng dụng điện toán đám mây.
* Bảo mật dịch vụ FTP.
* Bảo mật truy cập control panel.
* Bảo mật các dịch vụ truyền dữ liệu trong mạng nội bộ, file sharing, extranet.
* Bảo mật VPN Access Servers, Citrix Access Gateway …
* Website không được xác thực và bảo mật sẽ luôn ẩn chứa nguy cơ bị xâm nhập dữ liệu, dẫn đến hậu quả khách hàng không tin tưởng sử dụng dịch vụ.

## Phương thức hoạt động trên DB2

* Tạo cơ sở dữ liệu chính và thiết lập chứng chỉ số của bạn.
* Sử dụng công cụ GSKCapiCmd để tạo cơ sở dữ liệu chính của bạn. Nó phải là cơ sở dữ liệu khóa loại hệ thống quản lý chứng chỉ (CMS).
* Khi bạn tạo cơ sở dữ liệu khóa, cơ sở dữ liệu đó sẽ được tự động điền các chứng chỉ người ký từ một số tổ chức phát hành chứng chỉ (CA)
* Thêm chứng chỉ cho máy chủ của bạn vào cơ sở dữ liệu khóa của bạn. Máy chủ gửi chứng chỉ này cho các máy khách trong quá trình bắt tay SSL để cung
* cấp xác thực cho máy chủ
* Trích xuất chứng chỉ bạn vừa tạo vào một tệp để bạn có thể phân phối chứng chỉ đó cho các máy tính đang chạy máy khách sẽ thiết lập kết nối SSL đến máy chủ DB2 của bạn.

## Thiết lập cấu hình

**Bước 1:** Điều kiện tiên quyết

* Để SSL hoạt động, cần có chứng chỉ kỹ thuật số. Bạn sẽ cần chứng chỉ số do tổ chức phát hành chứng chỉ (CA) ký hoặc bạn có thể tự ký chứng chỉ số của mình. Nếu cơ sở dữ liệu và máy khách của bạn nằm trong mạng riêng thì chứng chỉ số tự ký sẽ ổn.
* Các bước để định cấu hình SSL bằng chứng chỉ số tự ký và chứng chỉ số do CA ký là giống nhau.

**Bước 2:**  Đưa gskit vào biến môi trường LD\_LIBRARY\_PATH, LIBPATH, SHLIB\_PATH.

PATH=$HOME/sqllib/gskit/bin:$PATH

export PATH

LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:$HOME/sqllib/lib64/gskit

export LD\_LIBRARY\_PATH

LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:$HOME/sqllib/lib32/gskit

export LD\_LIBRARY\_PATH

LIBPATH=$LIBPATH:$HOME/sqllib/lib64/gskit

export LIBPATH

LIBPATH=$LIBPATH:$HOME/sqllib/lib32/gskit

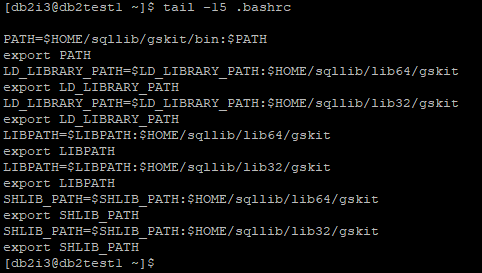
export LIBPATH

SHLIB\_PATH=$SHLIB\_PATH:$HOME/sqllib/lib64/gskit

export SHLIB\_PATH

SHLIB\_PATH=$SHLIB\_PATH:$HOME/sqllib/lib32/gskit

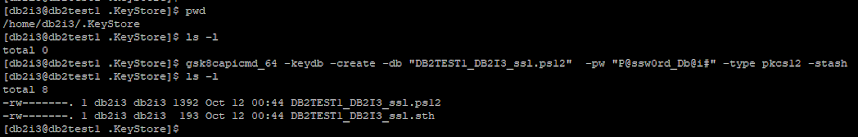
export SHLIB\_PATH



**Bước 3:** Tạo KeyStore và chứng chỉ số tự ký, trích xuất chứng chỉ từ Keystore

Tạo Kho khóa cục bộ kiểu PKCS12 bằng lệnh gsk8capicmd\_64 bên dưới sử dụng tùy chọn stash.

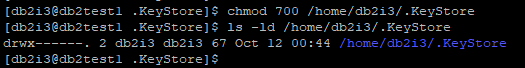
gsk8capicmd\_64 -keydb -create -db "DB2TEST1\_DB2I3\_ssl.ps12" -pw "P@ssw0rd\_Db@i#" -type pkcs12 -stash



**gsk8capicmd\_64**sẽ không cung cấp bất kỳ đầu ra nào khi hoàn thành, nó sẽ tạo 2 tệp trong thư mục làm việc hiện tại của bạn. Cần tạo Keystore với tùy chọn stash, nếu không, đối với mỗi lần truy cập Keystore, bạn cần cung cấp mật khẩu Keystore.

Bạn nên đảm bảo rằng Kho khóa và tệp lưu trữ của bạn được bảo mật, để thực hiện việc này, hãy thay đổi quyền truy cập thư mục Kho khóa của bạn thành 700.

*chmod 700 /home/db2i3/.KeyStore*



Tạo chứng chỉ kỹ thuật số tự ký bằng lệnh gsk8capicmd\_64 dưới đây

*gsk8capicmd\_64 -cert -create -db "DB2TEST1\_DB2I3\_ssl.ps12"     -pw "P@ssw0rd\_Db@i#" -label "DB2TEST1\_DB2I3\_ssl"  -dn "CN=db2test1"*

*[Create self signed digital certificate](https://lh3.googleusercontent.com/-wvnhfhfWNaY/X4NcKT6iKjI/AAAAAAAAGBE/RUDU4j37NjYeUTDMadmAOOp02x04QBpoACLcBGAsYHQ/image.png)*

gsk8capicmd\_64 không cung cấp bất kỳ đầu ra nào khi hoàn thành. -db tùy chọn tên KeyStore cục bộ, -pw cho biết mật khẩu của KeyStore cục bộ, -label cho biết nhãn cho chứng chỉ sẽ được sử dụng trong các bước sau, -db cho biết tên phân biệt X.500 nhận dạng duy nhất chứng chỉ. Tùy chọn in -dn ngoài CN còn có các tùy chọn khác như O, OU, L, ST, C, DC, EMAIL nhưng những tùy chọn này không bắt buộc.

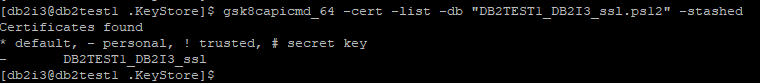
Trong trường hợp chứng chỉ số do CA ký, bạn cần nhập chứng chỉ đó vào cơ sở dữ liệu cục bộ của mình bằng lệnh dưới đây

*gsk8capicmd\_64 -cert -import -db "SSL\_CERTIFICATE.crt" -pw "P@ssw0rd" -target "DB2TEST1\_DB2I3\_ssl.ps12" -target\_stashed*

- db cho biết tên chứng chỉ, -pw cho biết mật khẩu được liên kết với chứng chỉ, -target cho biết Kho khóa cục bộ, -target\_stashed cho biết tệp stash có sẵn để xác thực quyền truy cập vào KeyStore cục bộ.

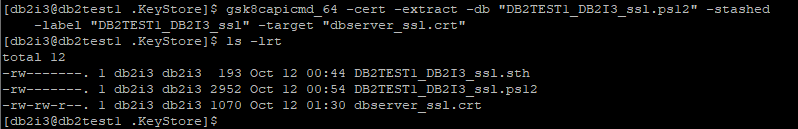
Xác minh Chứng chỉ trong KeyStore bằng lệnh gsk8capicmd\_64 dưới đây

*gsk8capicmd\_64 -cert -list -db "DB2TEST1\_DB2I3\_ssl.ps12" -stashed*

[](https://lh3.googleusercontent.com/-ZgFHu7930bY/X4NiCLlQIqI/AAAAAAAAGBQ/oO0r1zCivg4KieXnhwK4aGYQM-XlUbzWQCLcBGAsYHQ/image.png)

Trích xuất chứng chỉ từ Local KeyStore bằng lệnh gsk8capicmd\_64 dưới đây  để phân phối cho hệ thống khách hàng.

*gsk8capicmd\_64 -cert -extract -db "DB2TEST1\_DB2I3\_ssl.ps12" -stashed -label "DB2TEST1\_DB2I3\_ssl" -target "dbserver\_ssl.crt"*

[](https://lh3.googleusercontent.com/-BP-mQ_07rMc/X4Nkf4o_lHI/AAAAAAAAGBc/ibj8YJ3G2ksembd7dRfTAYwD9gNL067IQCLcBGAsYHQ/image.png)

Khi hoàn thành, nó sẽ tạo ra một tệp được chỉ định với tùy chọn -target, -db cho biết tên KeyStore cục bộ, - stashed cho biết tệp stash có sẵn để xác thực quyền truy cập vào KeyStore cục bộ, -label cho biết nhãn đối với chứng chỉ cần được trích xuất,  -pw cho biết mật khẩu được liên kết với giấy chứng nhận.

**Bước 4:** Cập nhật các tham số của Trình quản lý cơ sở dữ liệu (DBM) và biến đăng ký.

Cập nhật các thông số DBM

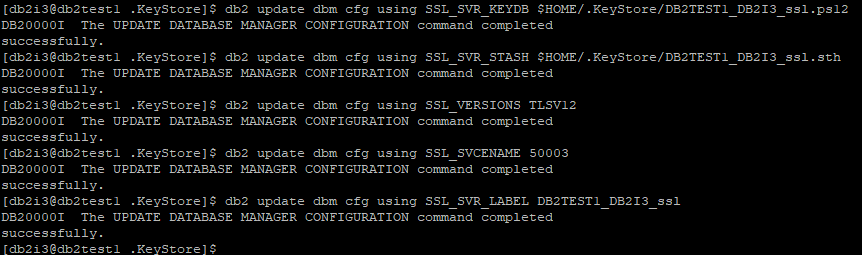
*db2 update dbm cfg using SSL\_SVR\_KEYDB $HOME/.KeyStore/DB2TEST1\_DB2I3\_ssl.ps12*

*db2 update dbm cfg using SSL\_SVR\_STASH $HOME/.KeyStore/DB2TEST1\_DB2I3\_ssl.sth*

*db2 update dbm cfg using SSL\_VERSIONS TLSV12*

*db2 update dbm cfg using SSL\_SVCENAME 50003*

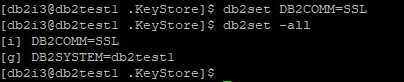
*db2 update dbm cfg using SSL\_SVR\_LABEL DB2TEST1\_DB2I3\_ssl*

[](https://lh3.googleusercontent.com/-Ks6ZDfkytCI/X4NogENOJnI/AAAAAAAAGBo/6mccV80HrNUj6CXiLmsNCkWx_MDjtVk0ACLcBGAsYHQ/image.png)

* SSL\_SVR\_KEYDB spec ify đường dẫn đầy đủ đủ điều kiện của KeyStore cục bộ
* SSL\_SVR\_STASH chỉ định đường dẫn đủ điều kiện của tệp lưu trữ KeyStore cục bộ
* SSL\_VERSIONS chỉ định phiên bản TLS được hỗ trợ. TLSV12 cho biết chỉ TLS 1.2 sẽ được hỗ trợ. Bạn có thể giữ nguyên giá trị này hoặc TLSV1 để chỉ hỗ trợ TLS 1.0 và 1.1. Bạn có thể cập nhật nó lên  TLSV1,TLSV12 để hỗ trợ TLS 1.0,1.1 và 1.2.
* SSL\_SVCENAME chỉ định cổng mà trên đó DB2 sẽ chấp nhận các kết nối TLS.
* SSL\_SVR\_LABEL chỉ định nhãn chứng chỉ trong KeyStore cục bộ sẽ được sử dụng cho kết nối SSL

**Cập nhật biến đăng ký**

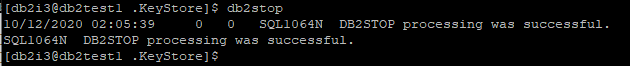
*db2set DB2COMM=SSL*

[](https://lh3.googleusercontent.com/-hVMcDMn5Aqs/X4Nr3e7dKpI/AAAAAAAAGCA/37zUod5wu8Ere073LhDiC1nE_O5DYrongCLcBGAsYHQ/image.png)

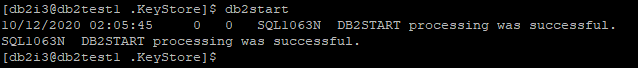
DB2COMM chỉ định giao thức được sử dụng để giao tiếp với các máy khách từ xa. SSL cho biết DB2 sẽ chỉ hỗ trợ giao thức SSL cho giao tiếp máy khách từ xa. Bạn cũng có thể đặt nó thành SSL, TCPIP để hỗ trợ cả SSL, giao thức TCPIP cho giao tiếp máy khách từ xa.

**Bước 5:**Khởi động lại phiên bản DB2.

Dừng Phiên bản DB2 bằng lệnh *db2stop*

[](https://lh3.googleusercontent.com/-_GaDPPG-nGA/X4Nssn7pX9I/AAAAAAAAGCM/Is1rpBXCHS8Y3RjQIUBE6rHk8FqhkVo9QCLcBGAsYHQ/image.png)

Khởi động Phiên bản DB2 bằng lệnh *db2start*

[](https://lh3.googleusercontent.com/-pxd2eZy8hfY/X4NtBOE_8RI/AAAAAAAAGCU/MxwwlmphJ0wVKdzfiba19AxoA1Apm22pwCLcBGAsYHQ/image.png)

**Bước 6:** Định cấu hình Máy khách DB2 để sử dụng SSL cho giao tiếp

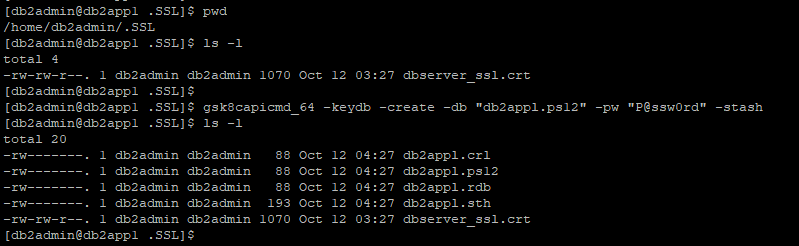
Sao chép tệp chứng chỉ dbserver ssl.crt được tạo ở bước 3 vào hệ thống Máy khách.

DB2 CLP, máy khách Máy chủ dữ liệu IBM hoặc máy khách Thời gian chạy máy chủ dữ liệu IBM

Định cấu hình hồ sơ người dùng để bao gồm cả đường dẫn gskit 32 và 64 bit trong LD\_LIBRARY PATH, LIBPATH , SHLIB\_PATH, hãy làm theo Bước 2.  Nếu gskit trong không có sẵn, hãy cài đặt gskit trước.

Tạo Kho khóa cục bộ kiểu PKCS12 bằng lệnh gsk8capicmd\_64 bên dưới sử dụng tùy chọn stash.

*gsk8capicmd\_64 -keydb -create -db "db2appl.ps12"  -pw "P@ssw0rd" -type pkcs12 -stash*

[](https://lh3.googleusercontent.com/-RvwsKsJ4Auk/X4OOWivSHQI/AAAAAAAAGCk/KTZFl-CDzwYKnihH1hIulNAOibsKqMtswCLcBGAsYHQ/image.png)

Thêm chứng chỉ được trích xuất từ ​​DB Server KeyStore vào KeyStore của hệ thống máy khách. Trước khi thêm chứng chỉ vào hệ thống máy khách, KeyStore đảm bảo bạn đã sao chép chứng chỉ Máy chủ từ máy chủ DB.

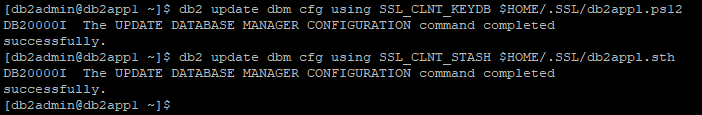
*gsk8capicmd\_64 -cert -add -db db2appl.ps12 "-pw" P @ ssw0rd "-label" DB2TEST1\_DB2I3\_ssl "-tệp" dbserver\_ssl.crt "*

*[Thêm chứng chỉ máy chủ vào Local KeyStore](https://lh3.googleusercontent.com/-DLFE2vhEJ1c/X4OPGl5MA4I/AAAAAAAAGC0/TSkuqcpgL0kH2nsay-EKu8GSSM2yM0GbgCLcBGAsYHQ/image.png)*

Cập nhật các thông số DBM của ứng dụng khách

*db2 update dbm cfg using SSL\_CLNT\_KEYDB $HOME/.SSL/db2appl.ps12*

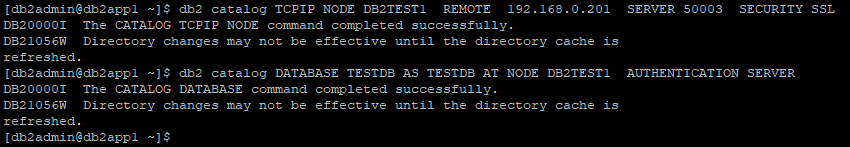
*db2 update dbm cfg using SSL\_CLNT\_STASH $HOME/.SSL/db2appl.sth*

*[](https://lh3.googleusercontent.com/-KkXDqGBFwZY/X41Q1aL7EGI/AAAAAAAAGEc/rhtxpLOmeCkuDZh2RKin-cMLIOAiVnWkgCLcBGAsYHQ/image.png)*

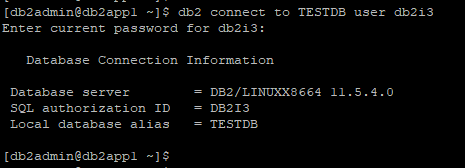
Cơ sở dữ liệu từ xa danh mục trên hệ thống khách hàng

*db2 catalog TCPIP NODE DB2TEST1  REMOTE  192.168.0.201  SERVER 50003  SECURITY SSL*

*db2 catalog DATABASE TESTDB AS TESTDB AT NODE DB2TEST1  AUTHENTICATION SERVER*

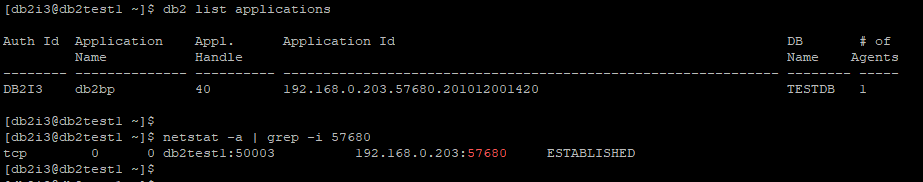
*[](https://lh3.googleusercontent.com/-PEsoGaa5hFw/X4On2ujDDYI/AAAAAAAAGD8/ATHItNKapds7s0c-5MOxNH3gxbDSktSRQCLcBGAsYHQ/image.png)*

Kiểm tra kết nối DB2

[](https://lh3.googleusercontent.com/-fET0-fw8hRc/X4Ojh5VXInI/AAAAAAAAGDQ/RNrPU0lJiQM3voZdgLE2JaVCszSuuxFWwCLcBGAsYHQ/image.png)

Xác minh kết nối là kết nối SSL

Xác định id ứng dụng kết nối máy khách trên máy chủ DB bằng lệnh *ứng dụng danh sách db2*.

[](https://lh3.googleusercontent.com/-p_i7eI4sH9g/X4OmRp0IUcI/AAAAAAAAGDw/H6sUkkqjhyU12IYY5CI4LfDxrMs0wEaFQCLcBGAsYHQ/image.png)

# IBM Security Guardium: giải pháp bảo mật dữ liệu toàn diện

## Giải pháp giúp phân tích, bảo vệ và đáp ứng để bảo mật dữ liệu toàn diện

Những điểm nổi bật của giải pháp IBM Security Guardium:

* Áp dụng cách tiếp cận chủ động, toàn diện để bảo vệ dữ liệu quan trọng trên tất cả các loại nền tảng, bao gồm cơ sở dữ liệu chính, kho dữ liệu, nền tảng dữ liệu lớn, môi trường đám mây, hệ thống tệp và hơn thế nữa.
* Tổng chi phí thấp hơn bằng cách tự động phát hiện dữ liệu nhạy cảm, phát hiện rủi ro và thực hiện hành động.
* Bảo vệ dữ liệu nhạy cảm chống lại các mối đe dọa với mã hóa, mặt nạ, chuyển hướng, giám sát hoạt động, chặn động, cảnh báo và cách ly.
* Tận dụng tự động tuân thủ dữ liệu để có được báo cáo đúng đến đúng người vào đúng thời điểm.
* Thích ứng với những thay đổi trong bối cảnh CNTT và hỗ trợ hành trình bảo vệ dữ liệu đầy đủ.

Ngày nay, các vi phạm bảo mật dữ liệu phổ biến hơn bao giờ hết và có tác động mạnh hơn. Các nghiên cứu toàn cầu cho thấy tổng chi phí trung bình của một vụ vi phạm dữ liệu hiện là 4 triệu USD. Hơn nữa, việc mất bí mật thương mại, thiết kế sản phẩm hoặc sở hữu trí tuệ khác có thể gây ra sự hủy hoại tài chính cho một tổ chức. Do giá trị của nó, dữ liệu quan trọng và nhạy cảm là cốt lõi của các tương tác kinh doanh, điều này cũng khiến nó trở thành mục tiêu hấp dẫn để tấn công.

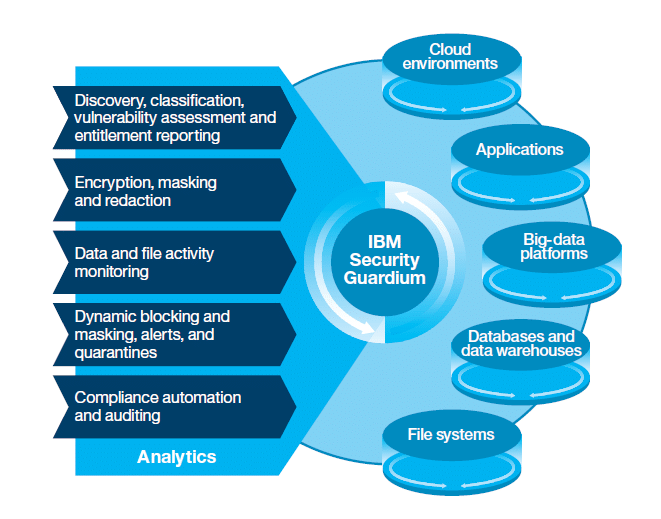
Theo truyền thống, các tổ chức đã tập trung vào hệ thống phòng thủ trên chu vi (perimeter) để bảo vệ thông tin quan trọng của họ. Nhưng các công cụ truyền thống, chẳng hạn như phần mềm chống vi-rút và tường lửa, không được trang bị cho các mối đe dọa tiên tiến của ngày hôm nay, mà nhiều lần đến từ chính bên trong tổ chức. Thêm vào đó, dữ liệu liên tục phát triển, thay đổi và di chuyển, vì vậy các biện pháp bảo vệ dữ liệu cũng phải có khả năng thích ứng để tuân theo dữ liệu. Số lượng người dùng, ứng dụng và hệ thống ngày càng tăng cần truy cập ngay vào các loại dữ liệu nhạy cảm khác nhau, cư trú hoặc sao chép vào cơ sở dữ liệu, kho dữ liệu, chia sẻ tệp, nền tảng dữ liệu lớn, môi trường đám mây và hơn thế nữa. Theo dõi ai có quyền truy cập vào dữ liệu động, phân tán và khác biệt này và ai đang chia sẻ dữ liệu đó (và với ai), có vẻ như là một nhiệm vụ không thể vượt qua.

IBM Security Guardium được thiết kế để bảo vệ dữ liệu quan trọng, bất cứ nơi nào nó cư trú. Nền tảng bảo vệ dữ liệu toàn diện này trao quyền cho các nhóm bảo mật tự động phân tích những gì đang xảy ra trên môi trường dữ liệu để giúp giảm thiểu rủi ro, bảo vệ dữ liệu nhạy cảm khỏi các mối đe dọa bên trong và bên ngoài và thích ứng liền mạch với các thay đổi ảnh hưởng đến bảo mật và tuân thủ dữ liệu.

## Bảo mật dữ liệu toàn diện

Guardium cung cấp một cách tiếp cận toàn diện để bảo vệ dữ liệu quan trọng, điều này là rất cần thiết cho sự thành công và sự sống còn của doanh nghiệp. Tận dụng giao diện người dùng đồ họa từ đầu đến cuối, các nhóm bảo mật có thể xác định và khắc phục rủi ro đối với dữ liệu nhạy cảm, cho dù dữ liệu đang chuyển động hay đang nghỉ. Và cách tiếp cận thống nhất này mở rộng ra một loạt các kho dữ liệu có cấu trúc và không cấu trúc, bao gồm cơ sở dữ liệu, kho dữ liệu, Hadoop, NoSQL, hệ thống trong bộ nhớ, chia sẻ tệp, v.v.

Trên thực tế, Guardium có khả năng linh hoạt để đáp ứng một loạt các yêu cầu bảo mật và bảo vệ dữ liệu, từ tuân thủ cơ bản đến bảo vệ dữ liệu toàn diện, theo cách hiệu quả, có thể mở rộng. Giải pháp nhiều lớp bao gồm phân tích mối đe dọa dữ liệu tự động, bảo vệ dữ liệu động và khả năng hiển thị của doanh nghiệp để thích ứng với những thay đổi trong môi trường dữ liệu nhạy cảm.



## Phân tích các mối đe dọa đối với dữ liệu nhạy cảm

Để bảo vệ dữ liệu hiệu quả, các tổ chức cần phải hiểu chính xác những gì họ cần bảo vệ và sau đó bảo vệ triệt để nó.

Guardium cho phép các nhóm bảo mật:

* Khám phá và phân loại dữ liệu nhạy cảm và quyền lợi – và phát hiện ra rủi ro tuân thủ – một cách tự động.
* Biết ai đang truy cập dữ liệu, phát hiện sự bất thường và ngăn chặn mất dữ liệu.
* Phân tích nhanh các mô hình sử dụng dữ liệu để phát hiện và khắc phục rủi ro.
* Hỗ trợ phân tích với các phân tích nâng cao tự động và học máy để phát hiện và ngăn chặn hành vi bất thường và rủi ro.
* Tận dụng các phân tích phát hiện mối đe dọa chuyên biệt để phát hiện và ngăn chặn vi phạm sớm, chẳng hạn như bằng cách tìm và cảnh báo về việc tiêm SQL hoặc các phương pháp lưu trữ độc hại.
* Cung cấp bảng điều khiển để giúp các bên liên quan thấy được sự bảo mật dữ liệu và/hoặc trạng thái tuân thủ và tiến trình theo thời gian, để hiểu rõ hơn về cách sáng kiến tăng giá trị cho doanh nghiệp.

Guardium giúp các nhóm bảo mật tự động khám phá và phân loại thông tin nhạy cảm – từ một giao diện người dùng đồ họa (graphical user interface) dễ sử dụng. Sử dụng một loạt các bước, nhân viên bảo mật có thể khám phá tất cả các nguồn dữ liệu chứa thông tin nhạy cảm, bao gồm cả cơ sở dữ liệu không được mã hóa, sau đó sử dụng nhãn phân loại có thể tùy chỉnh và khả năng quản lý quyền để tự động hóa việc thực thi chính sách bảo mật. Phát hiện dữ liệu nhạy cảm cũng có thể được lên kế hoạch để thực hiện thường xuyên để giúp ngăn chặn việc các máy chủ giả mạo và đảm bảo rằng không có thông tin quan trọng nào bị bỏ lỡ.

Để giúp thực thi các chính sách và bảo vệ dữ liệu nhạy cảm, Guardium có thể liên tục theo dõi ai đang truy cập (hoặc cố gắng truy cập) dữ liệu nhạy cảm trong thời gian thực. Vượt xa hơn việc giám sát dữ liệu truyền thống, Guardium có khả năng phát hiện ngoại lệ với trí thông minh gia tăng để phân tích và hiểu rủi ro dựa trên những thay đổi trong hành vi. Nó sử dụng thuật toán học máy tiên tiến để phát hiện truy cập dữ liệu bất thường dựa trên thông tin theo ngữ cảnh chi tiết – “ai, người nào, ở đâu, khi nào và như thế nào” mỗi lần truy cập dữ liệu. Với một quá trình học tập thích ứng, nó so sánh các mô hình hoạt động bình thường mới với các hoạt động mới khi chúng tích lũy. Giao diện người dùng trực quan giúp xác định chính xác sự bất thường, vì vậy các quản trị viên có thể đi sâu vào để điều tra nguyên nhân gốc rễ.

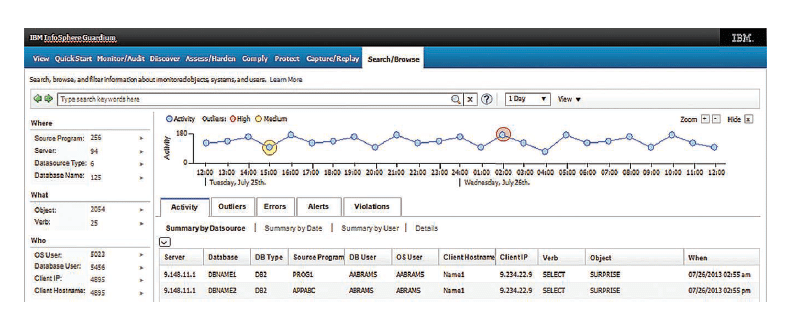
Ngoài khả năng truy sâu, Guardium cho phép nhân viên bảo mật tìm kiếm nhanh các báo cáo audit và các mục khác trong giao diện, cũng như chạy các tìm kiếm nhanh, toàn doanh nghiệp trên chính dữ liệu. Ở đó, bạn không cần phải hiểu các cấu trúc liên kết, tổng hợp hoặc cân bằng tải cơ bản. Các yêu cầu tìm kiếm có thể giúp trích xuất thông tin chi tiết từ hoạt động truy cập dữ liệu cụ thể, cho dù tập trung vào các nguồn dữ liệu cụ thể, người dùng hoặc ngày. Một bảng điều khiển điều tra mới cũng có thể giúp tiết lộ các mẫu, sự bất thường và các mối quan hệ trên dữ liệu, giúp thu hẹp phạm vi với các chế độ xem mặc định thực tiễn tốt nhất. Ngoài ra còn có một công cụ Connection Profiling báo cáo về tất cả các kết nối đã thử với một nguồn dữ liệu cụ thể.

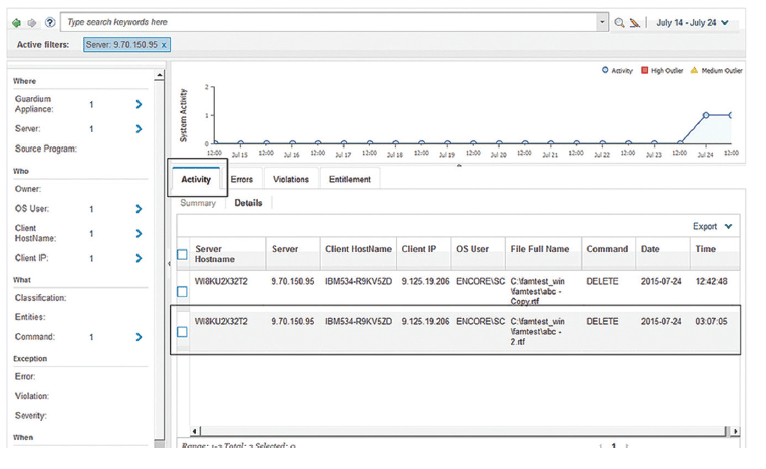
## Bảo vệ dữ liệu nhạy cảm

Sự leo thang của các mối đe doạ liên quan đến dự liệu nhạy cảm và các nhiệm vụ tuân thủ ngày càng tăng đang thúc đẩy các tổ chức suy nghĩ lại về các chiến lược bảo vệ dữ liệu của họ. Guardium cho phép các nhóm bảo mật:

* Bảo vệ doanh nghiệp khỏi rủi ro tài chính với việc tuân thủ dữ liệu tự động và khả năng audit mở rộng.
* Kiểm soát dữ liệu quan trọng thông qua mã hóa, mặt nạ, điều chỉnh, chặn động, cảnh báo và kiểm dịch.
* Sử dụng theo dõi và chặn hoạt động trong thời gian thực để giúp ngăn chặn truy cập tệp và dữ liệu bên trong và bên ngoài.

Guardium giúp nắm bắt và kiểm tra tất cả lưu lượng dữ liệu nhạy cảm, bao gồm quyền truy cập cục bộ của người dùng đặc quyền, với quy trình kiểm tra an toàn, chống giả mạo. Trên thực tế, nó cung cấp một kho lưu trữ kiểm toán duy nhất, tập trung và chuẩn hóa để báo cáo tuân thủ toàn doanh nghiệp, tối ưu hóa hiệu suất, điều tra và pháp y (investigations and forensics). Các tổ chức có thể tự động hóa toàn bộ quy trình kiểm toán tuân thủ dữ liệu – bao gồm phân phối báo cáo cho các đội giám sát, sign-off và leo thang – với các báo cáo được cấu hình sẵn cho Sarbanes-Oxley (SOX), Tiêu chuẩn bảo mật dữ liệu công nghiệp thẻ thanh toán (Payment Card Industry Data Security Standard – PCI DSS) và bảo mật dữ liệu.





Hơn nữa, Guardium trao quyền cho các nhóm bảo mật để bảo vệ dữ liệu nhạy cảm khỏi các mối đe dọa bên trong và bên ngoài bằng mã hóa dựa trên tệp, cơ sở dữ liệu và đánh giá lỗ hổng dữ liệu lớn, khả năng che giấu và xử lý dữ liệu tĩnh. Nó cũng hỗ trợ mã hóa và che giấu dữ liệu thời gian thực, cũng như chặn, cảnh báo và cách ly những người dùng đáng ngờ. Trên thực tế, nó có thể hạn chế quyền truy cập từ các tác nhân giả mạo vào dữ liệu nhạy cảm trên hầu hết các nguồn, bao gồm môi trường đám mây, nền tảng dữ liệu lớn (big-data) và hệ thống tệp (file systems).

Guardium cũng giúp thực thi sự phân biệt nhiệm vụ bằng cách liên tục theo dõi tất cả các hoạt động dữ liệu nhạy cảm, bao gồm cả giám sát truy cập hệ thống tệp theo thời gian thực. Nó cho phép các tổ chức phát hiện, ghi lại và chặn hoạt động trái phép và đáng ngờ của người dùng đặc quyền. Ví dụ: Guardium có thể phát hiện một bản sao hàng loạt các tệp hoặc thư mục nhạy cảm, phát hiện đột biến trong hoạt động truy cập tệp của một quản trị viên cụ thể, tạo cảnh báo về quyền truy cập không chính xác, chặn truy cập vào các tài liệu nhạy cảm nhất và tạo báo cáo tùy chỉnh cho mọi hoạt động. Guardium cũng có thể giúp khám phá các phơi nhiễm với cơ sở dữ liệu và cơ sở hạ tầng dữ liệu lớn để đảm bảo rằng nền tảng cho dữ liệu được làm bảo vệ.

## Thích nghi với sự thay đổi

Cơ sở hạ tầng dữ liệu luôn thay đổi và phát triển – tạo ra các thách thức và tốn kém để theo kịp các lỗ hổng bảo mật mới nổi và luôn thay đổi. Guardium cung cấp cho các tổ chức sức mạnh để:

* Hỗ trợ các công nghệ dữ liệu truyền thống và đột phá, như Hadoop, NoQuery và đám mây.
* Dễ dàng mở rộng kiến trúc bảo vệ dữ liệu, phát triển từ tuân thủ quy định đến bảo vệ dữ liệu toàn diện.
* Giảm chi phí và cải thiện kết quả bằng cách sử dụng một cơ sở hạ tầng bảo vệ dữ liệu duy nhất, một hệ thống tự động cân bằng tải trên toàn bộ môi trường dữ liệu.

Guardium cho phép các tổ chức thích ứng với những thay đổi trong môi trường dữ liệu, mở rộng bảo vệ dữ liệu để giải quyết người dùng, nền tảng và loại dữ liệu mới. Nó cũng cung cấp một nền tảng đơn giản hóa việc quản trị bảo mật dữ liệu bằng cách kết hợp cách thức hoạt động của CNTT, thông qua tự động hóa, tập trung hóa và tích hợp. Trọng tâm nền tảng rộng bao gồm hỗ trợ cho cơ sở dữ liệu truyền thống, môi trường đám mây, hệ thống dựa trên Hadoop, NoQuery, hệ thống trong bộ nhớ và hệ thống tệp. Guardium cung cấp kiểm soát nhanh có thể được triển khai cho các yêu cầu tuân thủ cụ thể và sau đó dễ dàng mở rộng để cung cấp bảo vệ bổ sung khi nhu cầu kinh doanh phát triển.

Không giống như một giải pháp điểm, Guardium hỗ trợ tích hợp với các giải pháp bảo mật hàng đầu trong ngành, các tiêu chuẩn dễ bị tổn thương, các ứng dụng và hơn thế nữa. Guardium cũng cung cấp tích hợp tốt nhất với các giải pháp Bảo mật của IBM, như IBM QRadar® SIEM, để bảo vệ dữ liệu chủ động. Guardium gửi các sự kiện và thông tin phân loại / phát hiện cơ sở dữ liệu của mình tới QRadar SIEM, cho phép tương quan hiệu quả hơn với hoạt động đe dọa. Ngoài ra, Guardium có thể nhận thông báo trạng thái và cảnh báo từ QRadar SIEM để giúp bảo vệ chống lại các nguồn IP lừa đảo, người dùng lừa đảo và các lỗ hổng mới, cho dù trong các ứng dụng, hệ điều hành hoặc các nguồn dữ liệu khác. Ví dụ, tích hợp Guardium và QRadar có thể giúp các tổ chức bảo vệ chống lại các cuộc tấn công tiềm năng thông qua các ứng dụng; phát hiện các cuộc tấn công cơ sở dữ liệu (chẳng hạn như thông qua SQL injection) và chặn chúng trước khi dữ liệu có thể được trích xuất; và xác định các lỗ hổng ở lớp ứng dụng để khắc phục lỗi.

## Guardium mang tới giá trị cho một loạt các ngành công nghiệp

* Một công ty bảo hiểm lớn hiện có thể quản lý bảo mật cho khoảng 1.000 cơ sở dữ liệu chỉ với một nhân viên full-time.
* Một công ty tiện ích lớn đã đạt được lợi tức đầu tư (ROI) 55% trong vòng chưa đầy một năm, giúp đảm bảo tuân thủ SOX và PCI cho 4,5 triệu tài khoản.
* Một ngân hàng toàn cầu có thể giám sát hơn 5.000 nguồn dữ liệu, bao gồm các giao dịch dữ liệu lớn, trong thời gian thực mà không ảnh hưởng đến hiệu suất của các ứng dụng quan trọng.
* Một công ty viễn thông quốc tế hiện có thể giám sát và phản hồi tập trung trong thời gian thực đối với hoạt động truy cập dữ liệu trên hàng ngàn cơ sở dữ liệu phân tán tại 16 trung tâm dữ liệu trên toàn thế giới.
* Một nhà sản xuất ô tô có thể giám sát và kiểm toán 500 cơ sở dữ liệu sản xuất để giúp tăng cường bảo mật, đồng thời giảm 90% yêu cầu của nhân viên an ninh.

## Về giải pháp của IBM Security

IBM Security cung cấp một trong những danh mục đầu tư tích hợp và tiên tiến nhất về các sản phẩm và dịch vụ bảo mật doanh nghiệp. Danh mục đầu tư, được hỗ trợ bởi nghiên cứu và phát triển IBM X-Force® nổi tiếng thế giới, cung cấp thông tin bảo mật để giúp các tổ chức bảo vệ toàn diện con người, cơ sở hạ tầng, dữ liệu và ứng dụng của họ, cung cấp các giải pháp để nhận dạng và quản lý truy cập, bảo mật cơ sở dữ liệu, phát triển ứng dụng, quản lý rủi ro , quản lý điểm cuối, an ninh mạng và nhiều hơn nữa. Những giải pháp này cho phép các tổ chức quản lý hiệu quả rủi ro và triển khai bảo mật tích hợp cho thiết bị di động, đám mây, phương tiện truyền thông xã hội và các kiến trúc kinh doanh doanh nghiệp khác. IBM vận hành một trong những tổ chức nghiên cứu, phát triển và bảo mật lớn nhất thế giới, theo dõi 15 tỷ sự kiện bảo mật mỗi ngày tại hơn 130 quốc gia và nắm giữ hơn 3.000 bằng sáng chế bảo mật.

# Mã hóa dữ liệu dùng hệ thống file mã hóa AIX (AIX EFS)

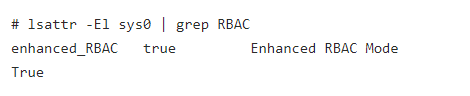
## Giới thiệu về hệ thống file mã hóa AIX

### Khái niệm

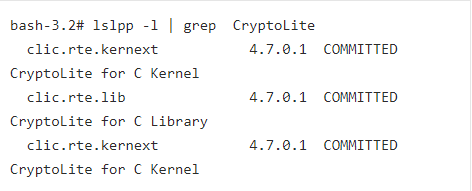
Hệ thống tệp được mã hóa (EFS) là mã hóa cấp hệ thống tệp J2 thông qua các kho khóa riêng lẻ. Điều này cho phép mã hóa tệp để bảo vệ dữ liệu bí mật khỏi những kẻ tấn công có quyền truy cập vật lý vào máy tính.

### Điều kiện tiên quyết

* RBAC phải được kích hoạt. Nên để mặc định trên AIX 6.1. Nếu không sử dụng chdev để kích hoạt nó.



* CryptoLite cần được cài đặt, xác minh bằng lệnh dưới đây:



* Cho phép hệ thống sử dụng EFS:

+ EFS là một phần của hệ điều hành AIX® cơ bản. Để bật EFS, root (hoặc bất kỳ người dùng nào có ủy quyền RBAC **aix.security.efs**, hãy xem các tác nhân [EFS](https://www.ibm.com/docs/en/ssw_aix_72/security/efs_actors.htm) để biết thêm thông tin) phải sử dụng lệnh **efsenable** để kích hoạt EFS và tạo môi trường EFS. Đây là một kích hoạt hệ thống một lần. Sau khi EFS được bật, khi người dùng đăng nhập, khóa và kho khóa của nó được tạo và bảo vệ hoặc mã hóa một cách thầm lặng bằng mật khẩu đăng nhập của người dùng. Các khóa người dùng sau đó được sử dụng một cách dễ dàng bởi hệ thống tệp J2 khi mã hóa hoặc giải mã các tệp EFS. Mọi tệp EFS đều được bảo vệ bằng khóa tệp duy nhất của riêng nó và khóa tệp này lần lượt được bảo vệ hoặc mã hóa với chủ sở hữu tệp hoặc khóa nhóm tùy thuộc vào quyền đối với tệp.

+Theo mặc định, Hệ thống tệp J2 không được bật EFS. Khi nó được kích hoạt EFS, Hệ thống tệp J2 quản lý minh bạch mã hóa và giải mã trong nhân cho các yêu cầu đọc và ghi. Các lệnh quản trị người dùng và nhóm (chẳng hạn như **mkgroup**, **chuser** và **chgroup**) quản lý minh bạch kho khóa của người dùng và nhóm.

+Các lệnh EFS sau được cung cấp để cho phép người dùng quản lý khóa và mã hóa tệp của họ:

**Efskeymgr:** Quản lý và quản lý các khóa

**Efsmgr:** Quản lý mã hóa tệp / thư mục / hệ thống tệp

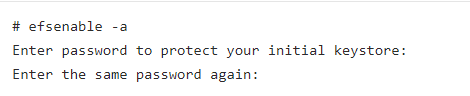
## Vai trò

Hệ thống tệp được mã hóa cho phép người dùng cá nhân trên hệ thống mã hóa dữ liệu của họ trên hệ thống tệp J2 thông qua các kho lưu trữ khóa cá nhân của họ.

Một khóa được liên kết với mỗi người dùng. Các khóa này được lưu trữ trong kho khóa được bảo vệ bằng mật mã và khi đăng nhập thành công, các khóa của người dùng sẽ được tải vào hạt nhân và được liên kết với thông tin xác thực quy trình. Sau đó, khi quy trình cần mở tệp được bảo vệ bằng EFS, các thông tin đăng nhập này sẽ được kiểm tra và nếu tìm thấy khóa phù hợp với bảo vệ tệp, quy trình có thể giải mã khóa tệp và do đó nội dung tệp. Quản lý khóa dựa trên nhóm cũng được hỗ trợ.

## Thiết lập

### Bật EFS trên hệ thống

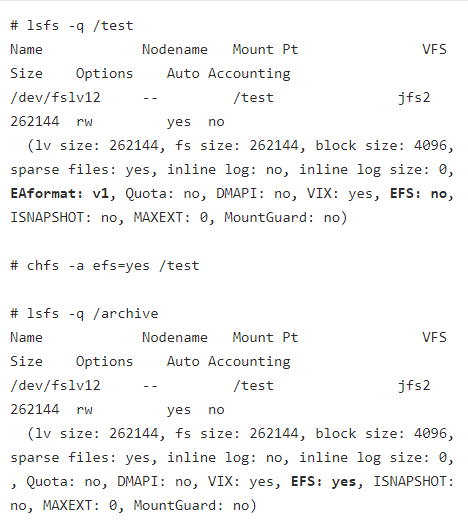


Nếu mật khẩu của bạn cho EFS trùng với mật khẩu đăng nhập của bạn, phần mở rộng nhân EFS sẽ được tải tự động vào nhân. Vì vậy, bạn sẽ có thể truy cập các tệp được mã hóa mà không cần phải cung cấp mật khẩu.  
Nếu không thì `**efskeymgr -o ksh**` sẽ không được thực thi để tải khóa.

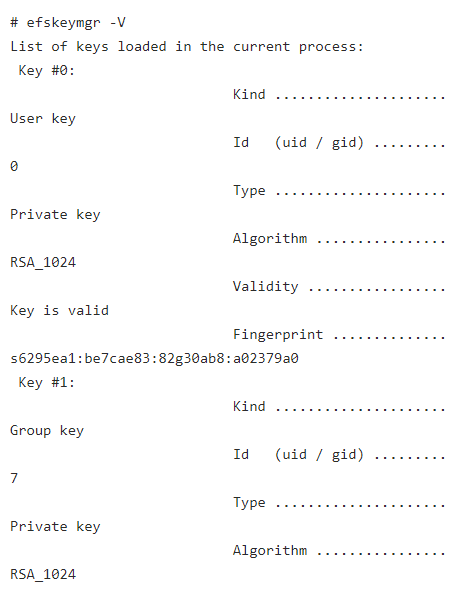
### Kích hoạt thuộc tính V2

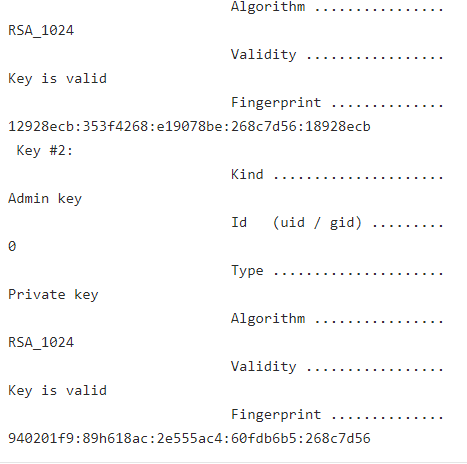
Để có khả năng mã hóa tệp, hệ thống tệp sẽ lưu giữ tệp này cần được bật EFS (efs = yes) và phải kích hoạt Thuộc tính mở rộng V2.

Điều này có thể được xác minh bằng lsfs –q



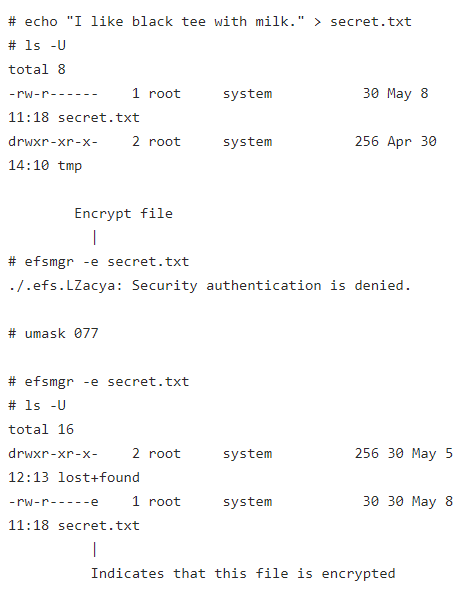
Xem các khóa được liên kết với nhân hiện tại:



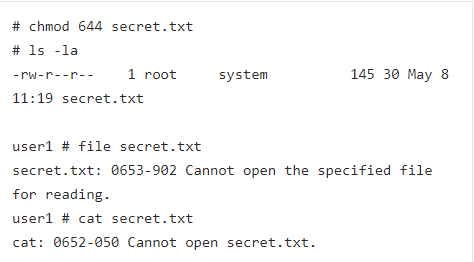


### Mã hóa file

Chúng ta sẽ tạo một tệp và mã hóa nó:

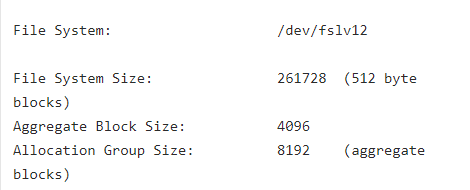


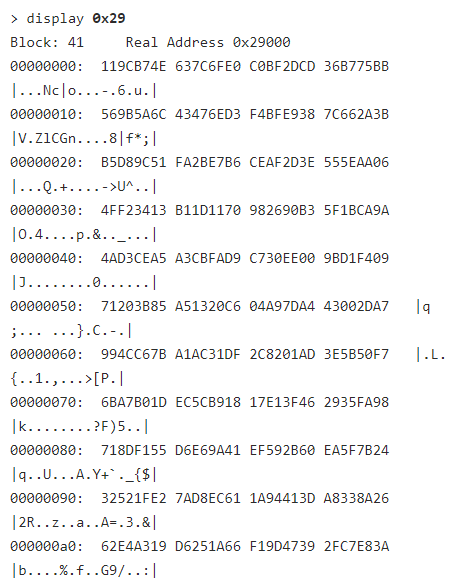
Bây giờ chúng ta đặt quyền của tệp thành 644 và cố gắng đọc tệp với tư cách người dùng khác.



Khi root, chúng ta sẽ liệt kê số inode của tệp, lấy con trỏ khối và đọc trực tiếp từ hệ thống tệp bằng cách sử dụng **fsdb** để xem tệp được lưu trữ có được mã hóa hay không.

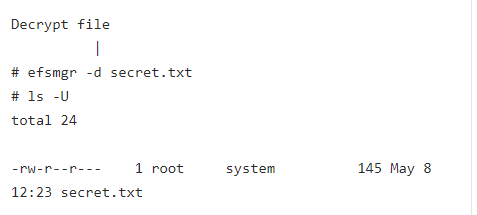






Chúng ta thấy tệp đĩa đã được mã hóa.

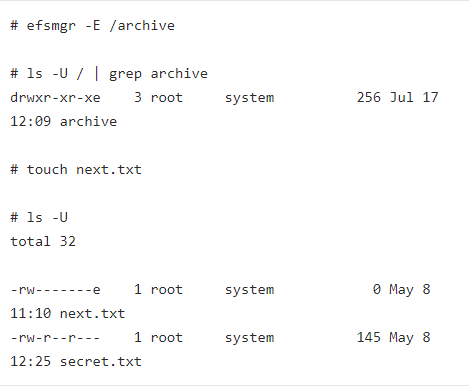
### Giải mã một tệp tin



### Kế thừa mã hóa

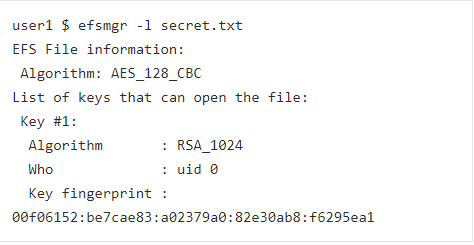
Nếu bạn bật Kế thừa Mã hóa trên một thư mục, tất cả các tệp mới được tạo trong thư mục đó sẽ được mã hóa tự động.

Bật sử dụng kế thừa mã hóa:

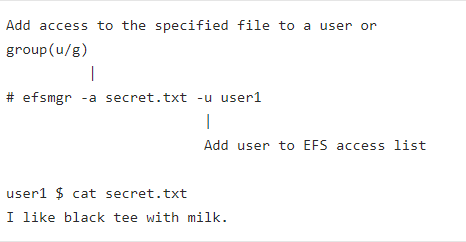


### Cấp quyền truy cập cho người dùng khác

Giả sử chúng ta có user1 và muốn ai có quyền truy cập EFS vào tệp:



Ủy quyền cho người dùng sử dụng:



# DB2 NATIVE ENCRYTION

## Giới thiệu về DB2 native encrytion

### Một số khái niệm về db2 native encrytion

+Mã hóa gốc Db2® sử dụng cách tiếp cận hai tầng để mã hóa dữ liệu. Dữ liệu được mã hóa bằng Khóa mã hóa dữ liệu (DEK), lần lượt được mã hóa bằng Khóa chính (MK). DEK được mã hóa được lưu trữ cùng với dữ liệu trong khi MK được lưu trữ trong kho khóa bên ngoài Db2.

Mã hóa gốc Db2 đảm bảo rằng DEK không bao giờ bị lộ ra bên ngoài cơ sở dữ liệu được mã hóa, nhật ký giao dịch hoặc tệp sao lưu. Không có giao diện nào được cung cấp để truy cập DEK ở dạng văn bản rõ ràng hoặc dạng mã hóa. Vì MK được lưu trữ ở một vị trí khác với dữ liệu được mã hóa, khả năng DEK được mã hóa được tiếp xúc đồng thời với MK được sử dụng để mã hóa là rất khó xảy ra. Vì nguy cơ DEK bị lộ là cực kỳ thấp nên nhu cầu xoay nó là không đáng kể. Việc xoay MK, được sử dụng để bảo vệ DEK, có thể được thực hiện một cách hiệu quả mà không cần phải giải mã và mã hóa lại dữ liệu

+ Db2 mã hóa dữ liệu bằng khóa mã hóa dữ liệu (DEK) trước khi dữ liệu được ghi vào đĩa. DEK được lưu trữ, mã hóa bằng khóa chính (MK), trong cơ sở dữ liệu hoặc hình ảnh sao lưu. Bản thân DEK được tạo bởi Db2 khi cần thiết, chẳng hạn như khi một cơ sở dữ liệu được mã hóa hoặc bản sao lưu cơ sở dữ liệu được mã hóa được tạo. Một DEK duy nhất tồn tại cho mỗi cơ sở dữ liệu được mã hóa và cho mỗi bản sao lưu được mã hóa.

+ Khóa chính (MK) là khóa mã hóa được sử dụng để mã hóa khóa mã hóa dữ liệu (DEK). Mỗi cơ sở dữ liệu được mã hóa được liên kết với một khóa chính tại một thời điểm. Trừ khi được hướng dẫn khác, Db2 tự động tạo ra một MK trong các hoạt động này:

* Tạo cơ sở dữ liệu
* Xoay phím chính
* Khôi phục vào cơ sở dữ liệu mới

Các khóa chính được xác định bằng một nhãn mà Db2 sử dụng để nhận dạng duy nhất từng khóa chính. Theo mặc định, Db2 tạo nhãn cho mọi MK mới được tạo. Bạn có thể ghi đè hành vi này bằng cách cung cấp một nhãn cụ thể cho một MK cụ thể. Các lý do để tạo MK với một nhãn cụ thể bao gồm:

* Theo dõi các nhãn MK và các khóa tương ứng của chúng để khôi phục ngoại vi mà không cần có toàn bộ kho khóa trên trang web dự phòng
* có một cặp HADR yêu cầu các khóa được đồng bộ hóa
* mã hóa bản sao lưu cho cơ sở dữ liệu không được mã hóa

+ Khóa chính được lưu trữ trong kho khóa. Kho khóa có thể là tệp được truy cập trực tiếp bởi Db2 (cục bộ) hoặc kho khóa của bên thứ ba mà Db2 giao tiếp qua mạng (tập trung).

### Vai trò

Mã hóa gốc Db2 cung cấp khả năng mã hóa tích hợp để bảo vệ các hình ảnh sao lưu cơ sở dữ liệu và các tệp cơ sở dữ liệu quan trọng khỏi bị truy cập không phù hợp khi chúng đang ở yên trên phương tiện lưu trữ bên ngoài.

Mã hóa là một thành phần quan trọng trong việc bảo vệ dữ liệu ngoại tuyến. Nhiều quy định của chính phủ và tiêu chuẩn công nghiệp yêu cầu sử dụng nó.

Các tính năng mã hóa gốc Db2:

* triển khai đơn giản
* không yêu cầu các thay đổi đối với lược đồ dữ liệu hoặc các ứng dụng cơ sở dữ liệu
* sử dụng miễn phí trên tất cả các nền tảng và cấu hình Db2 được hỗ trợ.

Các khả năng mã hóa được sử dụng bởi Db2 được chứng nhận FIPS 140-2 và sử dụng các thuật toán mật mã tuân thủ NIST SP 800-131A. Db2 cũng tự động phát hiện và sử dụng mọi khả năng tăng tốc phần cứng cơ bản của CPU để mã hóa khi có sẵn.

Khi bạn mã hóa cơ sở dữ liệu, mã hóa gốc Db2 bảo vệ tất cả các tệp chứa dữ liệu của bạn, chẳng hạn như:

* Tất cả các không gian bảng (cả do hệ thống xác định và do người dùng xác định)
* Tất cả các loại dữ liệu trong một không gian bảng (bao gồm các kiểu dữ liệu LOB và XML)
* Tất cả nhật ký giao dịch, bao gồm cả các tệp nhật ký đã lưu trữ
* TẢI dữ liệu SAO CHÉP
* TẢI tập tin dàn dựng

Mã hóa gốc Db2 cũng có thể được sử dụng để mã hóa các bản sao lưu cơ sở dữ liệu, ngay cả khi cơ sở dữ liệu nguồn không được mã hóa.

### Thiết lập cấu hình

**Bước 1:** Tạo một Keystore

./sqllib/gskit/bin/gsk8capicmd\_64

-keydb

-create

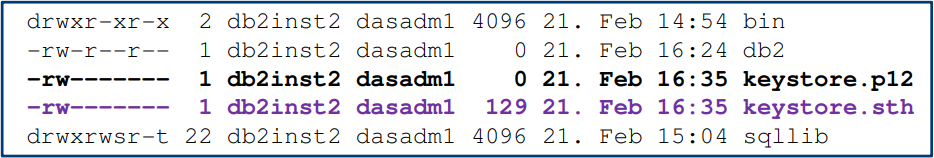
-db keystore.p12

-pw Vankhiemvu\_At150529

-strong

-type pkcs12

-stash



**Bước 2**: Điều chỉnh lại cấu hình DB2

db2 update dbm cfg

using keystore\_type pkcs12

keystore\_location /home/db2inst2/keystore.p12

**Bước 3:** Tạo Master key

Master key có thể được tạo tự động tại thời điểm khi ta tạo cơ sở dữ liệu.

Nếu tạo theo cách thủ công thì cần được chỉ định để tạo các bản sao lưu mã hóa của các cơ sở dữ liệu không được mã hóa.

./sqllib/gskit/bin/gsk8capicmd\_64

-secretkey

-add

-db keystore.p12

-stashed

-label mylabel

-file mysecretfile

**Bước 4:** Tạo File bí mật

File cần phải:

* Là file nhị phân
* Có kích thước bằng kích thước của khóa mã hóa
* Cần phải xóa bỏ sau khi sử dụng

dd if=/dev/urandom of=mysecretfile bs=32 count=1

chmod 600 mysecretfile

Stashfile:

* Lưu mật khẩu của Keystore theo cách khó đoán nhât.
* Có thể tự động điền mật khẩu.
* Stashfile (.sth) chỉ có thể được truy cập bởi chủ sở hữu.
* Cơ sở dữ liệu được mã hóa chỉ được truy cập bởi mật khẩu của Keystore

db2start open keystore using

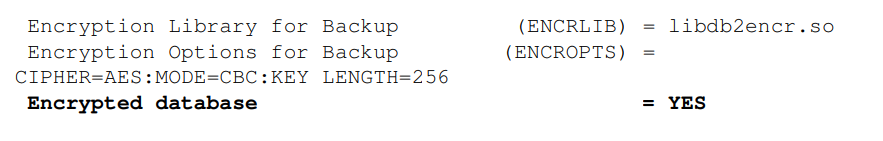
db2start open keystore passarg filename.

**Bước 5:** Tạo Cơ sở dữ liệu

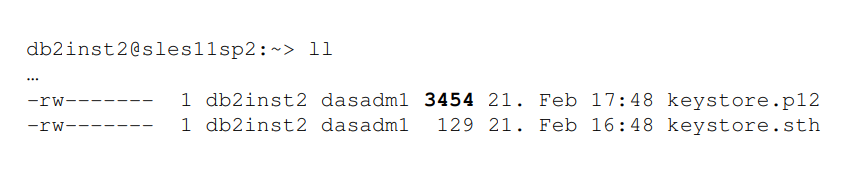
1. Tiêu chuẩn (Standard):

db2 create db dbenc encrypt

* Trong DB CFG:



* Khóa chính được tự động thêm vào keydb



1. Cá nhân (Individualy):

Nhãn xác định khóa chính trong Keystore.

Kĩ thuật mã hóa có thể được chọn một cách độc lập.

db2 create db dbenc encrypt

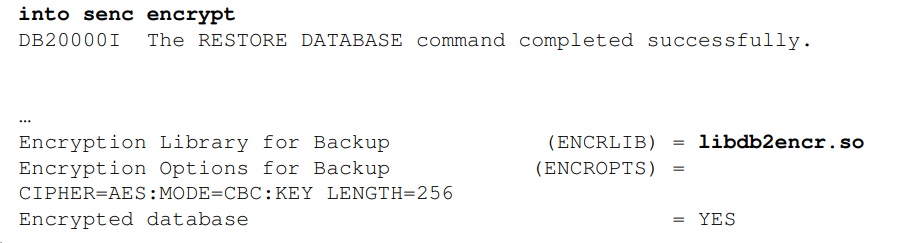
cipher aes

key length 128

master key label mklabel

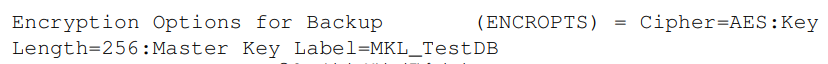
**Bước 6:** Mã hóa tất cả các cơ sở dữ liệu hiện có :

db2 restore db sample from /diskb/backup taken at 20150221181051



**Backup Encryption**: Mã hóa bản sao cơ sở dữ liệu khi chưa được mã hóa .

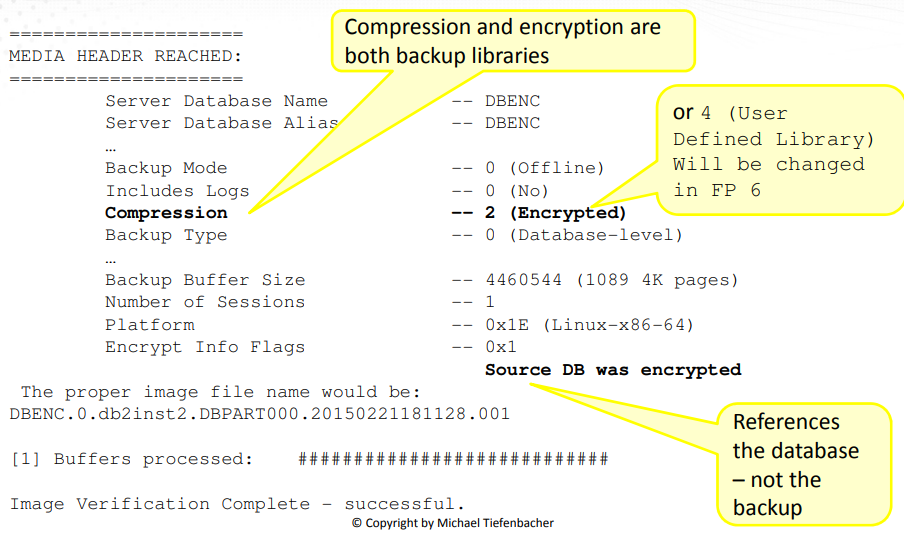
* Lựa chọn 1: DB CFG – ENCROPTS (Mặc định lâu dài)
* db2 "update db cfg for test using encropts 'Cipher=AES:Key Length=256:Master Key Label=MKL\_TestDB'"



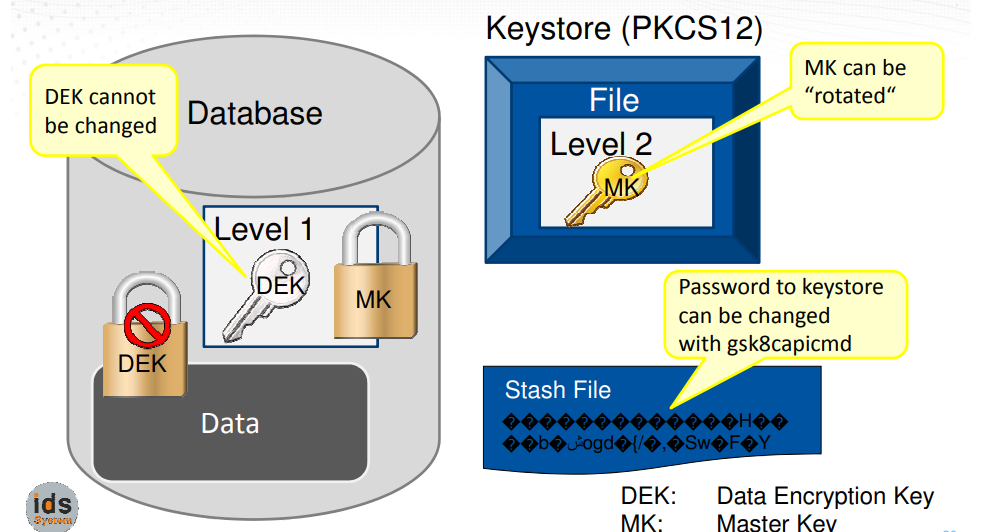
* Lựa chọn 2: ENCRLIB và ENCROPTS được sao lưu bởi lệnh command (Chỉ mã hóa được 1 bản sao lưu duy nhất)
* db2 "backup db sample to /diskb/backup encrypt encrlib '/home/db2inst2/sqllib/lib64/libdb2encr.so' encropts 'Master Key Label=MKL\_SampleDB'"

**Check Backup – db2ckbkp:**

db2ckbkp -h /diskb/backup/DBENC.0.db2inst2.DBPART000.20150221181128.001



**Thay đổi mật khẩu:**



**Xoay vòng khóa chính:**

db2 connect to dbenc

db2 "call admin\_rotate\_master\_key('MKL\_dbencDB')"

Value of output parameters

Parameter Name: LABEL

Parameter Value: MKL\_dbencDB

Return Status = 0

**Thay đổi mật khẩu của Keystore:**

./sqllib/gskit/bin/gsk8capicmd\_64 -keydb -changepw -db keystore.p12 -new\_pw IDS\_Michael –stash

**Thông tin mã hóa:**

ADMIN\_GET\_ENCRYPTION\_INFO:

Trả về thông tin mã hóa:

* Thuật toán mã hóa
* Thông tin Keystore
* Thời gian quay vòng

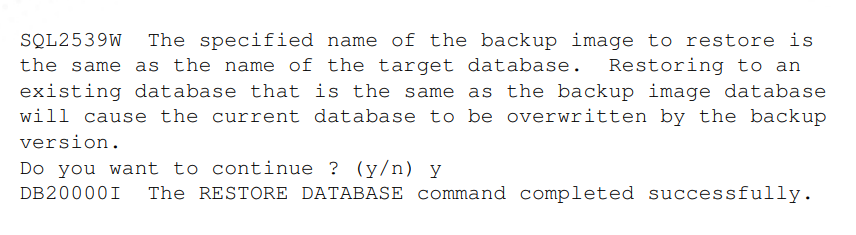
….........................

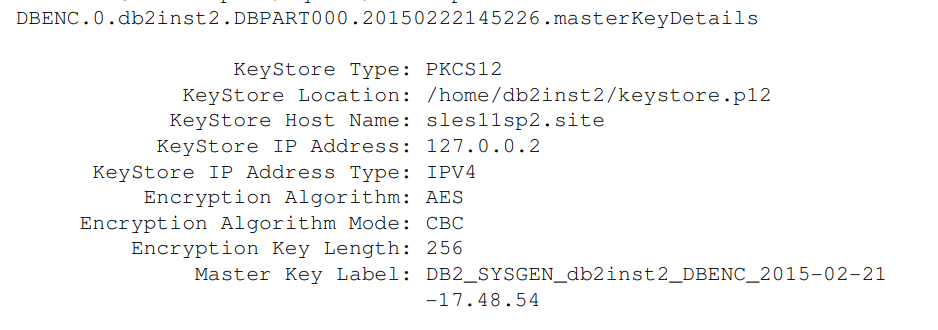
db2 select \* from table (admin\_get\_encryption info ())

db2pd –d db2encr –encryptioninfo

**Khôi phục:**

db2 "restore db dbenc from /diskb/backup/ taken at 20150222145226 encropts 'show master key details' “





\*Chú ý:

* Khi ta khôi phục lại, cơ sở dữ liệu có thể sẽ bị ghi đè, nếu không muốn bị ghi đè ta có thể chọn “y”
* Nếu cơ sở dữ liệu khôi phục không tồn tại thì một cơ sở dữ liệu mới sẽ được tạo ra và cần được gỡ bỏ sau đó.
* Nếu không được mã hóa thì cơ sở dữ liệu có thể bị ghi đè.

# Kết luận và tài liệu tham khảo

### Kết luận

Mã hóa dữ liệu bằng DB2 giúp người dùng mã hóa thông tin trong database đảm bảo tính an toàn của ứng dụng, an toàn người dùng. Giúp người dùng yên tâm sử dụng sản phẩm ứng dụng.

### Tài liệu tham khảo

-[How to Encrypt File System in AIX? | ASGAUR](http://www.asgaur.com/wp/how-to-encrypt-file-system-in-aix/)

-[Database encryption using AIX encrypted file system (EFS) - IBM Documentation](https://www.ibm.com/docs/en/db2/11.5?topic=rest-database-encryption-using-aix-encrypted-file-system)

-[IBM InfoSphere Guardium Data Encryption for encryption of data at rest - IBM Documentation](https://www.ibm.com/docs/en/db2/11.5?topic=rest-infosphere-guardium-data-encryption)

-[SSL là gì? Tổng quan về chứng chỉ SSL quan trọng phải biết - Trung tâm hỗ trợ kỹ thuật | MATBAO.NET](https://wiki.matbao.net/ssl-la-gi-tong-quan-ve-ssl/#loi-ich-khi-su-dung-ssl-la-gi)

- [Triển khai mã hóa cơ sở dữ liệu gốc DB2 - Tài liệu của IBM](https://www.ibm.com/docs/en/db2/10.5?topic=encryption-implementing-db2-native-database)