TẠP CHÍ KHOA HỌC, TRƯỜNG ĐẠI HỌC HỒNG ĐỨC - SỐ 24. 2015

MÃ HÓA DỮ LIỆU - CÂN BẰNG GIỮA SỰ BẢO MẬT VÀ HIỆU SUẤT THỰC THI ỨNG DỤNG

Lê Đình Nghiệp[1], Trịnh Thị Phú[1], Lê Văn Hào[1]

TÓM TẮT

Phát triển một chiến lược mã hóa cơ sở dữ liệu (CSDL) phải xem xét đến yếu tố cân bằng giữa yêu cầu về an ninh với mong muốn cho hiệu suất cao. Mã hóa mức CSDL độc lập với mã hóa ở mức tập tin hay ứng dụng là phương pháp lý tưởng để bảo vệ dữ liệu nhạy cảm và cho hiệu suất tối ưu. Chỉ dựa vào phạm vi bảo mật và kiểm soát truy cập sẽ không cung cấp bảo mật đầy đủ. Giải pháp mã hóa CSDL đóng gói được chứng minh là lựa chọn tốt nhất để bảo vệ dữ liệu nhạy cảm. Đây là một giải pháp chuyên biệt và phức tạp, nếu nguồn lực nội bộ không có chuyên môn về mật mã trong môi trường thông tin, chuyên gia bên ngoài nên được sử dụng để đảm bảo hiệu suất thực thi cao. Bài viết này xem xét các khía cạnh hiệu suất của các kiến trúc chi phối việc mã hoá CSDL.

Từ khóa: *Cơ sở dữ liệu, hiệu suất thực thi, an ninh cơ sở dữ liệu, mật mã, mã hóa.*

1. MỞ ĐẦU

Cùng với sự bùng nổ thông tin, sự bảo mật của nó cũng yêu cầu tăng lên. Hiện nay có rất nhiều kiến trúc, kỹ thuật, và công cụ sẵn có và các tổ chức có thể áp dụng để đảm bảo cả 2 mặt an ninh và hiệu suất thực thi đợợc tối ợu hóa. Mỗi cách tiếp cận có ợu điểm và nhợợc điểm của nó. An ninh CSDL là một lĩnh vực nghiên cứu rộng bao gồm các chủ đề nhợ bảo mật cơ sở dữ liệu thống kê [5], phát hiện xâm nhập trái phép [7], và gần đây nhất là bảo tồn sự riêng tợ trong khai thác dữ liệu [6]. Các nghiên cứu về an toàn thông tin trợớc [3] [1] đây không giải quyết các vấn đề quan trọng về hiệu suất thực thi. Trong bài viết này, chúng tôi xác định và đánh giá các vấn đề quan trọng nhất đảm bảo tính bảo mật cao đồng thời vẫn đảm bảo hiệu suất thực thi trong CSDL, ứng dụng một kiến trúc cụ thể cài đặt cho CSDL thi trắc nghiệm trực tuyến. Để đạt đợợc điều đó, chúng tôi đã phân tích nhiều giải pháp khác nhau.

2. CÁC NHÂN TỐ HỖ TRỢ MÃ HÓA

Có ba nhân tố chính hỗ trợ mã hóa trong CSDL. Một trong các nhân tố đó là đặc tính chia nhỏ đợợc của dữ liệu đợợc mã hóa. Cột, hàng, trang dữ liệu thợờng có

[1] *ThS. Giảng viên Khoa Công nghệ Thông tin và Truyền thông, trường Đại học Hồng Đức*

kích thước là 4KB có thể là các lựa chọn tốt. Trong đó cột dữ liệu là sự lựa chọn tốt nhất, bởi vì nó sẽ làm giảm thiểu kích thước cần mã hóa. Tuy nhiên, chúng tôi đã phát hiện ra, điều này yêu cầu phương pháp mã hóa được nhúng bên trong CSDL quan hệ hoặc các máy chủ CSDL. Nếu bỏ qua đặc tính chia nhỏ để bảo vệ sẽ ảnh hưởng không nhỏ đến mức độ bảo mật [8]. Nhân tố thứ hai là sự không tương thích giữa phần mềm và mức phần cứng của các thuật toán mã hóa. Kết quả cho thấy nhân tố này làm cho tác động đáng kể về hiệu suất. Chúng tôi đã phát hiện ra sự mã hóa bên trong các CSDL quan hệ dựa trên sự cài đặt ở mức phần cứng của các thuật toán mã hóa mất một chi phí khởi tạo đáng kể cho một thao tác mã hoá. Chỉ có một vài nhà cung cấp CSDL hiện đang hỗ trợ mã hóa ở mức hàng hoặc trang dữ liệu làm giảm chi phí này trên dữ liệu lớn. Nhân tố thứ ba là vị trí của các dịch vụ mã hóa - dịch vụ cục bộ, dịch vụ từ xa, hoặc dịch vụ mạng đính kèm. Chọn điểm thực hiện sẽ ảnh hưởng đáng kể đến mô hình bảo mật tổng thể. Dữ liệu mã hóa càng sớm, càng an toàn hơn trong môi trường CSDL.

2.1. Mã hóa mức CSDL

Mã hóa mức CSDL cho phép đảm bảo dữ liệu khi nó được đọc/ghi từ một CSDL. Loại mã hóa này thường được thực hiện ở mức cột bên trong một bảng CSDL, nếu kết hợp bảo mật CSDL với truy cập điều khiển, có thể ngăn chặn hành vi trộm cắp dữ liệu quan trọng. Mã hóa mức CSDL cũng bảo vệ chống lại một loạt mối đe dọa, như phương tiện lưu trữ bị đánh cắp, Các tấn công vào thiết bị lưu trữ, tấn công mức CSDL. Mã hóa mức CSDL loại bỏ tất cả thay đổi ứng dụng được yêu cầu trong mô hình mức ứng dụng, và tạo ra một xu hướng phát triển mã hóa nhúng bên trong một hệ quản trị CSDL thông qua việc sử dụng các thủ tục lưu trữ và các kích hoạt. Vì mã hóa/giải mã chỉ xảy ra bên trong CSDL, giải pháp này không yêu cầu người dùng phải hiểu hay khám phá những đặc điểm truy cập ứng dụng dữ liệu được mã hóa. Giải pháp này để chắc chắn dữ liệu an toàn, nó đòi hỏi một số công việc tích hợp ở mức CSDL, bao gồm sửa đổi lược đồ CSDL đã có, sử dụng các kích hoạt và các thủ tục lưu trữ để thực hiện mã hóa/giải mã. Ngoài ra, cần xem xét cẩn thận đến tác động về mặt hiệu năng, đặc biệt là khi việc hỗ trợ cho tăng tốc độ tìm kiếm theo chỉ số trên dữ liệu mã hóa không được sử dụng. Trước tiên, cần áp dụng một phương pháp tiếp cận để mã hóa những dữ liệu nhạy cảm. Sau đó, phải xem xét mức độ mã hóa được áp dụng để tận dụng phần cứng tăng mức độ an ninh, giảm tải quá trình mã hóa nhằm giảm thiểu các tác động xấu đến hiệu suất. Kiểu mã hoá này dễ bị hư tổn ở chỗ nó không có khả năng bảo vệ chống lại các cuộc tấn công mức ứng dụng vì chức năng mã hóa được thực hiện nghiêm ngặt bên trong hệ quản trị CSDL.

2.2. Mã hóa ở mức lơu trữ

Mã hóa cấp lơu trữ cho phép mã hóa dữ liệu ở hệ thống lơu trữ, ở cấp độ tập tin hoặc ở cấp độ khối. Ngày nay trong môi trơờng lơu trữ lớn, mã hóa ở mức lơu trữ xác định một yêu cầu đảm bảo an ninh dữ liệu mà không cần sử dụng mặt nạ hoặc phân vùng LUN (Logical Unit Number). Trong khi giải pháp này có thể phân nhóm làm việc và cung cấp các bảo mật, nó cũng có một số hạn chế. Chỉ bảo vệ chống lại một phạm vi hẹp các mối đe dọa, cụ thể là khi phơơng tiện truyền thông bị trộm cắp và các tấn công hệ thống lơu trữ. Tuy nhiên, mã hóa ở mức lơu trữ cấp không bảo vệ chống lại phần lớn các cuộc tấn công mức ứng dụng hoặc CSDL, đe dọa tới dữ liệu nhạy cảm. Cơ chế bảo mật lơu trữ hiện tại chỉ cung cấp mã hóa mức khối, chúng không cung cấp khả năng mã hóa dữ liệu bên trong một ứng dụng hoặc CSDL mức trơờng dữ liệu. Hậu quả là chỉ có thể mã hóa toàn bộ một CSDL, chứ không mã hóa đơợc các thông tin cụ thể nằm trong nó.

3. LỰA CHỌN KIẾN TRÚC CÀI ĐẶT MÃ HÓA

Chúng tôi đã xem xét một số kết hợp có thể có của các cách tiếp cận mã hóa khác nhau, cụ thể là ở cấp độ phần cứng và phần mềm, tính chia nhỏ dữ liệu khác nhau. Chúng tôi bắt đầu với phần mềm mã hóa ở cấp cơ sở. Sau đó chúng tôi phát triển hỗ trợ tăng tốc tìm kiếm có hỗ trợ trên chỉ số với các trơờng đơợc mã hóa, và xem xét hiệu suất thấp khi tìm kiếm trên các trơờng mã hóa, bao gồm các trơờng chỉ số chính.

3.1. Mã hóa phần mềm cơ bản

Trơớc tiên, chúng tôi xem xét một số thuật toán mã hóa AES, RSA và Blowfish sử dụng cho việc cài đặt. Chúng tôi tiến hành thí nghiệm bằng cách sử dụng các thuật toán này và thấy rằng hiệu suất và bảo mật của thuật toán AES là tốt hơn so với RSA và Blowfish. AES rất nhanh so với thuật toán mã hóa nổi tiếng khác nhơ DES. DES là một thuật toán mã hóa khối 64-bit, có nghĩa là dữ liệu đơợc mã hóa và giải mã trong khối 64-bit. Điều này không có ý nghĩa trên dữ liệu ngắn. Dữ liệu 8-bit, khi mã hóa sẽ cho kết quả 64 bit. Chúng tôi cũng đã cài đặt phơơng pháp bảo mật nhằm duy trì kiểu, chiều dài trơờng dữ liệu sau khi đơợc mã hoá trong CSDL của mình. Để cài đặt AES, ngơời dùng tự định nghĩa một hàm (còn gọi là hàm ngoài) và đăng ký vào CSDL. Một khi nó đã đơợc đăng ký, nó có thể đơợc sử dụng để mã hóa dữ liệu trên một hoặc nhiều trơờng. Việc truy xuất dữ liệu là an toàn vì dữ liệu lơu trữ đơợc mã hóa trơớc khi thao tác trên chúng.

3.2 Mã hóa cấp độ phần cứng cơ bản

Chúng tôi đã nghiên cứu sử dụng HSM FIPS -140- 1 mức 3 với sự kết hợp giữa các khóa phần cứng và phần mềm. Các khóa chính đơợc tạo ra đơợc mã

hóa/giải mã trên HSM. Các khóa chính không đoợc đoa ra bên ngoài HSM. Chi phí mã hóa/giải mã bao gồm chi phí khởi tạo, trong đó gồm hàm và lời gọi tới phần cứng, và chi phí thực hiện thuật toán mã hóa/giải mã, chi phí này phụ thuộc vào kích thoớc của dữ liệu đầu vào. Điều này nhấn mạnh rằng, chi phí khởi tạo sẽ đoợc tính mỗi khi một hàng đoợc xử lý để mã hóa. Chúng tôi sử dụng phần cứng mã hóa từ các nhà cung cấp khác nhau, bao gồm IBM, Eracom, nCipher, và Chrysalis làm thử nghiệm. Trong thử nghiệm, chúng tôi sử dụng bộ vi xử lý kiến trúc mã hóa IBM S/390 có sẵn trong môi troờng IBM OS/390. IBM DB2 dùng cho OS/390 cung cấp một thiết bị gọi là "editproc" (còn gọi là các thủ tục chỉnh sửa thoờng xuyên), có thể đoợc kết hợp với một bảng CSDL. Một thủ tục chỉnh sửa đoợc gọi cho một hàng của bảng CSDL mỗi khi hàng đoợc truy cập bởi DBMS. Chúng tôi đăng ký một thủ tục chỉnh sửa mã hóa/giải mã cho các bảng. Khi một yêu cầu đọc/ghi tác động đến một hàng trong bảng dữ liệu này, thủ tục chỉnh sửa gọi thuật toán mã hóa/giải mã mà đoợc cài đặt trong phần cứng cho toàn bộ hàng. Chúng tôi lựa chọn DES [3] cho việc mã hóa mức phần cứng.

4. HIỆU SUẤT CỦA CÁC KIẾN TRÚC MÃ HÓA

Có ba kiến trúc mã hóa là: 1) thiết bị mã hóa đính kèm mạng, 2) Phần mềm, 3) Sự kết hợp phần mềm và Module an toàn phần cứng (HSM). Mỗi kiến trúc có oợu và nhoợc điểm của nó.

4.1. Xem xét hiệu suất

Chúng tôi nghiên cứu SQL chuẩn sử dụng trong thí nghiệm của mình. Dùng một số thí nghiệm đơn giản trên Oracle và DB2. Các khía cạnh công nghệ để phát triển CSDL riêng nhoị một thành phần cơ sở hạ tầng công nghệ dẫn đến các thách thức nghiên cứu mới. Đầu tiên và troớc hết là vấn đề quản lý khóa. Hầu hết các công ty xem dữ liệu của họ nhoị là một tài sản vô giá. Hệ thống quản lý khoá sẽ cần phải cung cấp biện pháp an ninh đủ mạnh để bảo vệ việc sử dụng phân tán khóa. Chúng tôi đề xuất kết hợp giữa phần cứng và phần mềm dựa trên hệ thống mã hóa dữ liệu là giải pháp cho vấn đề này. Đề xuất một khả năng kiểm soát và chính sách phân phối để kiểm soát việc sử dụng các khóa. Nghiên cứu chi tiết về giải pháp này đoợc trình bày doới đây.

4.2. Mã hóa đính kèm mạng

Mã hóa đính kèm mạng (Encryption Network Attached (NAED)) đoợc cài đặt nhoị một dụng cụ mã hóa đính kèm mạng với quy mô, số loợng của các thiết bị có sẵn. Một NAED là một thiết bị phần cứng mà coị trú trên mạng, định vị các khóa

mã hóa và thực thi tất cả các thao tác mã hóa. Cấu trúc này bổ sung khóa vật lý tách rời với dữ liệu. Tuy nhiên, việc bổ sung này đi kèm với một giá đắt, hiệu suất có thể còn tồi tệ hơn 10-100 lần so với phương pháp khác. Các tiêu chuẩn đã cho thấy một thông lượng giữa 440 và 1.100 hàng mã hóa mỗi giây. Một hệ thống với 12 máy chủ cơ sở dữ liệu thực hiện mã hóa 4.200 hàng mã hóa mỗi giây với năm dụng cụ mã hóa đính kèm mạng.

Có ba chi phí đi kèm với kiến trúc mạng này. Hãy cùng khám phá một ví dụ đơn giản để chứng minh các chi phí trên khi một người dùng yêu cầu 500.000 dòng dữ liệu được mã hóa:

Khi một người dùng yêu cầu dữ liệu an toàn, hệ thống an ninh quản lý tiến trình lấy dữ liệu được mã hóa từ cơ sở dữ liệu, đảm bảo yêu cầu là từ một người dùng có thẩm quyền, và thực hiện quá trình giải mã. Trong kiến trúc này, các tác nhân mã hóa xử lý các yêu cầu và khôi phục các dữ liệu được mã hóa từ CSDL. Nó sẽ gửi các dữ liệu được mã hóa qua mạng để được giải mã bởi các NAED. Bên trong NAED là khóa và các thuật toán để giải mã dữ liệu. Tuy nhiên, một khi giải mã, thông tin bản rõ cần phải được gửi lại đường truyền đến máy chủ CSDL. Điều này đòi hỏi chúng ta bảo mật lại thông tin trong quá trình chuyển, thường là thông qua một quá trình trao đổi an toàn như SSL. Khi dữ liệu tới tác nhân trên máy chủ CSDL, nó phải cho ra được bản rõ, và sau đó được phục vụ cho đến ứng dụng gọi đến.

1. Kiến trúc NAED có ba đặc điểm của mã hóa. Trong ví dụ trên, 500.000 dòng dữ liệu được gửi đi trên đường truyền được giải mã tại NAED. Các bản rõ được mã hóa sử dụng SSL để gửi lại qua mạng và giải mã ở các CSDL để được bản rõ phục vụ cho các ứng dụng.

2. Chi phí mạng được tính bằng việc gửi tất cả 500.000 hàng qua mạng để được giải mã bởi các NAED và sau đó phải trở lại trên mạng để các CSDL.

3. NAED là một thiết bị không trạng thái và cần phải được khởi tạo/thiết lập trước khi mỗi hàng được giải mã. Trong ví dụ đơn giản này, NAED được thiết lập 500.000 lần. Các thiết lập cần chi phí khá lớn.

Kiến trúc thiết bị mã hóa đính kèm mạng (NAED) đã được chứng minh trong các thử nghiệm, do mất ba loại chi phí nên tồi tệ hơn cấu trúc khác. Mỗi lần đi chuyển trên mạng là khoảng 1 phần nghìn giây cho mỗi hàng. Trong ví dụ trên đây sẽ mất 500.000 x 1ms = 500 giây so với 1-25 giây với cấu trúc khác.

4.3. Hệ thống hybrid

Cấu trúc Hybrid kết hợp việc nâng cao hiệu suất của cấu trúc phần mềm với việc bổ sung an ninh cho thiết bị phần cứng. Một HSM, trong một số trường hợp, là một cách lý tưởng để bổ sung thêm bảo vệ cho các phần tử quan trọng nhất - các khóa

mã hóa. Các thiết bị HSM được chứng minh là nhanh và nhiều, tuyệt vời để lưu trữ những viên ngọc quý - các khóa mã hóa.

Việc thực hiện trong kiến trúc này là cơ bản giống với cấu trúc phần mềm trước đó, với một quá cảnh thường xuyên tới HSM để làm mới và lấy chìa khóa mã hóa tổng thể. Trong phần lớn thời gian xử lý, hiệu suất là giống với giải pháp phần mềm. Trong ví dụ 500.000 hàng của chúng tôi, ngược lại với cấu trúc NAED - nơi mà tất cả 500.000 hàng truyền đến NAED - dịch vụ mã hóa trong máy chủ CSDL truy cập các khóa từ HSM mỗi lần và sau đó tất cả các hoạt động mã hóa được hoàn thành trong CSDL bởi các dịch vụ phần mềm mã hóa.

Hệ thống Hybrid thực hiện quy trình phân phối có quy mô với số lượng các bộ vi xử lý và máy chủ CSDL có sẵn. Trong kiến trúc phần mềm máy chủ CSDL sẽ trở thành nền tảng cho các dịch vụ mã hóa. Khi các ứng dụng đòi hỏi thông tin an toàn, dịch vụ mã hóa yêu cầu các dữ liệu được mã hóa từ máy chủ CSDL, thực hiện giải mã cục bộ, trả về thông tin bản rõ cho các ứng dụng gọi đến. Tất cả chi phí mạng và mã hóa (ví dụ như SSL) đã được loại bỏ khỏi đường truyền, tối ưu hóa thời gian trả lời và thông lượng. Ngoài ra, vì nó không sử dụng một thiết bị phần cứng riêng biệt nên không có bất kỳ chi phí thiết lập. Trong ví dụ của chúng tôi, giải mã của 500.000 hàng được xử lý trong các máy chủ CSDL. Do sự giảm các vị trí mã hóa, loại bỏ các lưu lượng đường truyền, và chi phí thiết lập, nên hiệu suất rất cao. Trong ví dụ của chúng tôi 500.000 hàng, hiệu suất được cải thiện rất nhiều: 500.000 x 0,05 ms = 25 * giây.

SQL Server cho thấy thông lượng vào khoảng 3.000 đến 32.000 hàng được giải mã mỗi giây, tùy thuộc vào sự kết hợp tối ưu của mức mã hóa cột và mức mã hóa bảng, và số lượng dữ liệu bảng vùng đệm, SQL Server 2000 ban đầu kiểm tra sử dụng một hệ thống kiểm tra cấp thấp chạy Windows với một bộ xử lý 1,6 GHz, 1 GB bộ nhớ RAM vật lý, và 3 GB bộ nhớ RAM ảo.

Các tiêu chuẩn của DB2 của IBM cho thấy một thông lượng 187.000 dòng giải mã mỗi giây, với 20 người dùng đồng thời. Điều này cho khả năng giải mã 187.000 bảng CSDL hàng mỗi giây. Các bảng kiểm tra bao gồm 80 byte dữ liệu được mã hóa trong mỗi hàng. Chúng tôi bão hòa tất cả sáu bộ vi xử lý RS6000 sử dụng 100% khi thử nghiệm với 1.000 người sử dụng đồng thời.

4.4. Hạn chế số lượng các hoạt động mật mã

Như đã đề cập trong thảo luận ở trên, mỗi hoạt động mã hóa có thêm chi phí. Có nhiều kỹ thuật, hỗ trợ trong các giải pháp để giới hạn số hoạt động cần thiết. Sử dụng các kỹ thuật cho biết sự khác biệt giữa chấp nhận được và không thể chấp nhận trên phương diện hiệu suất.

Việc lập chỉ mục, cho phép mã hóa dữ liệu đoợc tìm kiếm mà không cần thiết của giải mã thành các bản rõ troớc. Nhiều giải pháp vẫn yêu cầu các dữ liệu đoợc giải mã troớc khi đoợc tìm kiếm. Điều này tạo ra một sự gia tăng lớn về số loợng các hoạt động mã hóa và do đó cản trở tới hiệu suất. Ngoài ra thêm một chỉ mục tìm kiếm tăng tốc giảm thời gian trả lời và số loợng hàng giải mã, từ 10 đến 30 lần cho một số các truy vấn khi so sánh với một giải pháp mà không đoợc sử dụng một chỉ mục tìm kiếm tăng tốc cho các cột đoợc mã hóa.

Tìm kiếm so khớp chính xác một giá trị đoợc mã hóa trong một cột là có thể, với điều kiện là các vector khởi động cùng đoợc sử dụng cho toàn bộ cột. Mặt khác, tìm kiếm so khớp một phần trên dữ liệu đoợc mã hóa trong một cơ sở dữ liệu có thể là thử thách và có thể dẫn đến việc duyệt toàn bộ bảng nếu hỗ trợ tìm kiếm theo chỉ số không đoợc sử dụng. Cột mã hóa có thể là một khóa chính hoặc một phần của một khóa chính, vì mã hóa một phần dữ liệu là ổn định (có nghĩa là nó luôn luôn tạo ra kết quả giống nhau), và không có hai phần riêng biệt của dữ liệu sẽ cho ra các bản mã hóa giống nhau, miễn là khóa và vectơ khởi động đoợc sử dụng là nhất quán. Tuy nhiên, khi mã hóa toàn bộ cột của một cơ sở dữ liệu có sẵn, tùy thuộc vào phoợng pháp chuyển đổi dữ liệu, ngoời quản trị có thể phải xóa những khóa chính đã có cũng nhoị các khoá liên kết khác và tạo lại chúng sau khi dữ liệu đoợc mã hóa. Vì lý do này, mã hóa cột là một phần của một ràng buộc khóa chính không đoợc khuyến cáo nếu việc hỗ trợ tìm kiếm theo chỉ số trên dữ liệu mã hóa không đoợc sử dụng. Tuy nhiên, sự quan tâm đặc biệt phải đoợc thực hiện trong quá trình tạo khóa. Để chuyển đổi một bảng hiện có mà vẫn giữ các dữ liệu đoợc mã hóa, tất cả các bảng mà nó có những ràng buộc troớc tiên phải đoợc xác định. Tất cả các bảng tham chiếu phải đoợc chuyển đổi cho phù hợp. Trong một số troờng hợp, các ràng buộc tham chiếu phải đoợc tạm thời vô hiệu hóa hoặc xóa bỏ để cho phép di chuyển dữ liệu hiện có. Chúng có thể đoợc kích hoạt hoặc tái tạo lại một khi dữ liệu cho tất cả các bảng liên quan đoợc mã hóa. Do tính phức tạp này, mã hóa một cột mà là một phần của ràng buộc khóa ngoại không đoợc khuyến cáo, nếu các công cụ triển khai tự động không đoợc sử dụng. Không giống nhoị các chỉ số và khóa chính, mã hóa các khóa ngoài thoờng không ảnh hoởng tới hiệu suất.

Một boớc quan trọng trong việc đoịa ra chiến loợc mã hóa là xác định các dữ liệu cần đoợc bảo vệ và các dữ liệu có thể tồn tại ở dạng bản rõ. Nếu bỏ qua boớc này nhiều dữ liệu đoợc mã hóa hơn so với yêu cầu. Mặc dù nó có vẻ vô hại, cung cấp nhiều bảo vệ hơn, nhoợng mã hóa nhiều dữ liệu hơn tất yếu phải trả một giá cho hiệu suất hoạt động. Cấp độ mã hóa cột cần đoợc cài đặt để hỗ trợ chính sách bảo mật dữ liệu.

KẾT LUẬN

Chúng tôi xác định hiệu suất là một vấn đề đặc biệt quan trọng thiết yếu để đánh giá các giải pháp khác nhau cho việc mã hóa cơ sở dữ liệu. Trong bài báo này, chúng tôi đã thảo luận về Hybrid, một giải pháp bảo mật CSDL đơợc xây dựng trên nhiều hệ CSDL quan hệ lớn. Mô hình Hybrid đơợa ra nhiều thách thức đáng kể trong đó có các chi phí bổ sung tìm kiếm trên dữ liệu đơợc mã hóa đảm bảo sự riêng tơ của dữ liệu, và quản lý của thành phần của cơ sở hạ tầng công nghệ thông tin. Chúng tôi đã giải quyết những vấn đề này và áp dụng nó vào trong CSDL thi trắc nghiệm online. Mô hình Hybrid nhấn mạnh một cách tiếp cận rộng cho vấn đề bảo mật dữ liệu và tính riêng tơ của dữ liệu trong đó một quản trị viên an ninh bảo vệ sự riêng tơ
ở mức độ của các trơờng và bản ghi, và quản trị CSDL cung cấp cơ chế liên tục để tạo, lơu trữ, đảm bảo truy cập an toàn tới cơ sở dữ liệu. Một mô hình nhơ vậy làm giảm bớt sự cần thiết cho các tổ chức mua phần cứng đắt tiền, đối phó với những thay đổi phần mềm, và thuê chuyên gia làm nhiệm vụ phát triển quản lý khóa mã hóa. Qua đánh giá với các chơơng trình mã hóa khác nhau chúng tôi thấy sự sụt giảm mạnh mẽ thời gian thực hiện truy vấn.

TÀI LIỆU THAM KHẢO

[1] G. Davida, D. Wells, and J. Kam. A database encryption system with subkeys. ACM Transactions on Database Systems, 6(2), 1981.

[2] DES. Data encryption standard. FIPS PUB 46, Federal Information Processing Standards Publication, 1977.

[3] J. He and M. Wang. Encryption in relational database management systems. In Proc. Fourteenth Annual IFIP WG 11.3 Working Conference on Database Security (DBSec‟00), Schoorl, The Netherlands, 2000.

[4] Mattsson, Ulf T., „A DATABASE ENCRYPTION SOLUTION‟, LinuxSecurity.com, 28 July 2004, http://www.linuxsecurity.com/content/view/116068/65/

[5] N. R. Adam and J. C. Wortman. Security-control methods for statistical databases. ACMComputing Surveys, 21(4):515– 556, Dec. 1989.

[6] R. Agrawal and J. Kiernan. Watermarking relational databases. In 28th Int‟l Conference on Very Large Databases, Hong Kong, China, August 2002.

[7] T. F. Lunt. A survey of intrusion detection techniques. Computer & Security, 12(4), 1993.

DATA ENCRYPTION - BALANCE BETWEEN SECURITY AND PERORMANCE

Le Dinh Nghiep, Trinh Thi Phu, Le Van Hao

ABSTRACT

Developing a database encryption strategy is necessary to consider the balance between security requirements and the desire for high performance. Encryption at the database level, versus application and file level is an ideal approach to protect sensitive data and optimize performance. Relying on perimeter security and database access control is not adequate security. Packaged database encryption solutions have proven to be the best alternative. This is a specialized and complex solution, if internal resources do not have the cryptography expertise in relation to information technology environment, outside expertise should be used to ensure superior performance. This paper reviews the performance aspects of topologies for database encryption.

Key word: *Performance, Database Security, Encryption, Privacy.*