

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA AN TOÀN THÔNG TIN**



**BÁO CÁO BÀI TẬP LỚN
HỌC PHẦN: AN TOÀN MẠNG NÂNG CAO
MÃ HỌC PHẦN: INT1483**

ĐỀ TÀI

Tìm hiểu về hạ tầng khóa công khai PKI, chứng chỉ số và chữ ký số. Cài đặt và demo thử nghiệm hệ thống quản lý chứng chỉ số EJBCA CE.

Các sinh viên thực hiện:

B21DCATxxx	Lê Xuân Vương
B21DCATxxx	Nguyễn Khắc Tuyên
B21DCATxxx	Đặng Quang Vinh
B21DCAT193	Mai Đức Trung
B21DCATxxx	Lương Hà Anh Quân

Tên nhóm: 10

Tên lớp: 02

Giảng viên hướng dẫn: TS. Hoàng Xuân Dậu

HÀ NỘI 3-2025

PHÂN CÔNG NHIỆM VỤ NHÓM THỰC HIỆN

Công việc/nhiệm vụ	SV thực hiện	Thời hạn hoàn thành
Giới thiệu	Lương Hà Anh Quân	12/3/2025
Tìm hiểu về hạ tầng khóa công khai PKI	Nguyễn Khắc Tuyên	20/2/2025
Chứng chỉ số và chữ kí số (Quá trình tạo kí và kiểm tra, chữ kí số RSA, DSA)	Đặng Quang Vinh	20/2/2025
Mục tiêu, chức năng của PKI	Lương Hà Anh Quân	20/2/2025
Ứng dụng PKI	Mai Đức Trung	20/2/2025
Cài đặt hệ thống EJBCA CE	Lê Xuân Vương	25/2/2025
Demo một số chức năng của EJBCA CE	Lê Xuân Vương	25/2/2025
Demo phân quyền SubCA trong EJBCA CE	Mai Đức Trung	10/3/2025
Demo cấp chứng chỉ cho Webserver	Mai Đức Trung	10/3/2025
Kết luận	Lê Xuân Vương	25/2/2025
Kết Chương	Đặng Quang Vinh	12/3/2025
Làm báo cáo	Lê Xuân Vương	12/3/2025
Làm slide	Lương Hà Anh Quân	12/3/2025

NHÓM THỰC HIỆN TỰ ĐÁNH GIÁ

TT	SV thực hiện	Thái độ tham gia	Mức hoàn thành CV	Kỹ năng giao tiếp	Kỹ năng hợp tác	Kỹ năng lãnh đạo
1	Lê Xuân Vương	5	5	5	5	5
2	Nguyễn Khắc Tuyên	5	5	4	5	4
3	Đặng Quang Vinh	5	5	5	4	4
4	Mai Đức Trung	5	4	5	5	4
5	Lương Hà Anh Quân	5	5	4	5	4

Ghi chú:

- Thái độ tham gia: Đánh giá điểm thái độ tham gia công việc chung của nhóm (từ 0: không tham gia, đến 5: chủ động, tích cực).
- Mức hoàn thành CV: Đánh giá điểm mức độ hoàn thành công việc được giao (từ 0: không hoàn thành, đến 5: hoàn thành xuất sắc).
- Kỹ năng giao tiếp: Đánh giá điểm khả năng tương tác, giao tiếp trong nhóm (từ 0: không hoặc giao tiếp rất yếu, đến 5: giao tiếp xuất sắc).
- Kỹ năng hợp tác: Đánh giá điểm khả năng hợp tác, hỗ trợ lẫn nhau, giải quyết mâu thuẫn, xung đột
- Kỹ năng lãnh đạo: Đánh giá điểm khả năng lãnh đạo (từ 0: không có khả năng lãnh đạo, đến 5: có khả năng lãnh đạo tốt, tổ chức và điều phối công việc trong nhóm hiệu quả).

MỤC LỤC

MỤC LỤC	4
DANH MỤC CÁC HÌNH VẼ	5
DANH MỤC CÁC BẢNG BIÊU.....	Error! Bookmark not defined.
DANH MỤC CÁC TỪ VIẾT TẮT	7
MỎ ĐẦU	8
CHƯƠNG 1. TÌM HIỂU VỀ CHỨNG CHỈ SỐ, CHỮ KÝ SỐ	9
1.1 Tổng quan	9
1.2 Chứng chỉ số.....	10
1.3 Chữ ký số	11
1.4 Các chuẩn và tiêu chuẩn liên quan	12
1.5 Các vấn đề bảo mật.....	13
1.6 Kết chương.....	14
CHƯƠNG 2. TÌM HIỂU VỀ KIẾN TRÚC HẠ TẦNG KHÓA CÔNG KHAI PKI.....	15
2.1 Khái quát.....	15
2.2 Kiến trúc hạ tầng khóa công khai PKI.....	15
2.3 Mục tiêu, chức năng của PKI.....	18
2.4 Mô hình PKI.....	20
2.5 Ứng dụng của hạ tầng PKI.....	23
2.6 Kết chương.....	26
CHƯƠNG 3. Demo thử nghiệm hệ thống quản lý chứng chỉ số EJBCA CE	27
3.1 Giới thiệu hệ thống quản lý chứng chỉ số EJBCA CE	27
3.2 Demo hệ thống quản lý chứng chỉ số EJBCA CE	28
3.3 Kết chương.....	65
KẾT LUẬN	66
TÀI LIỆU THAM KHẢO	67

DANH MỤC CÁC HÌNH VẼ

Hình 1 - Quy trình của hạ tầng khóa công khai PKI	16
Hình 2 - Mô hình tin cậy phân cấp.....	21
Hình 3 - Mô hình tin cậy cầu nối	22
Hình 4- Mô hình tin cậy lai.....	22
Hình 5- Mô hình tin cậy dạng lưới.....	23
Hình 6 - Chứng chỉ số xác thực tên miền	24
Hình 7 - Email đã được mã hóa và kí số	24
Hình 8 - Ví dụ về PDF có chữ ký số	25
Hình 9 - Chứng chỉ máy khách khi kết nối với VPN	25
Hình 10– Kiến trúc EJBCA CE	28
Hình 11– Cài docker trên máy	29
Hình 12 - Cài EJBCA CE qua docker	29
Hình 13 - Chạy EJBCA CE trên localhost với cổng 8080.....	29
Hình 14 - Giao diện hệ thống EJBCA CE	30
Hình 15 - Tạo hồ sơ chứng chỉ.....	31
Hình 16 - Clone ROOTCA về làm mẫu	31
Hình 17 - Chính sửa thuật toán và thời hạn của chứng chỉ.....	32
Hình 18 - Bỏ chọn các cấu hình không cần thiết vì là ROOT CA	32
Hình 19 - Bỏ LDAP DN và lưu chứng chỉ	33
Hình 20 - Tạo Crpyto Token cho chứng chỉ Root CA	33
Hình 21 - Sinh 3 khóa-khóa ký, khóa mã hóa, khóa test	34
Hình 22 - Tạo thành công Crypto token	34
Hình 23 - Tạo Root CA tên là MyFirstRootCA	35
Hình 24 - Cài đặt khóa vừa tạo với CA	35
Hình 25 - Cài đặt các thông tin của CA và thời gian hiệu lực của CA	36
Hình 26 - Cấu hình CRL- Danh sách thu hồi chứng chỉ	36
Hình 27 - Clone phần tạo chứng chỉ Root CA.....	37
Hình 28 - Cài đặt thuật toán ECDSA với đường cong P-256.....	38
Hình 29 - Clone từ template SUBCA làm Sub CA	38
Hình 30 - Khởi tạo thành công Sub CA	39
Hình 31 - Cài đặt cấu hình khóa cho chứng chỉ Sub CA.....	39
Hình 32 - Thời hạn của chứng chỉ Sub CA.....	40
Hình 33 - Cấu hình đảm bảo Sub CA không thể cấp thêm các CA phụ dưới nó.....	40
Hình 34 - Cấu hình điểm phân phối CRL trong chứng chỉ Sub CA	40
Hình 35 - Tạo Crypto Token cho RootCA	41
Hình 36 - Tạo ba cặp khóa cài ở bước một.....	41
Hình 37 - Tạo Crypto token cho Sub CA	42
Hình 38 - Tạo ba cặp khóa tương ứng với thuật toán cho Sub CA	42
Hình 39 - Tạo Root CA -G1	43
Hình 40 Cấu hình Root CA -G1 với Crypto token.....	43
Hình 41 Cấu hình thông tin CA và CRL	44
Hình 42 Cấu hình địa chỉ dẫn CRL và OCSP	45
Hình 43 Tạo Sub CA-G1	45

Hình 44 Cấu hình Sub Ca -G1 phù hợp với Crypto token.....	46
Hình 45 Cấu hình CA và điểm phân phối CRL	46
Hình 46 Cấu hình đường dẫn CRL và OCSP	47
Hình 47 Kết quả khi tạo được Root CA-G1 và Sub CA-G1.....	47
Hình 48 Tạo hồ sơ chứng chỉ TLS Server	48
Hình 49 Cấu hình thuật toán khóa và thời hạn chứng chỉ	49
Hình 50 Cấu hình X509v3 extenstions	50
Hình 51 Cấu hình điểm CRL và OCSP với TLS Server	51
Hình 52 Cấu hình người cấp chứng chỉ.....	51
Hình 53 Cấu hình DNS cho TLS Server	52
Hình 54 Cấu hình chứng chỉ nhận, người cấp chứng chỉ và định dạng nhận chứng chỉ	53
Hình 55 Các roles mặc định trong EJBCA	54
Hình 56 Một số quyền được thiết lập trong EJBCA	54
Hình 57 Giao diện quản lý RA.....	55
Hình 58 Chọn profile cho chứng chỉ mới	55
Hình 59 Chọn cách tạo cặp khóa.....	56
Hình 60 Xác nhận lại thông tin chứng chỉ	56
Hình 61 Danh sách các profile được CA cấu hình	56
Hình 62 Danh sách các hồ sơ định nghĩa thiết bị đầu cuối.....	57
Hình 63 Giao diện định nghĩa hồ sơ thiết bị đầu cuối.....	57
Hình 64 Tạo chứng chỉ mới cho SubCA	58
Hình 65 Điện thông tin cho chứng chỉ SubCA	58
Hình 66 Nhóm người dùng SubCA.....	59
Hình 67 IP máy chủ EJBCA	59
Hình 68 File chứng chỉ của SubCA1	59
Hình 69 Thêm chứng chỉ của SubCA1 vào trình duyệt	60
Hình 70 Dùng chứng chỉ xác thực người dùng EJBCA	60
Hình 71 Giao diện EJBCA trên SubCA1	61
Hình 72 Giao diện web trên Webserver	61
Hình 73 File định nghĩa thông tin Webserver	62
Hình 74 Lệnh khởi tạo private key.....	62
Hình 75 File được tạo ra từ file định nghĩa Webserver	62
Hình 76 Khởi tạo chứng chỉ mới cho Webserver tại RA Web	63
Hình 77 Thông tin chứng chỉ Webserver.....	63
Hình 78 File chứng chỉ được tạo	64
Hình 79 Kích hoạt module SSL của apache2	64
Hình 80 File config SSL cho Webserver	64
Hình 81 Kích hoạt file config SSL.....	64
Hình 82 IP máy chủ Webserver	65
Hình 83 Webserver đã được cấp chứng chỉ	65

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Thuật ngữ tiếng Anh/Giải thích	Thuật ngữ tiếng Việt/Giải thích
PKI	Public Key Infrastructure	Hệ tầng khóa công khai
CA	Certificate Authority	Tổ chức cấp chứng chỉ
RA	Registration Authority	Tổ chức đăng ký chứng chỉ
CSR	Certificate Signing Request	Yêu cầu cấp chứng chỉ
CRL	Certificate Revocation List	Danh sách thu hồi chứng chỉ
EJBCA	Enterprise Java Beans Certificate Authority	Hệ thống quản lý chứng chỉ EJBCA
RSA	Rivest–Shamir–Adleman	Thuật toán RSA
DSA	Digital Signature Algorithm	Thuật toán chữ ký số DSA
SHA	Secure Hash Algorithm	Thuật toán băm SHA
MD5	Message Digest Algorithm 5	Thuật toán băm MD5
TLS	Transport Layer Security	Bảo mật lớp truyền tải
SSL	Secure Sockets Layer	Lớp bảo mật kênh giao tiếp

MỞ ĐẦU

Trong bối cảnh phát triển nhanh chóng của công nghệ thông tin và các giao dịch điện tử, việc bảo mật thông tin trở thành yếu tố cực kỳ quan trọng. Để đảm bảo tính xác thực, toàn vẹn và bảo mật trong các giao dịch trực tuyến, việc sử dụng hạ tầng khóa công khai (PKI), chứng chỉ số và chữ ký số là những giải pháp không thể thiếu. Hai công nghệ này giúp xác thực danh tính, bảo vệ dữ liệu và đảm bảo tính toàn vẹn của các giao dịch trong môi trường mạng.

Mục tiêu của báo cáo này là nghiên cứu về hạ tầng khóa công khai PKI, các khái niệm cơ bản về chứng chỉ số và chữ ký số, đồng thời cài đặt và thử nghiệm hệ thống quản lý chứng chỉ số EJBCA CE. Đây là một công cụ mã nguồn mở mạnh mẽ và linh hoạt, hỗ trợ tổ chức triển khai hệ thống PKI hiệu quả, bảo mật cao.

Báo cáo bài tập lớn được chia thành ba chương với nội dung chính như sau:

Chương 1 giới thiệu về chứng chỉ số và chữ ký số, là hai công nghệ cốt lõi trong bảo mật giao dịch điện tử. Chúng tôi sẽ đi sâu vào việc phân tích các chuẩn quốc tế và các tiêu chuẩn bảo mật giúp đảm bảo tính toàn vẹn, xác thực và bảo mật trong môi trường trực tuyến. Các vấn đề bảo mật liên quan đến chứng chỉ số và chữ ký số cũng sẽ được trình bày chi tiết.

Chương 2 tập trung vào nghiên cứu kiến trúc hạ tầng khóa công khai (PKI), một hệ thống bảo mật giúp đảm bảo các giao dịch điện tử được thực hiện an toàn. Các thành phần cấu thành PKI, quy trình cấp phát và quản lý chứng chỉ số sẽ được làm rõ. Mô hình PKI và cách thức hoạt động của nó sẽ được trình bày chi tiết, cùng với các biện pháp bảo mật áp dụng trong PKI.

Chương 3 sẽ giới thiệu hệ thống quản lý chứng chỉ số EJBCA CE, tiến hành cài đặt và thử nghiệm các tính năng của hệ thống này. Các bước triển khai và cấu hình EJBCA sẽ được hướng dẫn chi tiết, cùng với đánh giá hiệu quả hoạt động của hệ thống trong môi trường thử nghiệm thực tế.

CHƯƠNG 1. TÌM HIỂU VỀ CHỨNG CHỈ SỐ, CHỮ KÝ SỐ

1.1 Tổng quan

Trong thời đại số hóa, khi các giao dịch điện tử ngày càng trở nên phổ biến và phức tạp, việc đảm bảo tính xác thực, toàn vẹn và phi chối bỏ của thông tin là yếu tố then chốt. Hai công nghệ then chốt được ứng dụng rộng rãi nhằm đảm bảo các thuộc tính này là chứng chỉ số và chữ ký số.

1.1.1 Chứng chỉ số

Chứng chỉ số được ví như “hộ chiếu điện tử” của các thực thể trong hệ thống mạng. Nó cung cấp thông tin định danh của chủ sở hữu kèm theo khóa công khai, cho phép các bên khác xác thực danh tính và thiết lập kênh giao tiếp an toàn. Các chứng chỉ số thường bao gồm:

- *Thông tin định danh*: Tên tổ chức, tên người dùng, domain, địa chỉ email, v.v.
- *Khóa công khai*: Dùng để mã hóa hoặc xác thực chữ ký số.
- *Thời hạn hiệu lực*: Ngày cấp và ngày hết hạn của chứng chỉ.
- *Chữ ký số của CA*: Dấu hiệu số của Tổ chức Cấp chứng chỉ, xác nhận tính hợp pháp và nguyên vẹn của chứng chỉ.

1.1.2 Chữ ký số

Chữ ký số được sử dụng nhằm xác nhận danh tính của người ký và đảm bảo rằng nội dung của tài liệu không bị thay đổi sau khi ký. Quá trình tạo chữ ký số thường gồm hai bước chính:

- *Tạo hàm băm (hash)*: Tính toán một giá trị duy nhất (digest) từ dữ liệu cần ký. Các thuật toán băm như SHA-256, SHA-3 thường được áp dụng.
- *Mã hóa hàm băm*: Sử dụng khóa riêng của người ký (thông qua thuật toán bắt đầu xứng như RSA hoặc DSA) để mã hóa giá trị hàm băm, tạo thành chữ ký số.

Khi người nhận cần xác thực, họ sẽ giải mã chữ ký số bằng khóa công khai tương ứng (được chứa trong chứng chỉ số của người ký) và so sánh giá trị hàm băm thu được với giá trị băm tính từ dữ liệu gốc. Nếu khớp, điều đó khẳng định tính toàn vẹn của dữ liệu cũng như xác nhận danh tính của người ký.

1.1.3 Phân loại chữ ký số: RSA vs.DSA

Trong hệ thống chữ ký số hiện đại, hai thuật toán phổ biến được sử dụng là:

- *RSA (Rivest–Shamir–Adleman)*:

Sử dụng cặp khóa gồm khóa công khai và khóa riêng. RSA không chỉ được ứng dụng rộng rãi để trao đổi khóa mà còn được dùng để tạo chữ ký số bằng cách mã hóa hàm băm bằng khóa riêng. Ưu điểm của RSA là khả năng hỗ trợ nhiều ứng dụng bảo mật, tuy nhiên tốc độ xử lý của nó chậm hơn so với các giải pháp khác khi kích thước thông điệp tăng lên.

- *DSA (Digital Signature Algorithm):*

DSA là tiêu chuẩn chữ ký số do NIST đề xuất và được mô tả trong FIPS 186. Thay vì mã hóa trực tiếp hàm băm bằng khóa riêng như RSA, DSA sử dụng các phép toán số học dựa trên số nguyên tố và các giá trị ngẫu nhiên để tạo ra chữ ký số. DSA được thiết kế riêng cho chữ ký số và có hiệu năng tối ưu trong một số ứng dụng, đồng thời cũng đảm bảo tính bảo mật và phi chối bỏ.

1.2 Chứng chỉ số

Chứng chỉ số (Digital Certificate) là một tập hợp thông tin điện tử được cấp bởi Tổ chức Chứng thực (Certificate Authority - CA) nhằm xác thực danh tính của một thực thể, chẳng hạn như cá nhân, tổ chức hoặc thiết bị trong môi trường số. Chứng chỉ số chứa các thông tin quan trọng như tên chủ sở hữu, khóa công khai, số seri, thời hạn hiệu lực, tổ chức cấp chứng chỉ (CA) và các thông tin khác tùy theo tiêu chuẩn áp dụng Cấu trúc chứng chỉ số

Một chứng chỉ số tiêu chuẩn thường được xây dựng theo định dạng X.509 – chuẩn quốc tế phổ biến được sử dụng trên hầu hết các hệ thống bảo mật hiện nay. Cấu trúc của chứng chỉ X.509 bao gồm các thành phần chính sau:

- Version: Xác định phiên bản của chuẩn X.509 mà chứng chỉ tuân theo
- Serial Number: Một số duy nhất được CA cấp để phân biệt chứng chỉ.
- Signature Algorithm: Thuật toán được sử dụng để tạo chữ ký số của chứng chỉ.
- Issuer: Thông tin của CA phát hành chứng chỉ, bao gồm tên tổ chức và các thuộc tính khác.
- Validity: Bao gồm ngày bắt đầu (Not Before) và ngày hết hạn (Not After) của chứng chỉ.
- Subject: Thông tin định danh của chủ sở hữu chứng chỉ (cá nhân, tổ chức hoặc máy chủ).
- Subject Public Key Info: Thông tin về khóa công khai của chủ thể, thường bao gồm loại khóa (RSA, DSA, ECC, ...) và các tham số kỹ thuật.
- Extensions: Các thông tin bổ sung tùy chọn như Key Usage (mục đích sử dụng khóa), Extended Key Usage, CRL Distribution Points, và Authority Information Access.

1.2.1 Quy trình tạo và xác thực chứng chỉ số

Quy trình cấp phát chứng chỉ số thường gồm 4 bước:

- *Tạo yêu cầu cấp chứng chỉ (CSR – Certificate Signing Request):*

Chủ thẻ sinh ra cặp khóa (khóa riêng và khóa công khai) và tạo yêu cầu chứa thông tin định danh cùng với khóa công khai. Yêu cầu này thường được ký số bằng khóa riêng của chủ thẻ để đảm bảo tính xác thực của dữ liệu.

- *Xác thực và phê duyệt:*

CA xác minh danh tính của người yêu cầu qua các phương thức xác thực (kiểm tra giấy tờ, email, hoặc đối chiếu thông tin). Khi xác thực thành công, CA tiến hành ký số yêu cầu để tạo ra chứng chỉ số.

- **Ký số chứng chỉ:**

CA sử dụng khóa riêng của mình để ký số chứng chỉ, tạo ra dấu hiệu số chứng nhận tính hợp lệ và toàn vẹn của chứng chỉ. Dấu hiệu này cho phép bất kỳ ai sau này sử dụng khóa công khai của CA để xác minh chứng chỉ.

- **Phân phối:**

Chứng chỉ số đã ký được gửi lại cho chủ thẻ hoặc được lưu trữ trên kho chứng chỉ công khai, cho phép các bên khác tra cứu và xác minh thông tin khi cần.

1.3 Chữ ký số

Chữ ký số là cơ chế bảo mật dùng để chứng minh tính xác thực của tài liệu và đảm bảo rằng nội dung không bị thay đổi sau khi ký. Quá trình tạo và xác thực chữ ký số thường dựa trên các thuật toán bắt đối xứng, trong đó **RSA** và **DSA** là hai trong số các thuật toán được sử dụng phổ biến.

1.3.1 Quy trình tạo chữ ký số

Các bước tạo chữ ký số bao gồm:

- **Tính hàm băm của tài liệu:**

Sử dụng thuật toán băm (ví dụ: SHA-256, SHA-512 hoặc SHA-3) để tạo ra giá trị băm (digest) đại diện cho toàn bộ nội dung của tài liệu.

Vì đặc tính của hàm băm, bất kỳ thay đổi nhỏ nào cũng làm thay đổi giá trị băm.

- **Mã hóa giá trị băm bằng khóa riêng:**

Người ký sử dụng khóa riêng của mình (theo thuật toán RSA hoặc DSA) để mã hóa giá trị băm, tạo thành chữ ký số.

Ví dụ, đối với RSA, chữ ký số SSS được tính theo công thức: $S = H(m)^{d_{priv}} \pmod{n}$ với $H(m)$ là hàm băm của thông điệp m , d_{priv} là khóa riêng và n là tham số của hệ thống RSA.

Trong khi đó, đối với DSA, quá trình ký số dựa trên các phép toán số học khác nhau theo chuẩn FIPS 186, tạo ra chữ ký số thông qua một cặp giá trị (thường là r và s).

- **Kết hợp chữ ký số với tài liệu:**

Tài liệu được gửi kèm theo chữ ký số và chứng chỉ số của người ký (chứa khóa công khai RSA hoặc DSA), cho phép bên nhận sau này thực hiện việc xác thực.

1.3.2 Quy trình xác thực chữ ký số

Để xác thực chữ ký số, bên nhận thực hiện các bước sau:

- **Giải mã chữ ký số:**

Bên nhận sử dụng khóa công khai (lấy từ chứng chỉ số của người ký) để giải mã chữ ký số, từ đó thu được giá trị hàm băm ban đầu được mã hóa.

Ví dụ, đối với RSA, quá trình này thực hiện theo công thức: $H'm=Se \mod n$ với e là khóa công khai tương ứng.

- *Tính lại hàm băm của tài liệu:*

Sử dụng cùng thuật toán băm (ví dụ: SHA-256) để tính lại giá trị hàm băm $H(m)$ từ tài liệu gốc.

- *So sánh giá trị hàm băm:*

Nếu $H'(m)$ (thu được từ chữ ký số) khớp với $H(m)$ (tính từ tài liệu), điều này xác nhận rằng tài liệu không bị thay đổi và chữ ký số là hợp lệ.

Nếu không khớp, quá trình xác thực thất bại, cho thấy có sự chỉnh sửa hoặc giả mạo dữ liệu.

1.4 Các chuẩn và tiêu chuẩn liên quan

Để đảm bảo tính tương thích và an toàn trong các giao dịch điện tử, chứng chỉ số và chữ ký số cần tuân theo các tiêu chuẩn quốc tế được thiết lập nhằm đảm bảo khả năng xác thực, bảo mật và toàn vẹn dữ liệu. Các tiêu chuẩn này giúp đảm bảo rằng chứng chỉ số có thể được sử dụng rộng rãi trên nhiều hệ thống, nền tảng khác nhau mà vẫn duy trì được độ tin cậy và an toàn cao. Một số chuẩn phổ biến bao gồm:

1.4.1 Chuẩn X.509

- *Định nghĩa:*

X.509 là tiêu chuẩn quốc tế về định dạng chứng chỉ số được sử dụng rộng rãi trong các hệ thống an ninh mạng.

- *Cấu trúc:*

Như đã trình bày, chứng chỉ X.509 bao gồm các thông tin như phiên bản, số sê-ri, thuật toán ký số, thông tin của CA, chủ thẻ, khóa công khai và các trường mở rộng.

- *Ứng dụng:*

X.509 được sử dụng trong SSL/TLS để xác thực máy chủ, thiết lập các kết nối an toàn qua Internet và trong các giao thức khác như S/MIME cho email bảo mật.

1.4.2 Các chuẩn PKCS

Bộ tiêu chuẩn PKCS (Public Key Cryptography Standards) do RSA Security phát hành bao gồm các tiêu chuẩn liên quan đến:

- *PKCS#1:*

Định nghĩa các giao thức và thuật toán liên quan đến RSA, bao gồm các phương thức mã hóa và ký số. Nó cũng đề cập đến các cơ chế padding an toàn như RSA-OAEP (cho mã hóa) và RSA-PSS (cho ký số) nhằm tăng cường bảo mật.

- *PKCS#7:*
Được sử dụng để bao bọc dữ liệu được ký số và mã hóa, thường áp dụng trong các ứng dụng như ký số tài liệu điện tử.
- *PKCS#10:*
Định nghĩa cấu trúc của yêu cầu cấp chứng chỉ (CSR) mà các ứng dụng sử dụng để gửi đến CA.

1.4.3 Các tiêu chuẩn bổ sung và quy định pháp lý

Ngoài X.509 và PKCS, còn có các quy định nhằm đảm bảo hệ thống chứng chỉ số và chữ ký số đạt mức độ tin cậy cần thiết:

- *EU eIDAS:*
Quy định của Liên minh Châu Âu về xác thực điện tử và giao dịch điện tử, phân loại mức độ tin cậy của chứng chỉ số (từ Loại 1 đến Loại 4) và quy định các yêu cầu đối với chữ ký số để được công nhận pháp lý.
- *Luật Giao dịch điện tử (các quốc gia):*
Nhiều quốc gia đã ban hành các quy định công nhận giá trị pháp lý của chữ ký số, đảm bảo rằng giao dịch điện tử được bảo vệ tương đương với giao dịch truyền thống.
- *FIPS (Federal Information Processing Standards):*
Các tiêu chuẩn do chính phủ Hoa Kỳ ban hành. Trong đó, FIPS PUB 186 mô tả DSA – tiêu chuẩn chữ ký số được sử dụng trong một số ứng dụng an ninh chính phủ.

1.5 Các vấn đề bảo mật

Dù chứng chỉ số và chữ ký số là công cụ mạnh mẽ để bảo vệ giao dịch điện tử, chúng vẫn đối mặt với một số rủi ro và điểm yếu cần được khắc phục.

1.5.1 Rủi ro với chứng chỉ số

- *Phân tích và khai thác khóa:*
Nếu kẻ tấn công thu thập được chứng chỉ số cùng với khóa công khai, họ có thể khai thác lỗ hổng trong thuật toán mã hóa hoặc áp dụng các phương pháp tấn công phân tích số học. Việc sử dụng kích thước khóa đủ lớn (ví dụ: 2048-bit trở lên đối với RSA) là cần thiết để làm cho việc phá khóa trở nên không khả thi.
- *Tấn công giả mạo chứng chỉ:*
Kẻ tấn công có thể tạo ra chứng chỉ giả mạo bằng cách lợi dụng lỗ hổng trong quy trình xác thực của CA hoặc xâm nhập vào hệ thống của CA. Vì vậy, bảo vệ khóa riêng của CA và quy trình xác thực chặt chẽ là rất quan trọng.
- *Thu hồi chứng chỉ:*
Nếu chứng chỉ bị xâm phạm hoặc phát hiện lỗi, quá trình thu hồi cần diễn ra nhanh chóng và hiệu quả. Hệ thống CRL hoặc OCSP phải được duy trì liên tục để cập nhật trạng thái chứng chỉ, tránh trường hợp chứng chỉ đã bị thu hồi vẫn được sử dụng.

1.5.2 Rủi ro với chữ ký số

- *Tấn công theo thời gian và tấn công lỗi:*

Các cài đặt của thuật toán ký số, đặc biệt là RSA hoặc DSA, có thể bị tấn công qua việc đo lường thời gian thực hiện hoặc khai thác lỗi trong quá trình tính toán. Việc áp dụng các kỹ thuật làm mờ thời gian (time blinding) và kiểm tra tính nhất quán của phép toán có thể giúp giảm thiểu rủi ro này.

- *Tấn công padding oracle:*

Nếu không áp dụng cơ chế padding an toàn như RSA-OAEP (cho mã hóa) hoặc RSA-PSS (cho ký số), kẻ tấn công có thể khai thác lỗi trong quá trình giải mã để thu thập thông tin về cấu trúc dữ liệu. Việc sử dụng cơ chế padding an toàn là bước quan trọng để phòng chống tấn công này.

1.5.3 Biện pháp khắc phục

- *Tăng cường kích thước khóa:*

Sử dụng khóa có độ dài tối thiểu 2048-bit cho RSA (và các thuật toán tương tự) nhằm đảm bảo an toàn chống lại các cuộc tấn công brute force và phân tích số học.

- *Áp dụng các kỹ thuật padding hiện đại:*

Sử dụng RSA-OAEP cho quá trình mã hóa và RSA-PSS cho quá trình ký số nhằm bảo vệ chống lại các tấn công padding oracle và cải thiện tính bảo mật.

- *Kiểm tra và xác minh liên tục:*

Thiết lập cơ chế giám sát và xác thực liên tục đối với chứng chỉ số và chữ ký số (ví dụ: hệ thống CRL/OCSP và các công cụ kiểm tra tự động).

- *Đào tạo và nâng cao nhận thức:*

Đảm bảo rằng nhà quản trị hệ thống và người dùng cuối được đào tạo về cách sử dụng, lưu trữ và quản lý khóa mật mã, cũng như các biện pháp phòng chống tấn công để giảm nguy cơ mất mát thông tin.

1.6 Kết chương

Chương này đã giới thiệu khái quát về chứng chỉ số, chữ ký số và các công nghệ bảo mật liên quan trong giao dịch điện tử. Các thành phần và quy trình của chứng chỉ số, từ yêu cầu cấp chứng chỉ (CSR) đến quy trình xác thực và phân phối chứng chỉ, đã được trình bày chi tiết. Đồng thời, các thuật toán phổ biến như RSA và DSA được phân tích để giải thích cách thức tạo và xác thực chữ ký số. Chương cũng đã đề cập đến các tiêu chuẩn quốc tế như X.509, PKCS và các quy định pháp lý liên quan nhằm đảm bảo tính bảo mật và tin cậy trong việc sử dụng chứng chỉ số và chữ ký số. Cuối cùng, các vấn đề bảo mật và biện pháp khắc phục, như tăng cường kích thước khóa và áp dụng kỹ thuật padding an toàn, đã được trình bày để nâng cao hiệu quả bảo vệ thông tin trong môi trường số.

CHƯƠNG 2. TÌM HIỂU VỀ KIẾN TRÚC HẠ TẦNG KHÓA CÔNG KHAI PKI

2.1 Khái quát

Cơ sở hạ tầng của khóa công khai viết tắt là PKI (Public Key Infrastructure), PKI là một hệ thống (phần cứng, phần mềm) có nhiệm vụ đảm bảo cho giao dịch điện tử, cho việc trao đổi các thông tin mật, thông qua việc sử dụng các khóa mã và xác thực. PKI cho phép: đảm bảo sự tin cậy, quản lý truy nhập, đảm bảo tính toàn vẹn của thông tin, xác thực người dùng, chống trói bỏ các giao dịch thương mại điện tử và hỗ trợ các ứng dụng công nghệ thông tin. PKI dùng để quản lý việc sinh và phân phối các cặp khóa công khai và bí mật, công bố các khóa công khai (cùng với việc nhận dạng của người dùng) như giâyys chứng nhận người dùng trên các tạp chí nổi tiếng. Khái niệm PKI thường được dùng để chỉ toàn bộ hệ thống bao gồm nhà cung cấp chứng thực số (CA) cùng các cơ chế liên quan, đồng thời với toàn bộ việc sử dụng toàn bộ các thuật toán mật mã khóa công khai trong trại đổi thông tin. Tuy nhiên PKI không nhất thiết sử dụng các thuật toán mã hóa công khai.

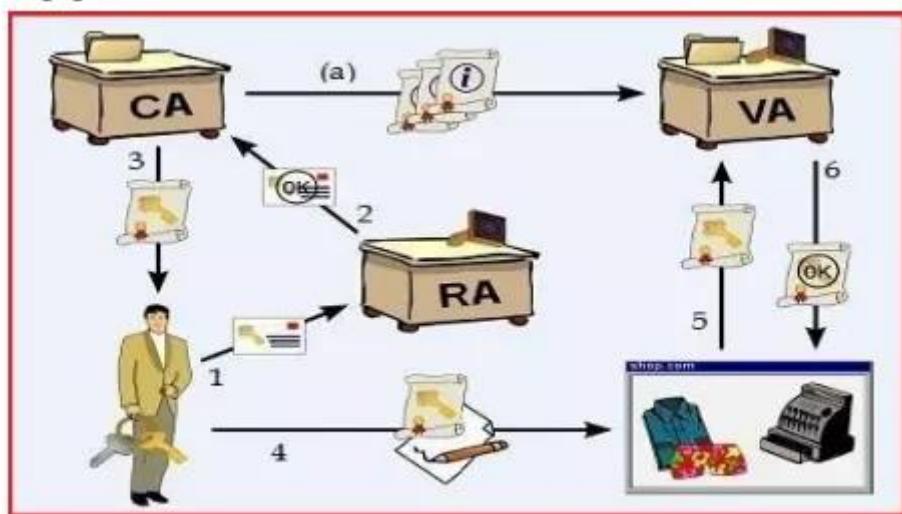
2.2 Kiến trúc hạ tầng khóa công khai PKI

2.2.1 Các thành phần của một hạ tầng cơ sở khóa công khai

PKI là cơ cấu tổ chức gồm con người, tiến trình, chính sách, thủ tục, phần cứng và phần mềm dùng để phát sinh, quản lý, lưu trữ, triển khai và thu hồi các chứng nhận khóa công khai. PKI gồm các thành phần chính sau:

- *Thực thể cuối (End Entity – EE):*
 - Đối tượng sử dụng chứng nhận (chứng thư số): có thể là một tổ chức, một người cụ thể hay một dịch vụ trên máy chủ, ...
- *Tổ chức chứng nhận (Certificate Authority – CA):*
 - Có nhiệm vụ phát hành, quản lý và hủy bỏ các chứng thư số
 - Là thực thể quan trọng trong một PKI mà được thực thể cuối tín nhiệm
 - Gồm tập hợp các con người và các hệ thống máy tính có độ an toàn cao
- *Chứng nhận khóa công khai (Public Key Certificate):*
 - Một chứng nhận khóa công khai thể hiện hay chứng nhận sự ràng buộc của danh tính và khóa công khai của thực thể cuối
 - Chứng nhận khóa công khai chứa đủ thông tin cho những thực thể khác có thể xác nhận hoặc kiểm tra danh tính của chủ nhận chứng nhận đó
 - Định dạng được sử dụng rộng rãi nhất của chứng nhận số dựa trên chuẩn IETF X.509.
- *Tổ chức đăng ký chứng nhận (Registration Authority – RA):* Mục đích chính của RA là để giảm tải công việc của CA
 - Xác thực cá nhân, chủ thể đăng ký chứng thư số.
 - Kiểm tra tính hợp lệ của thông tin do chủ thể cung cấp.

- Xác nhận quyền của chủ thẻ đối với những thuộc tính chứng thư số được yêu cầu.
 - Kiểm tra xem chủ thẻ có thực sự sở hữu khóa riêng đang được đăng ký hay không (chứng minh sở hữu).
 - Tạo cặp khóa bí mật, công khai. (nếu chủ thẻ yêu cầu)
 - Phân phối bí mật được chia sẻ đến thực thể cuối (ví dụ khóa công khai của CA).
 - Thay mặt chủ thẻ thực thể cuối khởi tạo quá trình đăng ký với CA.
 - Lưu trữ khóa riêng.
 - Khởi sinh quá trình khôi phục khóa
 - Phân phối thẻ bài vật lý (thẻ thông minh)
- *Kho lưu trữ chứng nhận (Certificate Repository – CR):*
- Hệ thống (có thể tập trung hoặc phân tán) lưu trữ chứng thư và danh sách các chứng thư bị thu hồi
 - Cung cấp cơ chế phân phối chứng thư và danh sách thu hồi chứng thư (CRLs - Certificate Revocation Lists).



Hình 1 - Quy trình của hạ tầng khóa công khai PKI

- (1): Người dùng gửi yêu cầu phát hành thẻ chứng thư số và khóa công khai của nó đến RA;
- (2): Sau khi xác nhận tính hợp lệ định danh của người dùng thì RA sẽ chuyển yêu cầu này đến CA;
- (3): CA phát hành thẻ chứng thư số cho người dùng;
- (4): Sau đó người dùng “ký” thông điệp trao đổi với thẻ chứng thư số mới vừa nhận được từ CA và sử dụng chúng (thẻ chứng thực số + chữ ký số) trong giao dịch;

- (5): Định danh của người dùng được kiểm tra bởi đối tác thông qua sự hỗ trợ của VA;
- (6): Nếu chứng thư số của người dùng được xác nhận tính hợp lệ thì đối tác mới tin cậy người dùng và có thể bắt đầu quá trình trao đổi thông tin với nó (VA nhận thông tin về thẻ chứng thư số đã được phát hành từ CA (a))

2.2.2 Cách hoạt động của PKI

PKI cung cấp một nền tảng an ninh toàn diện bằng cách sử dụng các cặp khóa bắt đối xứng và chứng chỉ số. Các quá trình tạo khóa, đăng ký và phát hành chứng chỉ, phân phối và lưu trữ chứng chỉ, xác thực danh tính, mã hóa và giải mã dữ liệu, chữ ký số và quản lý vòng đời chứng chỉ hoạt động đồng bộ để đảm bảo an ninh, tính toàn vẹn và xác thực trong các giao dịch và trao đổi thông tin trực tuyến. Về cơ bản, ta có thể thấy quy trình hoạt động của PKI như sau:

- **Người dùng tạo một cặp khóa:**
 - Người dùng tạo một cặp khóa gồm khóa công khai (public key) và một khóa bí mật (private key).
 - Khóa công khai có thể được chia sẻ rộng rãi, trong khi khóa bí mật phải được giữ an toàn và bí mật.
- **Người dùng gửi yêu cầu cấp chứng chỉ đến CA:**
 - Người dùng hoặc thiết bị gửi một yêu cầu chứng chỉ (Certificate Signing Request – CSR) đến một cơ quan chứng thực (Certificate Authority – CA).
 - Yêu cầu này bao gồm khóa công khai và thông tin nhận dạng của người dùng hoặc thiết bị.
- **CA xác minh danh tính của người dùng:** CA xác minh danh tính của người dùng hoặc thiết bị dựa trên thông tin cung cấp trong CSR để đảm bảo rằng họ là chủ sở hữu hợp pháp của khóa công khai.
- **CA cấp chứng chỉ cho người dùng:** Nếu CA xác minh thành công danh tính của người dùng, họ sẽ cấp cho người dùng một chứng chỉ kỹ thuật số. Chứng chỉ này chứa thông tin về danh tính của người dùng và khóa công khai của họ và được ký bởi khóa bí mật của CA.
- **Phân phối và lưu trữ chứng chỉ số**
 - Chứng chỉ số được gửi lại cho người dùng hoặc thiết bị, và có thể được phân phối cho bất kỳ ai cần xác thực danh tính của người dùng hoặc thiết bị đó.
 - Khóa bí mật được giữ kín bởi người dùng hoặc thiết bị và không được chia sẻ với bất kỳ ai khác.
- **Xác thực danh tính**

- Khi cần xác thực danh tính, người dùng hoặc thiết bị sẽ cung cấp chứng chỉ số của mình. Bên nhận sẽ kiểm tra chữ ký số trên chứng chỉ để xác nhận rằng nó được phát hành bởi một CA đáng tin cậy.
- Bên nhận cũng xác minh khóa công khai trong chứng chỉ thuộc về người dùng hoặc thiết bị được xác nhận.
- Người dùng sử dụng khóa công khai để mã hóa thông tin: Khi cần gửi dữ liệu an toàn, bên gửi sử dụng khóa công khai của bên nhận (từ chứng chỉ số của bên nhận) để mã hóa dữ liệu. Chỉ bên nhận mới có thể giải mã dữ liệu này bằng khóa bí mật tương ứng. Người dùng sử dụng khóa công khai của họ để mã hóa thông tin mà họ muốn gửi cho người khác.

Người nhận sử dụng khóa bí mật của người gửi để giải mã thông tin: Người nhận sử dụng khóa bí mật của người gửi (được cung cấp trong chứng chỉ kỹ thuật số) để giải mã thông tin đã được mã hóa.

- Tạo và xác minh chữ ký số
 - Người gửi sử dụng khóa bí mật của mình để ký số tài liệu hoặc thông điệp, đảm bảo rằng dữ liệu không bị thay đổi và xác nhận nguồn gốc của nó.
 - Người nhận sử dụng khóa công khai của người gửi (từ chứng chỉ số của người gửi) để xác minh chữ ký số. Nếu chữ ký số hợp lệ, người nhận có thể tin tưởng rằng dữ liệu không bị thay đổi và thực sự được gửi bởi người sở hữu khóa bí mật tương ứng.
- Lưu ý một số điều sau về vòng đời của chứng chỉ:
 - Chứng chỉ số có thời hạn và cần được gia hạn trước khi hết hạn. Người dùng gửi yêu cầu gia hạn đến CA, và CA sẽ xác minh và phát hành chứng chỉ mới.
 - Nếu khóa bí mật bị lộ hoặc thông tin trong chứng chỉ không còn chính xác, chứng chỉ có thể bị thu hồi. CA duy trì danh sách thu hồi chứng chỉ (Certificate Revocation List – CRL) hoặc sử dụng giao thức OCSP (Online Certificate Status Protocol) để thông báo về chứng chỉ đã bị thu hồi.

2.3 Mục tiêu, chức năng của PKI

PKI cho phép những người tham gia xác thực lẫn nhau và sử dụng thông tin từ các chứng thực khóa công khai để mật mã hóa và giải mã thông tin trong quá trình trao đổi. Thông thường, PKI bao gồm phần mềm máy khách (client), phần mềm máy chủ (server), phần cứng (như thẻ thông minh) và các quy trình hoạt động liên quan. Người sử dụng cũng có thể ký các văn bản điện tử với khóa bí mật của mình và mọi người đều có thể kiểm tra với khóa công khai của người đó. PKI cho phép các giao dịch điện tử được diễn ra đảm bảo tính bí mật, toàn vẹn và xác thực lẫn nhau mà không cần phải trao đổi các thông tin mật từ trước.

- Xác thực: Chứng minh định danh thực thể.
- Bí mật: Đảm bảo rằng không ai có thể đọc được thông báo ngoại trừ người nhận.

- Toàn vẹn: Đảm bảo rằng người nhận sẽ nhận được thông báo mà không bị thay đổi nội
- dung ban đầu.
- Tính chống chối bỏ: Cơ chế này sẽ chứng minh rằng người nhận/gửi đã thực sự gửi/nhận thông báo.

PKI tận dụng cả mật mã đối xứng và bất đối xứng để đạt được những tính năng cơ bản trên.

2.3.1 Xác thực

Về cơ bản, tính xác thực cung cấp 2 khía cạnh ứng dụng chính đó là định danh thực thể và định danh nguồn gốc dữ liệu.

2.3.1.1 Định danh thực thể

Định danh thực thể đơn giản dùng để định danh thực thể xác định nào đó có liên quan. Do đó, trên thực tế, định danh thực thể thông thường sẽ tạo ra một kết quả cụ thể mà sau đó được sử dụng để thực hiện các hoạt động khác hoặc truyền thông khác.

- Định danh thực thể bao gồm: Một nhân tố và nhiều nhân tố.
- Có rất nhiều cách để chứng minh định danh. Ta có thể chia ra làm bốn loại sau:
 - o Cái gì đó người dùng có (ví dụ thẻ thông minh hoặc thiết bị phần cứng).
 - o Cái gì đó người dùng biết (Ví dụ mật khẩu hoặc PIN).
 - o Cái gì đó là người dùng hoặc gắn với người dùng (ví dụ dấu vân tay hoặc vỗng mạc mắt).
 - o Cái gì đó người dùng thực hiện (ví dụ gõ các ký tự nào đó).
- Có hai kiểu xác thực được biết đến như là định danh thực thể, đó là xác thực cục bộ và xác thực từ xa.
 - o Xác thực cục bộ: Xác thực ban đầu của một thực thể tới môi trường cục bộ hầu như liên quan trực tiếp tới người dùng. Ví dụ như mật khẩu hoặc số định danh cá nhân (PIN) phải được nhập vào, sử dụng dấu vân tay để nhận dạng,...
 - o Xác thực từ xa (Xác thực của một thực thể tới môi trường ở xa): Nghĩa là có thể hoặc không cần liên quan trực tiếp tới người dùng. Trên thực tế, hầu hết các hệ thống xác thực từ xa phức tạp không hoàn toàn liên quan tới người dùng vì rất khó để bảo vệ hệ thống xác thực mà đưa ra các thông tin xác thực nhạy cảm, ví dụ như mật khẩu hoặc dấu vân tay, và truyền trên một kenh không an toàn.

2.3.1.2 Định danh nguồn gốc dữ liệu

Định danh nguồn gốc dữ liệu sẽ định danh một thực thể xác định nào đó như nguồn gốc của dữ liệu được đưa ra. Hoạt động định danh này không phải là định danh cô lập, cũng không phải hoàn toàn là định danh cho mục đích thực hiện các hoạt động khác.

2.3.2 Bí mật

Dịch vụ bí mật đảm bảo tính riêng tư của dữ liệu. Không ai có thể đọc được dữ liệu ngoại trừ thực thể nhận. Dịch vụ bí mật được yêu cầu khi dữ liệu khi:

- Được lưu trữ trên phương tiện (như phần cứng máy tính) mà người dùng không hợp pháp có thể đọc được.
- Được dự phòng trên thiết bị (ví dụ băng từ) mà có thể bị rơi vào tay người dùng không hợp pháp.
- Được truyền trên mạng không được bảo vệ

Các kỹ thuật mã hóa đảm bảo tính bí mật cần phải được áp dụng với mọi loại dữ liệu nhạy cảm.

2.3.3 Toàn vẹn dữ liệu

Toàn vẹn dữ liệu đảm bảo rằng dữ liệu không bị thay đổi. Sự đảm bảo này là một phần thiết yếu trong bất kỳ môi trường thương mại điện tử hoặc loại hình kinh doanh nào.

Mức độ toàn vẹn dữ liệu có thể đạt được bằng các cơ chế chẵn lẻ của các bit và mã kiểm tra dịch vòng (Cyclic Redundancy Codes - CRCs).

Để bảo vệ dữ liệu khỏi tấn công nhằm phá vỡ tính toàn vẹn dữ liệu, các kỹ thuật mã hóa được sử dụng. Do đó, khoá và các thuật toán phải được triển khai và phải được biết giữa các thực thể muốn cung cấp tính toàn vẹn dữ liệu với thực thể muốn được đảm bảo tính toàn vẹn của dữ liệu.

Dịch vụ toàn vẹn của PKI có thể được xây dựng dựa trên hai kỹ thuật:

- Chữ ký số: Mặc dù nó được dùng cho mục đích cung cấp sự xác thực, nhưng nó cũng được sử dụng để cung cấp tính toàn vẹn cho dữ liệu được ký. Nếu có sự thay đổi bất kỳ trước và sau khi ký thì chữ ký số sẽ bị loại bỏ khi kiểm tra, vì vậy việc mất tính toàn vẹn của dữ liệu sẽ dễ dàng bị phát hiện.
- Mã xác thực thông báo: Kỹ thuật này thông thường sử dụng một mã khối đối xứng (ví dụ DES, DES-CBC-MAC) hoặc một hàm băm mật mã (HMAC-SHA-1)

2.3.4 Chống chối bỏ

Dịch vụ chống chối bỏ là dịch vụ đảm bảo rằng thực thể không thể chối bỏ hành động của mình. Các biến thể thường được nhắc tới nhiều nhất là chống chối bỏ nguồn gốc (người dùng không thể chối bỏ rằng đã gửi một tài liệu hoặc một văn bản) hoặc chối bỏ sự tiếp nhận (người dùng không thể chối bỏ rằng đã nhận được văn bản hoặc tài liệu).

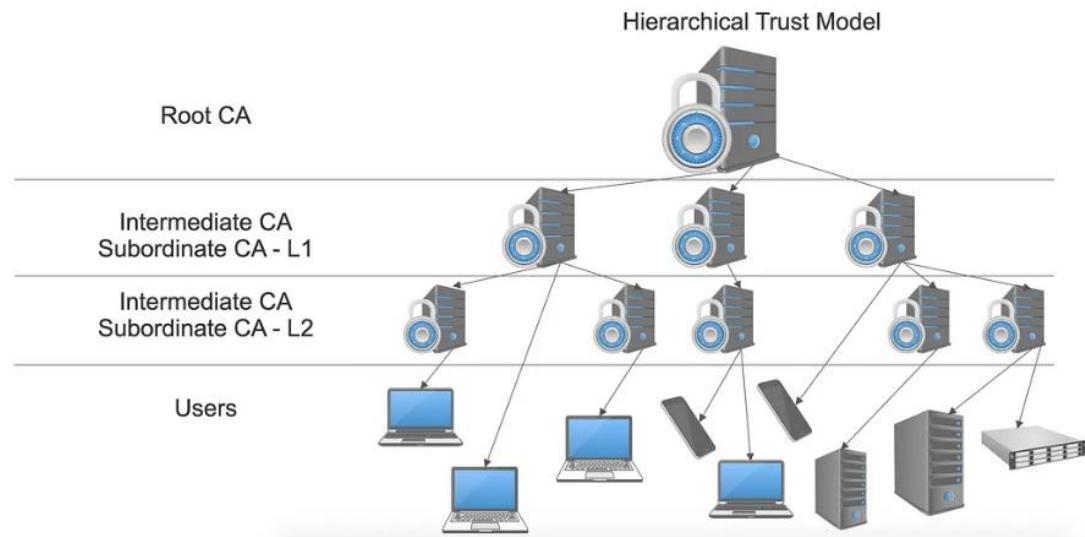
Một vài các biến thể khác của tính chống chối bỏ là: Chối bỏ đã tạo ra, chối bỏ đã chuyển, chối bỏ việc tán thành.

2.4 Mô hình PKI

2.4.1 Mô hình tin cậy phân cấp (Hierarchical Trust Model)

Mô hình phân cấp hoặc mô hình cây là mô hình phổ biến nhất để triển khai hạ tầng khóa công khai (PKI). Một CA gốc (Root CA) ở cấp cao nhất cung cấp tất cả thông tin, và các

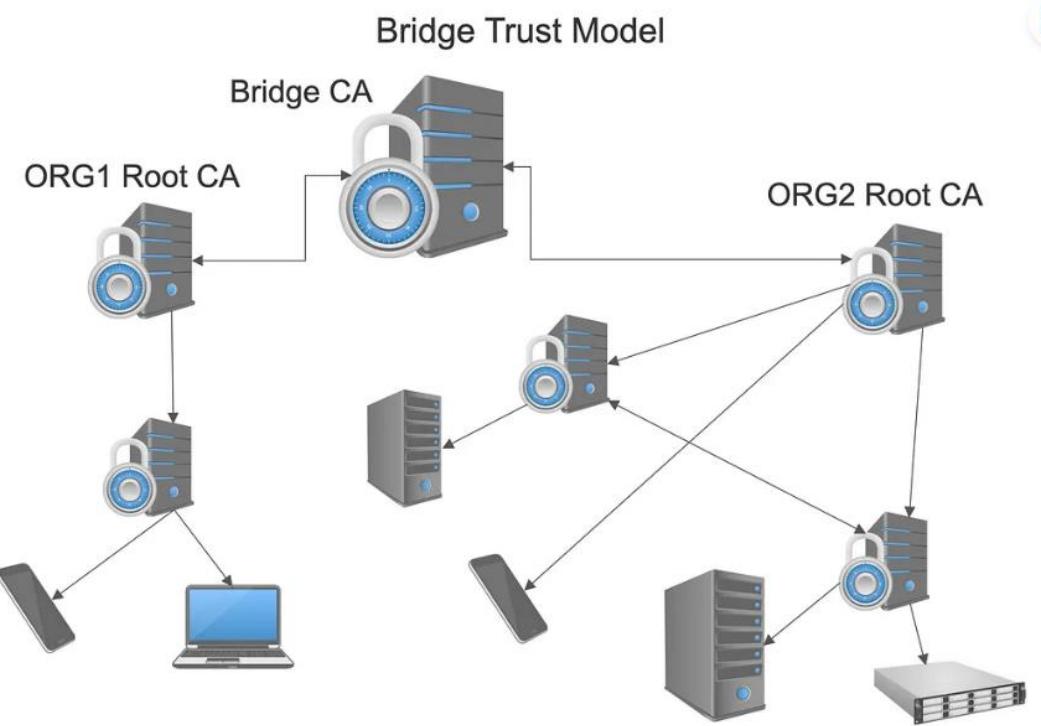
CA trung gian (Intermediate CAs) nằm ở cấp tiếp theo trong hệ thống phân cấp, chỉ tin tưởng vào thông tin do CA gốc cung cấp. CA gốc cũng chỉ tin tưởng các CA trung gian thuộc cùng hệ thống phân cấp. Cách sắp xếp này cho phép kiểm soát chặt chẽ ở tất cả các cấp trong hệ thống phân cấp. Đây có thể là mô hình phổ biến nhất trong các tổ chức lớn muốn mở rộng khả năng xử lý chứng chỉ. Mô hình phân cấp cho phép kiểm soát chặt chẽ các hoạt động liên quan đến chứng chỉ.



Hình 2 - Mô hình tin cậy phân cấp

2.4.2 Mô hình tin cậy cầu nối (Bridge Trust Model)

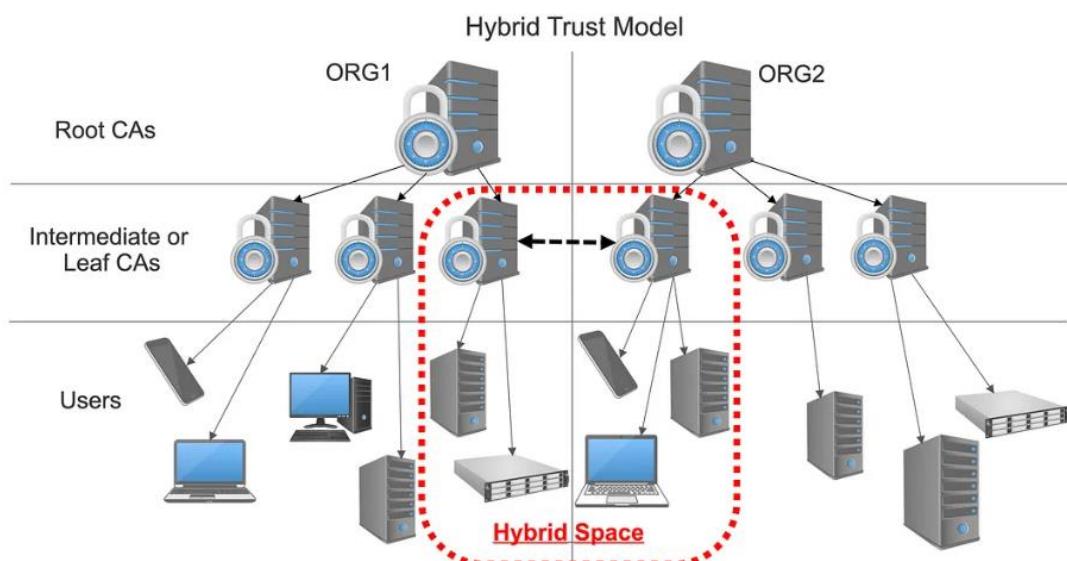
Trong mô hình tin cậy cầu nối, có nhiều mối quan hệ ngang hàng (P2P) giữa các CA gốc, cho phép các CA gốc giao tiếp với nhau và cấp chứng chỉ chéo (cross-certificates). Mô hình này cho phép thiết lập quy trình chứng thực giữa các tổ chức (hoặc các phòng ban). Trong mô hình này, mỗi CA trung gian chỉ tin tưởng vào các CA ở cấp trên và cấp dưới của nó, nhưng cấu trúc CA có thể được mở rộng mà không cần tạo thêm các lớp CA mới. Lợi ích chính của mô hình cầu nối là tính linh hoạt và khả năng tương tác cao giữa các tổ chức.



Hình 3 - Mô hình tin cậy cầu nối

2.4.3 Mô hình tin cậy lai (Hybrid Trust Model)

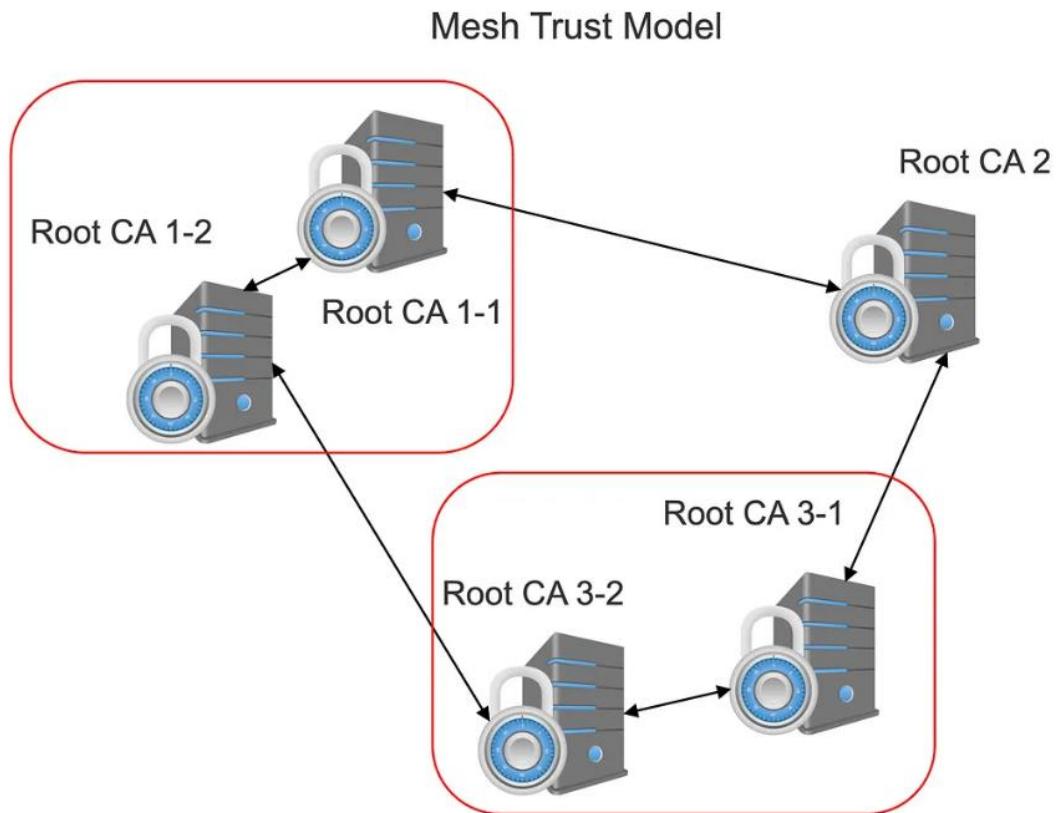
Đôi khi, bạn cần liên kết hai hoặc nhiều tổ chức hoặc phòng ban trong một số phần nhất định nhưng vẫn muốn tách biệt các phần khác. Khi bạn cần thiết lập sự tin cậy ở một số bộ phận của hai tổ chức nhưng không muốn mở rộng sự tin cậy này đến các phân đoạn khác, mô hình tin cậy lai là lựa chọn phù hợp. Bạn có thể cực kỳ linh hoạt khi xây dựng cấu trúc tin cậy lai, và tính linh hoạt này cũng cho phép bạn tạo ra các môi trường kết hợp. Lưu ý rằng trong cấu trúc này, các CA trung gian nằm ngoài môi trường lai chỉ tin tưởng vào CA gốc trực tiếp của chúng, trong khi các CA trung gian trong môi trường lai có thể tin tưởng vào tất cả các CA gốc được kết nối với bất kỳ CA trung gian nào trong môi trường lai



Hình 4- Mô hình tin cậy lai

2.4.4 Mô hình tin cậy dạng lưới (Mesh Trust Model)

Khi bạn muốn triển khai mô hình tin cậy phân cấp nhưng có kiểm tra cấp chứng chỉ chéo hoặc có một mạng lưới các CA gốc, mô hình tin cậy dạng lưới là lựa chọn tốt nhất. Mô hình này kế thừa các khái niệm của cấu trúc cầu nối nhưng có nhiều đường kết nối (multi-paths) và nhiều CA gốc (multi RootCAs). Các chứng chỉ trong mỗi CA gốc được ủy quyền cho tất cả các CA gốc, CA trung gian, và CA nhánh, cũng như tất cả người dùng cuối được kết nối với bất kỳ chuỗi CA nào.

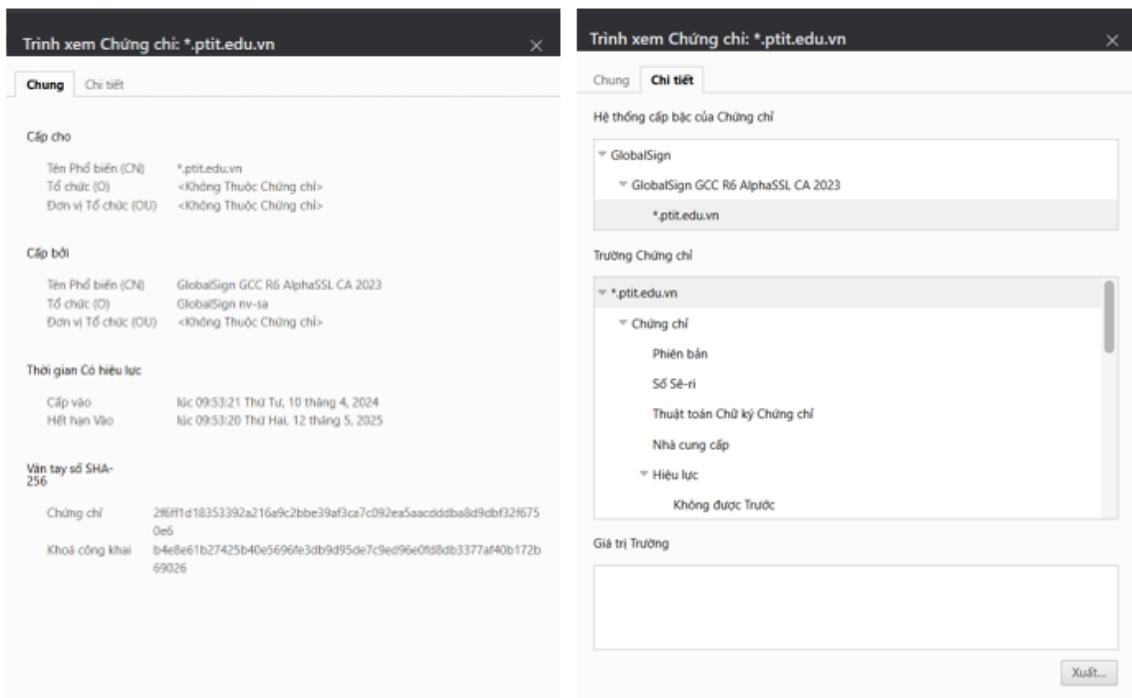


Hình 5- Mô hình tin cậy dạng lưới

2.5 Ứng dụng của hạ tầng PKI

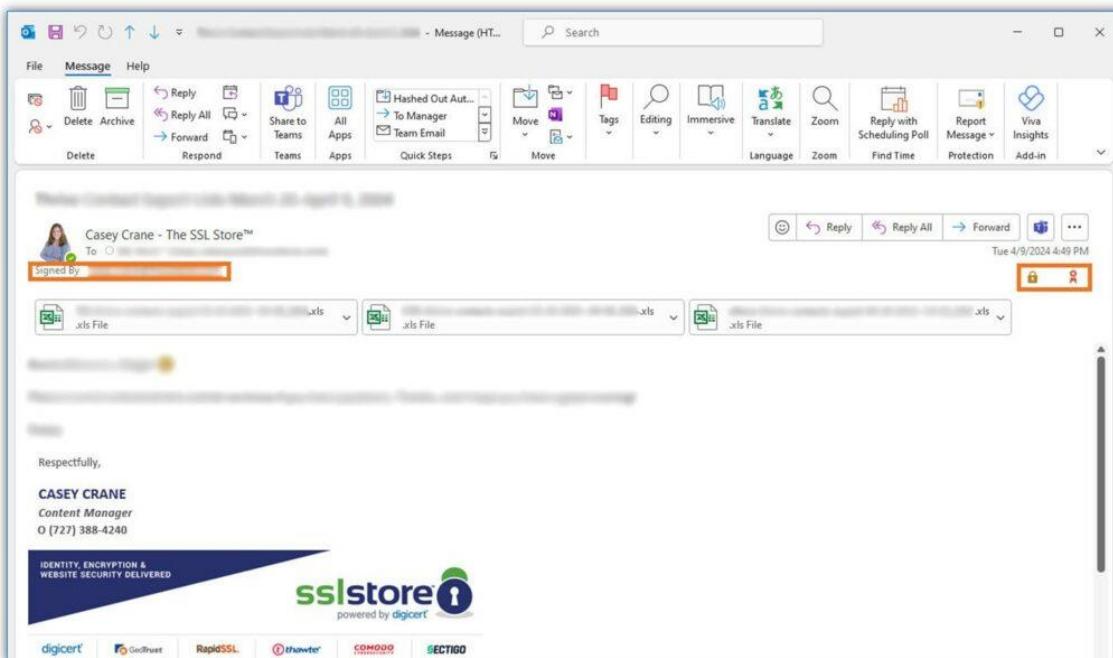
PKI (Public Key Infrastructure) được sử dụng rộng rãi trong nhiều lĩnh vực để bảo mật thông tin, xác thực danh tính và đảm bảo tính toàn vẹn của dữ liệu. Dưới đây là một số ứng dụng quan trọng của PKI trong thực tế:

- Bảo mật giao dịch và giao tiếp trong Internet
 - o PKI đóng vai trò quan trọng trong việc bảo mật các trang web bằng giao thức HTTPS. Cụ thể PKI dùng trong các chứng chỉ SSL/TLS mã hóa dữ liệu truyền tải giữa trình duyệt và máy chủ. Từ đó ngăn chặn kẻ tấn công có thể khai thác MITM vào dữ liệu được truyền đi. Ngoài ra nó cũng giúp xác thực danh tính của trang web, định danh, cung cấp thông tin chi tiết về tổ chức được xác minh.



Hình 6 - Chứng chỉ số xác thực tên miền

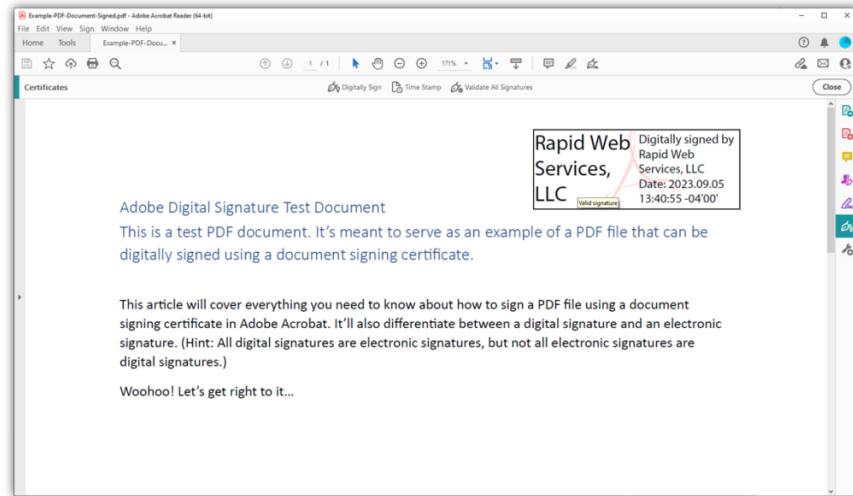
- Chứng chỉ S/MIME (Secure/Multipurpose Internet Mail Extensions) sử dụng PKI để mã hóa email, giúp đảm bảo chỉ người nhận hợp lệ mới có thể đọc được nội dung cũng như xác thực nguồn gốc email, chống giả mạo email lừa đảo (phishing).



Hình 7 - Email đã được mã hóa và kí số

- Chứng thực tài liệu điện tử: Hợp đồng và tài liệu của tổ chức rất quan trọng và không ai muốn nó bị can thiệp hay sửa đổi. Chính vì vậy các tổ chức, doanh

nghiệp hay chính phủ sử dụng chữ ký số để cấp chứng chỉ, hợp đồng, hóa đơn điện tử giúp giảm thiểu gian lận và tăng tính pháp lý cho tài liệu số.



Hình 8 - Ví dụ về PDF có chữ ký số

- Xác thực và kiểm soát truy cập

- Xác thực người dùng, thiết bị mà không dùng mật khẩu: Việc rò rỉ mật khẩu do quản lý bí mật và cấu hình bảo mật kém đã không còn là chuyện hiếm nữa. Chính vì vậy thay vì sử dụng phương thức sử dụng tài khoản mật khẩu để đăng nhập vào hệ thống, PKI có thể cung cấp khả năng đăng nhập không cần mật khẩu do PKI nội bộ xử lý để chỉ những người dùng, những thiết bị được xác minh và trao quyền mới có thể truy cập được vào hệ thống nội bộ.
- Xác thực VPN: xác thực VPN dựa trên PKI cho phép người dùng được ủy quyền kết nối với mạng riêng ảo của bạn mà không tiết lộ thông tin đăng nhập trong khi có găng tạo kết nối an toàn, được mã hóa.



Hình 9 - Chứng chỉ máy khách khi kết nối với VPN

- Xác thực mạng không dây: Hệ thống truy cập Wifi dựa trên PKI cho phép người dùng được ủy quyền chứng minh danh tính của mình mà không cần nhập mật khẩu, giúp họ có thể truy cập Wifi an toàn trong khi khóa riêng của chứng chỉ được lưu trữ an toàn trong thiết bị. Qua đó tránh nguy cơ bị tấn công MITM hay Evil Twin.

Ngoài ra còn rất nhiều ứng dụng khác về PKI trong xác thực giao dịch ngân hàng điện tử, các hệ thống IOT hay dịch vụ công trực tuyến, ...

2.6 Kết chương

Chương này đã trình bày chi tiết về cơ sở hạ tầng khóa công khai (PKI), bao gồm các thành phần, kiến trúc và cách thức hoạt động của hệ thống PKI. Các mô hình tin cậy khác nhau, như mô hình phân cấp, cầu nối, lai và dạng lưới, đã được giải thích để cho thấy sự linh hoạt trong việc triển khai và quản lý PKI trong các tổ chức và hệ thống. Đồng thời, chương cũng đề cập đến các mục tiêu và chức năng của PKI, bao gồm xác thực, bảo mật, toàn vẹn dữ liệu và chống chối bỏ. Các mô hình PKI đã được minh họa để cho thấy cách thức tổ chức và cấu trúc tin cậy trong các hệ thống bảo mật hiện nay.

CHƯƠNG 3. DEMO THỬ NGHIỆM HỆ THỐNG QUẢN LÝ CHỨNG CHỈ SỐ EJBCA CE

3.1 Giới thiệu hệ thống quản lý chứng chỉ số EJBCA CE

3.1.1 Tổng quan hệ thống chứng chỉ số EJBCA CE

EJBCA là một sản phẩm mã nguồn mở vô cùng mạnh mẽ và linh hoạt, được phát triển bởi Primekey. Đây là một hệ thống Certification Authority (CA) được xây dựng dựa trên công nghệ Java J2EE, mang lại hiệu suất hoạt động vượt trội cũng như khả năng tùy biến cao, giúp đáp ứng các yêu cầu phức tạp của các tổ chức và doanh nghiệp. So với các hệ thống mã nguồn mở khác, EJBCA không chỉ nổi bật về mặt hiệu năng mà còn về tính linh hoạt trong việc cấu hình và mở rộng các tính năng.

Hệ thống EJBCA được thiết kế để cung cấp một nền tảng vững chắc cho việc triển khai các hệ thống Public Key Infrastructure (PKI), với một loạt các thành phần hữu ích như OCSP (Online Certificate Status Protocol), RA Service (Registration Authority Service), và Publisher. Những tính năng này giúp cấu thành một hệ thống PKI hoàn chỉnh, từ đó hỗ trợ các tổ chức trong việc triển khai và quản lý các chứng chỉ số, bảo mật giao dịch trực tuyến, và nâng cao mức độ an toàn cho các hệ thống thông tin quan trọng.

Với EJBCA, các doanh nghiệp có thể hoàn toàn yên tâm về một giải pháp toàn diện, bảo mật, và dễ dàng tùy chỉnh để phục vụ các mục tiêu và yêu cầu riêng biệt của mình trong việc quản lý chứng chỉ số và xây dựng các hạ tầng bảo mật đáng tin cậy.

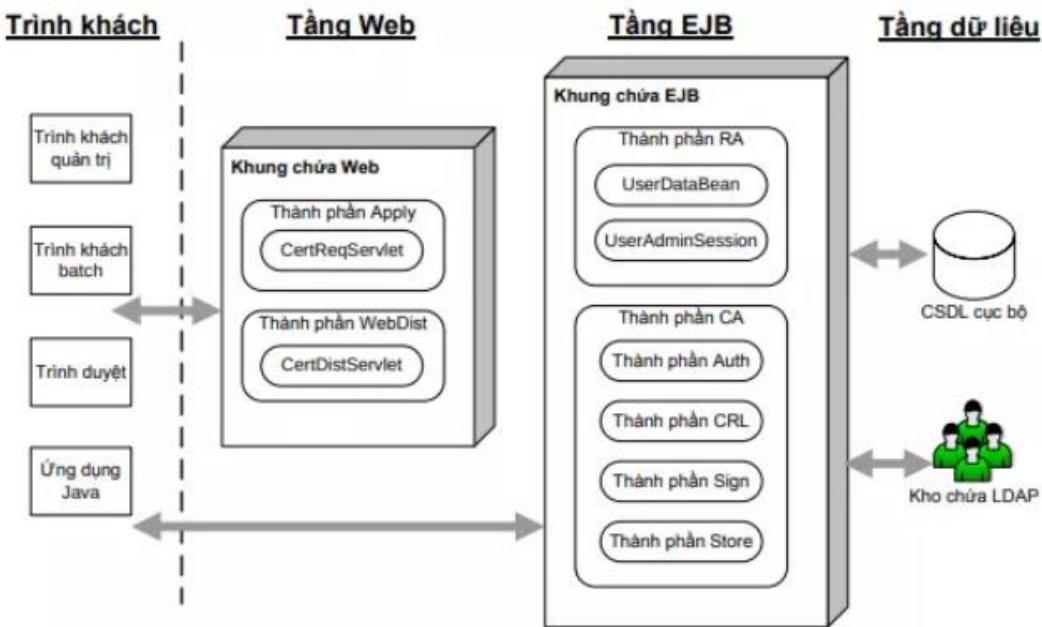
3.1.2 Đặc điểm kỹ thuật

Được xây dựng dựa trên Java, EJBCA thực sự là một nền tảng độc lập, chạy trên hầu như toàn bộ các phần cứng phổ biến cũng như các hệ điều hành thông dụng như Windows, Linux. Để có thể hoạt động, EJBCA cần chạy trên một nền tảng máy chủ ứng dụng (Application Server) cũng như một hệ thống Cơ sở dữ liệu nhất định. Về mặt này, EJBCA cũng hỗ trợ hầu hết các nền tảng App Server phổ biến hiện nay như: JBOSS – Oracle Weblogic – IBM Web Sphere... cũng như các hệ cơ sở dữ liệu từ miễn phí đến trả phí: MySQL, Oracle, IBM DB2, MS SQL, ...

Bên cạnh đó, EJBCA còn có một số đặc điểm đặc trưng sau:

- Cung cấp khả năng xây dựng CA theo nhiều mức, không giới hạn số lượng CA;
- Hỗ trợ thuật toán RSA với độ dài khóa lên tới 4096 bits;
- Hỗ trợ các thuật toán DSA với độ dài khóa lên tới 1024 bits;
- Hỗ trợ các hàm băm như MD5, SHA-1, SHA-256;
- Chứng thư được phát hành tuân thủ nghiêm ngặt chuẩn X509.

3.1.3 Kiến trúc



Hình 10– Kiến trúc EJBCA CE

- Tầng dữ liệu (Data Tier): Tầng dữ liệu lưu trữ các chứng nhận, CRL cũng như các thực thể cuối. EJBCA sử dụng một cơ sở dữ liệu mặc định để lưu trữ các thực thể cuối. Các chứng nhận được lưu trữ trong một kho chứa LDAP (Lightweight Directory Access Protocol).
- Thành phần CA: Thành phần có chức năng tạo các CA gốc, CA con, chứng nhận, CRL và giao tiếp với kho chứa LDAP để lưu trữ thông tin chứng nhận.
- Thành phần RA: Thành phần có chức năng tạo, xóa và hủy bỏ người dùng. Nó giao tiếp với cơ sở dữ liệu cục bộ để chứa thông tin người dùng.
- Tầng Web: Đây là giao diện (điển hình là giao diện người – máy bằng đồ họa) để trình khách tương tác với hệ thống EJBCA, đồng thời quy định các cấp độ và phạm vi truy cập thông tin khác nhau cho thực thể cuối.
- Trình khách: Trình khách là thực thể cuối hay người sử dụng như trình khách thu điện tử, máy chủ web, trình duyệt web hay cổng VPN. Các thực thể cuối không được phép phát hành chứng nhận đến các thực thể khác, nói cách khác chúng là các nút lá trong PKI.

3.2 Demo hệ thống quản lý chứng chỉ số EJBCA CE

3.2.1 Cài đặt EJBCA CE

3.2.1.1 Chuẩn bị

- máy Ubuntu 22.4 LTS (phiên bản mới nhất)
- Docker

3.2.1.2 Cài đặt hệ thống

- Cài docker bản mới nhất với quyền sudo

```
levuong@ubuntu:~$ sudo snap install docker
docker 27.2.0 from Canonical✓ installed
```

Hình 11– Cài docker trên máy

- Cài EJBCA với yêu cầu hệ thống:
 - o ít nhất hai lõi CPU
 - o ít nhất 1GB RAM
- Vùng chứa Docker EJBCA có thể được kéo thằng từ dòng lệnh bằng công cụ docker. Để tải xuống và giải nén hình ảnh vùng chứa Cộng đồng EJBCA mới nhất từ Docker Hub, hãy sử dụng lệnh sau:

```
levuong@ubuntu:~$ sudo docker pull keyfactor/ejbca-ce
Using default tag: latest
latest: Pulling from keyfactor/ejbca-ce
4585bcacf7fc: Pull complete
d3d0222e6ec1: Pull complete
4fc2fd70e35c: Pull complete
aafcccd09449d: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:03b12eef26f136c16af1608f9735202081c80f38d5e2aff5b431de96f06ab1ca
Status: Downloaded newer image for keyfactor/ejbca-ce:latest
docker.io/keyfactor/ejbca-ce:latest
```

Hình 12 - Cài EJBCA CE qua docker

- Sau đó bắt đầu với một phiên bản mà nơi bất kỳ ai có quyền truy cập mạng chưa được xác thực vào phiên bản đều có thể quản lý hệ thống:

```
levuong@ubuntu:~$ sudo docker run -it --rm -p 80:8080 -p 443:8443 -e TLS_SETUP_ENABLED="simple" keyfactor/ejbca-ce
2025-02-21 12:59:14,713+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Configuring logging for Application Server
2025-02-21 12:59:14,735+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Configuring logging for ejbca
2025-02-21 12:59:14,795+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) uid=10001 gid=0(root) groups=0()
2025-02-21 12:59:15,129+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Detected 2 available core(s).
2025-02-21 12:59:15,152+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Detected 4064296960 bytes available host memory.
2025-02-21 12:59:15,180+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Observable at 127.0.0.1:8090 under paths: /health /health/ready /health/live
2025-02-21 12:59:15,213+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) LOG_LEVEL_APP_OCSP_TRANSACTIONS setting is deprecated and does nothing
2025-02-21 12:59:15,221+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) LOG_LEVEL_APP_OCSP_AUDIT setting is deprecated and does nothing
2025-02-21 12:59:15,246+0000 WARN [/opt/keyfactor/bin/start.sh] (process:1) Using the H2 in memory database which is only suitable for ephemeral testing.
tables:
DATABASE_JDBC_URL
  jdbc:mariadb://database:3306/ejbca?characterEncoding=utf8
  jdbc:postgresql://database/ejbca
DATABASE_USER
  ejbca
DATABASE_PASSWORD

2025-02-21 12:59:15,271+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Looking for plugins to import and initialize under /opt/keyfactor/ejbca/plugins
2025-02-21 12:59:15,294+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) External hostname env.HTTPSERVER_HOSTNAME is set to 'localhost'.
2025-02-21 12:59:15,310+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Cluster Node ID is set to 'urifv1l5cmows0fkp'.
2025-02-21 12:59:15,336+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Using provided CLI username and secret
2025-02-21 12:59:15,381+0000 INFO [/opt/keyfactor/bin/start.sh] (process:1) Starting application server:
=====
JBoss Bootstrap Environment
JBOSS_HOME: /opt/keyfactor/wildfly-33.0.0.Final
JAVA: /usr/lib/jvm/java-17-slim/bin/java
```

Hình 13 - Chạy EJBCA CE trên localhost với cổng 8080

- Khi hệ thống được khởi động hoàn toàn, hãy truy cập GUI quản trị EJBCA nơi có thể quản lý phiên bản. Trong menu của GUI quản trị EJBCA, bạn sẽ tìm thấy một liên kết: <https://mycahostname:443/ejbca/adminweb/>

Name	Type	Library	Reference Type	Reference	Active	Auto-activation	Used	Actions
ManagementCA	Soft						Yes	Reactivate Delete

Create new... © 2002–2024. EJBCA® is a registered trademark.

Hình 14 - Giao diện hệ thống EJBCA CE

3.2.2 Một số chức năng của EJBCA CE

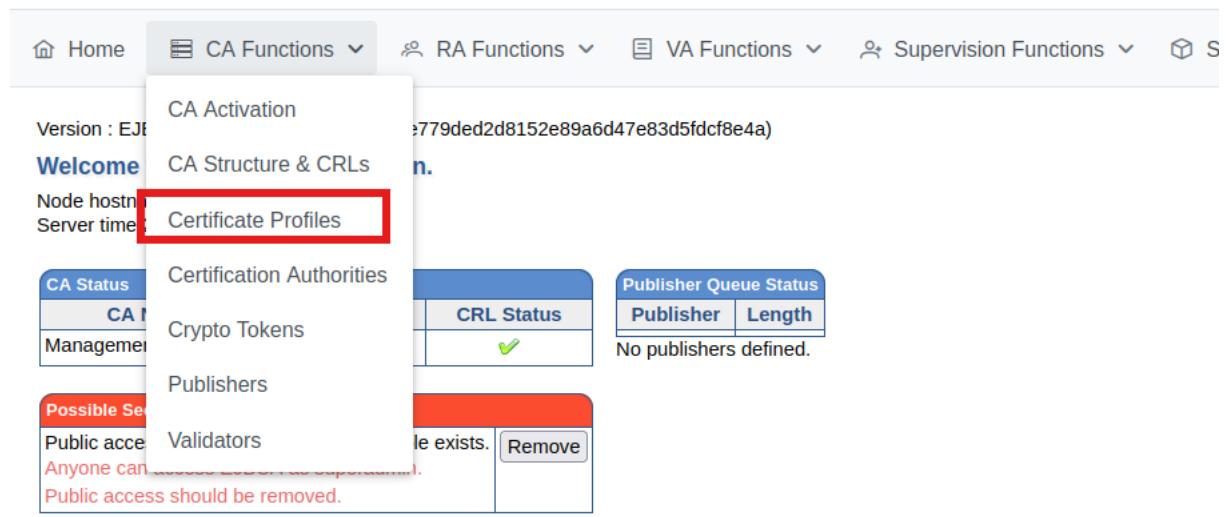
3.2.2.1 Cấp quyền root CA với chứng chỉ

(a) Tạo hồ sơ chứng chỉ

Bước đầu tiên trong quá trình tạo CA là tạo một hồ sơ chứng chỉ. Hồ sơ chứng chỉ định nghĩa các ràng buộc của chứng chỉ mới, ví dụ như các khóa có thể sử dụng và các phần mở rộng sẽ có. Để tìm hiểu thêm về hồ sơ chứng chỉ, bạn có thể tham khảo phần Tổng quan về Hồ sơ Chứng chỉ.

Để tạo một hồ sơ chứng chỉ làm mẫu cho việc tạo CA, chúng ta làm theo các bước sau:

- Trong EJBCA, dưới chức năng CA, nhấp vào *Certificate Profiles*.



The screenshot shows the EJBCA web interface. At the top, there is a navigation bar with links for Home, RA Functions, VA Functions, Supervision Functions, and a search bar. Below the navigation bar, there is a sidebar with sections for CA Status, Management, Possible Servers, and Public access. The 'Public access' section contains a warning message: 'Anyone can access this CA. Public access should be removed.' A red box highlights the 'CA Functions' dropdown menu. Inside the dropdown, the 'Certificate Profiles' option is also highlighted with a red box. To the right of the dropdown, there are sections for Certification Authorities (showing one entry), CRL Status (green checkmark), and Publisher Queue Status (no publishers defined). At the bottom right of the page, there is a copyright notice: '© 2002–2024. EJBCA® is a registered trademark of Keyfactor AB.'

Hình 15 - Tạo hồ sơ chứng chỉ

- Trang Quản lý Hồ sơ Chứng chỉ sẽ hiển thị một danh sách các hồ sơ mặc định.
- Nhập vào *Clone* bên cạnh mẫu ROOTCA mặc định để tạo một hồ sơ mới từ mẫu đó.

Manage Certificate Profiles

List of Certificate Profiles

Name	Actions					
ENDUSER	View	Edit	Delete	Rename	Clone	Export
OCSPSIGNER	View	Edit	Delete	Rename	Clone	Export
ROOTCA	View	Edit	Delete	Rename	Clone	Export
SERVER	View	Edit	Delete	Rename	Clone	Export

Hình 16 - Clone ROOTCA về làm mẫu

- Đặt tên cho hồ sơ chứng chỉ mới là *MyRootCAProfile*, và nhập vào *Create from template*.
- Để chỉnh sửa các giá trị mặc định của hồ sơ Root CA sao cho phù hợp với nhu cầu của bạn, tìm hồ sơ *MyRootCAProfile* mới tạo trong danh sách và nhấp vào *Edit*.
- Trong trang chỉnh sửa, xác nhận rằng loại hồ sơ là **Root CA** và cập nhật các thông số sau:

Available Key Algorithms	ECDSA RSA Ed25519 Ed448 FALCON-512 FALCON-1024 KYBER512 KYBER768 KYBER1024 DILITHIUM2 DILITHIUM3 DILITHIUM5
Available ECDSA curves	No elliptic curve algorithm with selectable curves selected.
Available Bit Lengths	1024 bits 1536 bits 2048 bits 3072 bits 4096 bits
Signature Algorithm	Inherit from issuing CA
Alternative Signature	<input type="checkbox"/> Use
Validity or end date of the certificate	30y

Hình 17 - Chính sửa thuật toán và thời hạn của chứng chỉ

- Dưới phần X.509v3 extensions, bỏ chọn các mục sau:

X.509v3 extensions	
Basic Constraints	<input checked="" type="checkbox"/> Use... <input checked="" type="checkbox"/> Critical
Path Length Constraint	<input type="checkbox"/> Add... Value 0
Authority Key ID	<input type="checkbox"/> Use
Subject Key ID	<input checked="" type="checkbox"/> Use <input type="checkbox"/> Truncated KeyID (method 2 in RFC5280 which is uncommon, keep unchecked for most use cases)
X.509v3 extensions Usages	
Key Usage	<input checked="" type="checkbox"/> Use... <input checked="" type="checkbox"/> Critical <input type="checkbox"/> Forbid encryption usage for ECC keys Key Usage: <input checked="" type="checkbox"/> Digital Signature <input type="checkbox"/> Data encipherment <input checked="" type="checkbox"/> CRL sign <input type="checkbox"/> Non-repudiation <input type="checkbox"/> Key agreement <input type="checkbox"/> Encipher only <input type="checkbox"/> Key encipherment <input checked="" type="checkbox"/> Key certificate sign <input type="checkbox"/> Decipher only
Extended Key Usage	<input type="checkbox"/> Use... <input type="checkbox"/> Critical
Certificate Policies	<input type="checkbox"/> Use... <input type="checkbox"/> Critical
X.509v3 extensions Names	
Subject Alternative Name	<input type="checkbox"/> Use... <input type="checkbox"/> Critical <input checked="" type="checkbox"/> Search enabled (search enabled SAN use more storage)
Issuer Alternative Name	<input type="checkbox"/> Use... <input type="checkbox"/> Critical
Subject Directory Attributes	<input type="checkbox"/> Use
Name Constraints	<input type="checkbox"/> Use... <input type="checkbox"/> Critical

Hình 18 - Bỏ chọn các cấu hình không cần thiết vì là ROOT CA

Dưới phần *Other Data*, tắt *LDAP DN order* vì đây là một phương thức đặt tên cũ cho thành phần *DISTINGUISHED NAME*, không được khuyến khích sử dụng.

The screenshot shows the 'Other Data' configuration screen in EJBCA. The 'LDAP DN order' section has its 'Use' checkbox unchecked. The 'Save' button at the bottom is highlighted with a red box.

Hình 19 - Bỏ LDAP DN và lưu chứng chỉ

- Để lưu hồ sơ chứng chỉ, nhập vào *Save*.

Hồ sơ **MyRootCAProfile** mới tạo sẽ được hiển thị trong danh sách hồ sơ chứng chỉ.

(b) Tạo crypto token

Trong EJBCA, các khóa mã hóa được lưu trữ trong một *crypto token*. Một *crypto token* có thể được lưu trữ trong cơ sở dữ liệu, gọi là *soft keystore*, hoặc trên một *Hardware Security Module (HSM)*.

Các bước sau sẽ mô tả cách tạo một *soft crypto token* và các khóa cần thiết:

- *Sign key*: Dùng để ký số từ CA.
- *Default key*: Dùng cho bất kỳ mã hóa nào mà CA cần thực hiện.
- *Test key*: Thường chỉ dùng trong các bài kiểm tra sức khỏe hoặc dịch vụ duy trì kết nối của HSM.

(Lưu ý: Là một thói quen tốt, bạn nên đánh số các khóa ký và mã hóa (default) khi tạo chúng để dễ dàng tham chiếu trong suốt thời gian sống của chứng chỉ.)

Bây giờ chúng ta tạo một *soft Root CA crypto token* và các khóa:

- Trong menu EJBCA, dưới chức năng CA, nhập vào *Crypto Tokens*.
- Nhập vào *Create new* và chỉ định các thông tin sau trên trang *New Crypto Token*:

The screenshot shows the 'Create New Crypto Token' form. The 'Name' field is set to 'MyFirstRootCACryptoToken'. The 'Type' dropdown is set to 'SOFT'. The 'Authentication Code' and 'Repeat Authentication Code' fields both contain '*****'. The 'Save' button is at the bottom.

© 2002–2021 EJBCA® is a registered trademark

Hình 20 - Tạo Crpyto Token cho chứng chỉ Root CA

- Nhập Save để tạo crypto token Root CA.
- Tiếp theo, tạo ba cặp khóa:

Crypto Token : MyFirstRootCACryptoToken

[Back to Crypto Token overview](#) [Switch to edit mode](#)

ID	622495822				
Name	MyFirstRootCACryptoToken				
Type	SoftCryptoToken				
Used	<input type="checkbox"/>				
Active	<input checked="" type="checkbox"/>				
Auto-activation	<input type="checkbox"/>				
Use explicit ECC parameters (ICAO CSCA and DS certificates)	<input type="checkbox"/>				
Allow export of private keys	<input type="checkbox"/>				

Alias	Key Algorithm	Key Specification	SubjectKeyID	Action
myFirstRootCaEncryptKey0001	RSA	4096	218d44b7c93917712e2b880df5b2ff583ee2622b	Test Remove Download Public Key
myFirstRootCaSignKey0001	RSA	4096	755f2076108e12eae0f0d5da4c7afb9f62426b9d	Test Remove Download Public Key
testKey	RSA	4096	1881805784fceae1b8cbda6a2cd5a2fa30fdcd31	Test Remove Download Public Key

[Remove selected](#)

signKey [RSA 4096](#) [Generate new key pair](#)

Hình 21 - Sinh 3 khóa-khóa ký, khóa mã hóa, khóa test

Bây giờ bạn đã tạo xong Root CA crypto token và các khóa, và có thể tiếp tục tạo CA đầu tiên của mình.

Manage Crypto Tokens

Name	Type	Library	Reference Type	Reference	Active	Auto-activation	Used	Actions	
ManagementCA	Soft				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Yes	Reactivate	Delete
MyFirstRootCACryptoToken	Soft				<input checked="" type="checkbox"/>		No	Deactivate	Delete

[Create new...](#)

© 2002–2024. EJBCA® is a registered trademark.

Hình 22 - Tạo thành công Crypto token

(c) Tạo root CA

Để tạo Root CA đầu tiên, làm theo các bước sau:

- Nhập vào *Certification Authorities* dưới chức năng CA.
- Trong trường *Add CA*, nhập tên *MyFirstRootCA* và nhấp *Create*.

Manage Certification Authorities

List of Certification Authorities

ManagementCA (Active)

[Edit CA](#) [Delete CA](#) [Import CA keystore...](#) [Import CA certificate...](#)

[Create Authenticated Certificate Signing Request](#)

Add CA

[Create...](#) [Rename selected](#)

Hình 23 - Tạo Root CA tên là MyFirstRootCA

- Trong trang *Create CA*, chọn *Root CA crypto token* là

MyFirstRootCACryptoToken (được tạo trong bước 2 - Tạo crypto token) trong danh sách *Crypto Token*. Các khóa *certSignKey* và *keyEncryptKey* sẽ tự động được chọn với các khóa bạn đã tạo, đối với **defaultKey**, chọn khóa

myFirstRootCaEncryptKey0001.

CA Name : MyFirstRootCA

Back to Certificate Authorities

CA Type X.509 CA CVC CA

Crypto Token [Select](#)

Signing Algorithm [Select](#)

Alternative Signing Algorithm [Select](#)

defaultKey [Select](#)

certSignKey [Select](#)

alternativeCertSignKey [Select](#)

crlSignKey Use same as Certificate Signing Key (certSignKey).

keyEncryptKey - Default key [Select](#)

Note: Only RSA or ECDH compatible ECC key algorithms (P-224, P-256, P-384, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571) may be used. Also, key must be of the same curve as the signing key.

testKey [Select](#)

Key sequence format [Select](#)

Key sequence

Hình 24 - Cài đặt khóa vừa tạo với CA

- Chỉ định các thông tin sau:

CA Certificate Data	
Subject DN	CN=MyFirstRootCA,O=PTIT,C=VN
Signed By	Self Signed
Certificate Profile	MyRootCAProfile
Validity ("y *mo *d *h *m *s) or end date of the certificate	30y
Subject Alternative Name	
Certificate Policy OID	

ISO 8601 date: 2025-03-18T00:00:00+00:00, y=30 years, mo=0 days
Leave policy OID blank to use default certificate profile values

Hình 25 - Cài đặt các thông tin của CA và thời gian hiệu lực của CA

- **CRL Expire Period:** Cập nhật thời gian sống của CRL thành 3 tháng bằng cách nhập 3 mo. CRL là danh sách các chứng chỉ đã bị thu hồi trước khi hết hạn.

CRL Specific Data	
Microsoft CA Compatibility Mode	<input type="checkbox"/> Use Warning! Microsoft CA Compatibility Mode is irreversible.
Authority Key ID	<input checked="" type="checkbox"/> Use <input type="checkbox"/> Critical
CRL Number	<input checked="" type="checkbox"/> Use <input type="checkbox"/> Critical
Issuing Distribution Point on CRLs	<input type="checkbox"/> Use <input type="checkbox"/> Critical
CA issuer URI	
Keep expired certificates on CRL	<input type="checkbox"/> Use
Use CRL partitions	<input type="checkbox"/> Use
CRL Expire Period ("y *mo *d *h *m")	3mo
CRL Issue Interval ("y *mo *d *h *m")	0m
CRL Overlap Time ("y *mo *d *h *m")	0m
Delta CRL Period ("y *mo *d *h *m")	0m
Generate CRL Upon Revocation	<input type="checkbox"/> Use A fresh CRL or delta CRL is generated after revocation or reactivation of a certificate.
Allow changing revocation reason	<input type="checkbox"/> Use
Allow invalidity date	<input type="checkbox"/> Use
Publishers	

Hình 26 - Cấu hình CRL- Danh sách thu hồi chứng chỉ

- Nhập Create để tạo Root CA.

Crypto token **MyFirstRootCACryptoToken** vừa tạo sẽ được hiển thị trong danh sách các CA.

3.2.2.2 Tạo một hệ thống phân cấp PKI trong EJBCA

Việc tạo một hệ thống phân cấp CA đa cấp được khuyến khích. Với cấu hình này, một Root CA (mỏ neo tin cậy) chỉ cấp chứng chỉ cho các CA phụ, trao quyền cho chúng cấp chứng chỉ mới. Các CA phụ có thể cấp chứng chỉ trực tiếp cho các thực thể cuối (end entities) hoặc tạo thêm các cấp CA phụ khác. Lợi ích chính của phương pháp này là Root CA có thể

được kiểm soát chặt chẽ và có thời gian hiệu lực dài, trong khi các CA phụ có thể có thời gian hiệu lực ngắn hơn và có thể bị thu hồi nếu cần. Điều này giúp giảm thiểu sự cần thiết phải cập nhật và thay thế chứng chỉ Root CA tin cậy.

Các CA phụ khác nhau cũng có thể được quản lý bởi các nhóm khác nhau, ngay cả khi chúng được lưu trữ trong cùng một phiên bản EJBCA. Ví dụ, bạn có thể có một CA cho các thực thể nội bộ và một CA cho các thực thể bên ngoài, các CA khác nhau cho các vùng khác nhau, hoặc các CA cho các mục đích khác nhau, và nhiều trường hợp khác.

(a) *Tạo hồ sơ chứng chỉ*

Tạo hồ sơ chứng chỉ Root CA:

- Trong EJBCA, dưới chức năng CA, nhấp vào *Certificate Profiles*.
- Trang *Quản lý Hồ sơ Chứng chỉ* sẽ hiển thị danh sách các hồ sơ mặc định và hồ sơ *Root CA Profile* đã tạo trong hướng dẫn trước.
- Nhấp vào *Clone* bên cạnh mẫu *MyRootCAProfile* để sử dụng mẫu này làm cơ sở cho việc tạo hồ sơ Root CA mới.

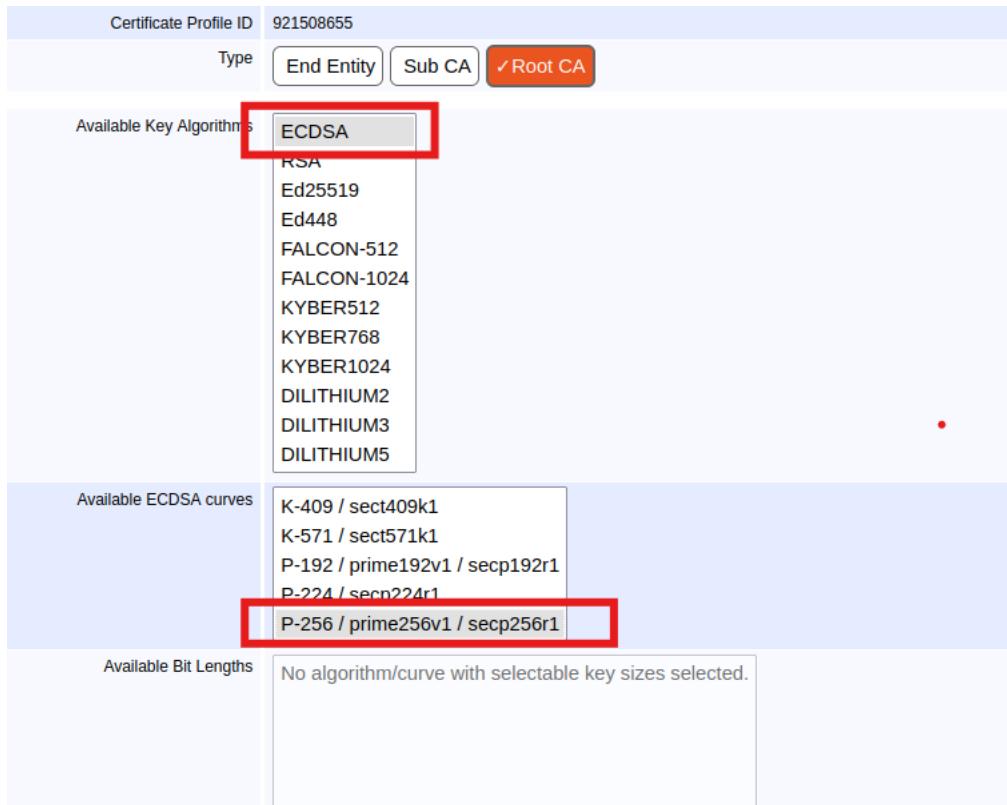
Manage Certificate Profiles

List of Certificate Profiles

Name	Actions					
ENDUSER	View	Edit	Delete	Rename	Clone	Export
OCSPSIGNER	View	Edit	Delete	Rename	Clone	Export
ROOTCA	View	Edit	Delete	Rename	Clone	Export
SERVER	View	Edit	Delete	Rename	Clone	Export
SUBCA	View	Edit	Delete	Rename	Clone	Export
MyRootCAProfile	View	Edit	Delete	Rename	Clone	Export
	Add					

Hình 27 - *Clone phần tạo chứng chỉ Root CA*

- Đặt tên hồ sơ chứng chỉ mới là **MyPKIRootCAProfile**, và nhấp vào **Create from template**.
- Để chỉnh sửa các giá trị của hồ sơ sao cho phù hợp với nhu cầu, tìm hồ sơ *MyPKIRootCAProfile* mới tạo trong danh sách và nhấp vào *Edit*.
- Trong trang chỉnh sửa, cập nhật các thông số sau để sử dụng các khóa elliptic curve thay vì khóa RSA:



Hình 28 - Cài đặt thuật toán ECDSA với đường cong P-256

- Nhập vào *Save* để lưu hồ sơ chứng chỉ Root CA.
- Hồ sơ MyPKIRootCAProfile mới tạo sẽ hiển thị trong danh sách hồ sơ chứng chỉ.
- Tạo hồ sơ chứng chỉ Sub CA**
- Trong trang *Quản lý Hồ sơ Chứng chỉ* của EJBCA, nhập vào *Clone* bên cạnh mẫu *SUBCA* để tạo hồ sơ mới từ mẫu đó.

Manage Certificate Profiles

List of Certificate Profiles

Name	Actions						
ENDUSER	View	Edit	Delete	Rename	Clone	Export	
OCSPSIGNER	View	Edit	Delete	Rename	Clone	Export	
ROOTCA	View	Edit	Delete	Rename	Clone	Export	
SERVER	View	Edit	Delete	Rename	Clone	Export	
SUBCA	View	Edit	Delete	Rename	Clone	Export	
MyPKIRootCAProfile	View	Edit	Delete	Rename	Clone	Export	
MyRootCAProfile	View	Edit	Delete	Rename	Clone	Export	
			Add				

Import/Export

Import Profiles from Zip file:

[Browse...](#) No file selected.

[Import Profiles](#) [Export Profiles](#)

Hình 29 - Clone từ template SUBCA làm Sub CA

- Đặt tên hồ sơ chứng chỉ mới là *MyPKISubCAProfile*, và nhập vào *Create from template*.

Manage Certificate Profiles

List of Certificate Profiles

Name	Actions					
ENDUSER	View	Edit	Delete	Rename	Clone	Export
OCSPSIGNER	View	Edit	Delete	Rename	Clone	Export
ROOTCA	View	Edit	Delete	Rename	Clone	Export
SERVER	View	Edit	Delete	Rename	Clone	Export
SUBCA	View	Edit	Delete	Rename	Clone	Export
MyPKIRootCAProfile	View	Edit	Delete	Rename	Clone	Export
MyPKISubCAProfile	View	Edit	Delete	Rename	Clone	Export
MyRootCAProfile	View	Edit	Delete	Rename	Clone	Export
	Add					

Import/Export

Import Profiles from Zip file:

[Browse...](#) No file selected.

[Import Profiles](#) [Export Profiles](#)

Hình 30 - Khởi tạo thành công Sub CA

- Để chỉnh sửa các giá trị của hồ sơ sao cho phù hợp với nhu cầu, tìm hồ sơ *MyPKISubCAProfile* mới tạo trong danh sách và nhấp vào *Edit*.
- Trong trang chỉnh sửa, cập nhật các thông số sau để sử dụng các khóa elliptic curve thay vì khóa RSA:

Certificate Profile: MyPKISubCAProfile

Back to Certificate Profiles

Certificate Profile ID: 1282227607

Type: End Entity Sub CA Root CA

Available Key Algorithms: ECDSA RSA Ed25519 Ed448 FALCON-512 FALCON-1024 KYBER512 KYBER768 KYBER1024 DILITHIUM2 DILITHIUM3 DILITHIUM5

Available ECDSA curves: K-409 / sect409k1 K-571 / sect571k1 P-192 / prime192v1 / secp192r1 P-224 / secp224r1 P-256 / prime256v1 / secp256r1

Available Bit Lengths: No algorithm/curve with selectable key sizes selected.

Signature Algorithm: Inherit from issuing CA

Hình 31 - Cài đặt cấu hình khóa cho chứng chỉ Sub CA

Validity or end date of the certificate	<input type="text" value="15y"/>
ISO 8601 date: [yyyy-MM-dd HH:mm:ssZZ]: '2025-03-18 01:39:14+00:00' (*y *mo *d *h *m *s) - y=365 days, mo=30 days	

Hình 32 - Thời hạn của chứng chỉ Sub CA

- Dưới phần X.509v3 extensions, cập nhật các thông số sau:
 - Chọn Path Length Constraint và đặt giá trị là 0 để đảm bảo Sub CA này không thể cấp thêm các CA phụ dưới nó và chỉ được phép cấp chứng chỉ cho các thực thể cuối.

X.509v3 extensions	
Basic Constraints	<input checked="" type="checkbox"/> Use... <input checked="" type="checkbox"/> Critical
Path Length Constraint	<input checked="" type="checkbox"/> Add... Value <input type="text" value="0"/>
Authority Key ID	<input checked="" type="checkbox"/> Use
Subject Key ID	<input checked="" type="checkbox"/> Use <input type="checkbox"/> Truncated KeyID (method 2 in F)

Hình 33 - Cấu hình đảm bảo Sub CA không thể cấp thêm các CA phụ dưới nó

- Dưới phần X.509v3 extensions - Validation data, cập nhật các thông số sau:
 - Kích hoạt CRL Distribution Points (Điểm phân phối CRL).
 - Kích hoạt Use CA defined CRL Distribution Point để cho phép thiết lập điểm phân phối CRL trong Root CA.
 - Kích hoạt Authority Information Access.
 - Kích hoạt Use CA defined OCSP locator: để kiểm tra trạng thái chứng chỉ trực tuyến, thay vì phải tải về toàn bộ CRL.
 - Kích hoạt Use CA defined CA issuer: Cấu hình một URI (địa chỉ) dẫn đến thông tin về CA phát hành chứng chỉ, cho phép xác minh nguồn gốc của chứng chỉ.

X.509v3 extensions		Validation data
CRL Distribution Points	<input checked="" type="checkbox"/> Use... <input type="checkbox"/> Critical	
Use CA defined CRL Distribution Point	<input checked="" type="checkbox"/> Use...	
CRL Distribution Point URI	<input type="text" value="http://localhost:80/ejbca/publicweb/webdist/certdist?cmd=crl&iss"/>	
CRL Issuer	<input type="text" value="CN=TestCA,O=AnaTom,C=SE"/>	
Freshest CRL (a.k.a. Delta CRL DP)	<input type="checkbox"/> Use...	
Authority Information Access	<input checked="" type="checkbox"/> Use...	
Use CA defined OCSP locator	<input checked="" type="checkbox"/> Use...	
OCSP Service Locator URI		
Use CA defined CA issuer	<input checked="" type="checkbox"/> Use...	
CA issuer URI		
Private Key Usage Period	<input type="checkbox"/> Start offset... <input type="text" value="(*y *mo *d *h *m *s)"/>	(*y *mo *d *h *m *s)
	<input type="checkbox"/> Period length... <input type="text" value="(*y *mo *d *h *m *s)"/>	(*y *mo *d *h *m *s)

Hình 34 - Cấu hình điểm phân phối CRL trong chứng chỉ Sub CA

- Dưới phần *Other Data*, bỏ chọn *LDAP DN order*.
- Nhập vào *Save* để lưu hồ sơ chứng chỉ Sub CA.

Hồ sơ *MyPKISubCAProfile* mới tạo sẽ hiển thị trong danh sách hồ sơ chứng chỉ.

(b) Tạo crypto tokens

Tạo crypto token Root CA

- Nhập vào *Create new* và chỉ định các thông tin sau trong trang *New Crypto Token*:
 - Name:** Đặt tên cho crypto token Root CA là *MyPKIRootCACryptoToken*.
 - Authentication Code:** Nhập mật khẩu sẽ được sử dụng để kích hoạt crypto token nếu container bị khởi động lại. Ghi nhớ mật khẩu này.

New Crypto Token

Name	MyPKIRootCACryptoToken
Type	SOFT
Auto-activation	<input type="checkbox"/> Use
Use explicit ECC parameters (ICAO CSCA and DS certificates)	<input type="checkbox"/> Use
Allow export of private keys	<input type="checkbox"/> Allow
Authentication Code	*****
Repeat Authentication Code	*****
<input type="button" value="Save"/>	

Hình 35 - Tạo Crypto Token cho RootCA

- Tiếp theo, tạo ba cặp khóa CA:

Crypto Token : MyPKIRootCACryptoToken

Back to Crypto Token overview	Switch to edit mode																				
ID	-988169087																				
Name	MyPKIRootCACryptoToken																				
Type	SoftCryptoToken																				
Used	<input type="checkbox"/>																				
Active	<input checked="" type="checkbox"/>																				
Auto-activation	<input type="checkbox"/>																				
Use explicit ECC parameters (ICAO CSCA and DS certificates)	<input type="checkbox"/>																				
Allow export of private keys	<input type="checkbox"/>																				
<table border="1"> <thead> <tr> <th>Alias</th> <th>Key Algorithm</th> <th>Key Specification</th> <th>SubjectKeyID</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>myPKIRootCaEncryptKey0001</td> <td>RSA</td> <td>4096</td> <td>b6d6b7f4382e2b74f21973eb1eea7eea80658822</td> <td><input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/></td> </tr> <tr> <td>myPKIRootCaSignKey0001</td> <td>ECDSA</td> <td>prime256v1 / secp256r1 / P-256</td> <td>c08dbac8e037dd5c133414865f9a8a928607e303</td> <td><input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/></td> </tr> <tr> <td>testKey</td> <td>ECDSA</td> <td>prime256v1 / secp256r1 / P-256</td> <td>f2b9023ff4b98440d99ae3a49b738ef2d86d75be</td> <td><input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/></td> </tr> </tbody> </table>		Alias	Key Algorithm	Key Specification	SubjectKeyID	Action	myPKIRootCaEncryptKey0001	RSA	4096	b6d6b7f4382e2b74f21973eb1eea7eea80658822	<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>	myPKIRootCaSignKey0001	ECDSA	prime256v1 / secp256r1 / P-256	c08dbac8e037dd5c133414865f9a8a928607e303	<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>	testKey	ECDSA	prime256v1 / secp256r1 / P-256	f2b9023ff4b98440d99ae3a49b738ef2d86d75be	<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>
Alias	Key Algorithm	Key Specification	SubjectKeyID	Action																	
myPKIRootCaEncryptKey0001	RSA	4096	b6d6b7f4382e2b74f21973eb1eea7eea80658822	<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>																	
myPKIRootCaSignKey0001	ECDSA	prime256v1 / secp256r1 / P-256	c08dbac8e037dd5c133414865f9a8a928607e303	<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>																	
testKey	ECDSA	prime256v1 / secp256r1 / P-256	f2b9023ff4b98440d99ae3a49b738ef2d86d75be	<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>																	
<input type="button" value="Remove selected"/>																					
<input type="text" value="testKey"/> <input type="button" value="ECDSA P-256 / prime256v1 / secp256r1"/> <input type="button" value="Generate new key pair"/>																					
<small>© 2002–2024. EJBCA® is a registered trademark.</small>																					

Hình 36 - Tạo ba cặp khóa cài ở bước một

Bây giờ bạn đã tạo xong **Root CA crypto token** và các khóa.

Tạo crypto token Sub CA

- Nhập vào *Create new* và chỉ định các thông tin sau trong trang *New Crypto Token*:
 - Authentication Code:** Nhập mật khẩu cho tự động kích hoạt.

New Crypto Token

Name	<input type="text" value="MyPKISubCACryptoToken"/>
Type	<input type="button" value="SOFT"/>
Auto-activation	<input checked="" type="checkbox"/> Use
Use explicit ECC parameters (ICAO CSCA and DS certificates)	<input type="checkbox"/> Use
Allow export of private keys	<input type="checkbox"/> Allow
Authentication Code	<input type="text" value="....."/>
Repeat Authentication Code	<input type="text" value="....."/>
<input type="button" value="Save"/>	

Hình 37 - Tạo Crypto token cho Sub CA

- Tiếp theo, tạo ba cặp khóa cho Sub CA:

Crypto Token : MyPKISubCACryptoToken

Back to Crypto Token overview		Switch to edit mode	
ID	-1318738124		
Name	MyPKISubCACryptoToken		
Type	SoftCryptoToken		
Used	<input type="checkbox"/>		
Active	<input checked="" type="checkbox"/>		
Auto-activation	<input checked="" type="checkbox"/>		
Use explicit ECC parameters (ICAO CSCA and DS certificates)	<input type="checkbox"/>		
Allow export of private keys	<input type="checkbox"/>		
Alias	Key Algorithm	Key Specification	SubjectKeyID
<input type="checkbox"/> myPkiRSubCaEncryptKey0001	RSA	4096	9a17c436c1e3afbd11ca3b0d0d006db99a628d4e
<input type="checkbox"/> myPkiRSubCaSignKey0001	ECDSA	prime256v1 / secp256r1 / P-256	0d4790dd6e501514dd9f89e30ae12d9d8143b09ff
<input type="checkbox"/> testKey	ECDSA	prime256v1 / secp256r1 / P-256	da39d520fe47de81869d4d75f597e47f4121866e
		Action	
		<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>	
		<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>	
		<input type="button" value="Test"/> <input type="button" value="Remove"/> <input type="button" value="Download Public Key"/>	
		<input type="button" value="Remove selected"/>	
<input type="text" value="testKey"/>		<input type="button" value="ECDSA P-256 / prime256v1 / secp256r1"/> <input type="button" value="Generate new key pair"/>	

Hình 38 - Tạo ba cặp khóa tương ứng với thuật toán cho Sub CA

Bây giờ bạn đã tạo xong **Sub CA crypto token** và các khóa.

(c) Tạo CA

Sau khi đã tạo hồ sơ chứng chỉ và các khóa CA, bạn có thể kết hợp chúng để tạo ra hệ thống CA phân cấp.

Khi bạn đã quyết định hệ thống phân cấp CA của mình, bạn cần quyết định mỗi chứng chỉ sẽ có hiệu lực trong bao lâu. Một nguyên tắc chung là CA phụ phải có một nửa thời hạn hiệu lực của CA phát hành. Một thiết lập điển hình, được sử dụng trong hệ thống phân cấp hai tầng ví dụ này, sẽ là đặt CA gốc thành 30 năm, CA cấp hai thành 15 năm. Điều này có nghĩa là một lần gia hạn CA phụ theo kế hoạch sẽ cần thiết trong suốt thời gian tồn tại của CA gốc.

Tạo Root CA

- Nhập vào *Certification Authorities* dưới chức năng CA.
- Trong trường *Add CA*, nhập tên *MyPKIRootCA-G1* và nhập *Create*.

List of Certification Authorities

ManagementCA (Active)
MyFirstRootCA (Active)

[Edit CA](#) [Delete CA](#) [Import CA keystore...](#) [Import CA certificate...](#)

[Create Authenticated Certificate Signing Request](#)

Add CA

MyPKIRootCA-G1

[Create...](#)

[Rename selected](#)

Hình 39 - Tạo Root CA -G1

- Trong trang *Create CA*, cập nhật các thông tin sau:
 - Chọn crypto token Root CA *MyPKIRootCACryptoToken*.
 - Chọn *SHA512withECDSA* làm thuật toán ký.
 - Các khóa *certSignKey* và *keyEncryptKey* sẽ tự động được chọn với các khóa bạn đã tạo, chọn *myPkiRootCaEncryptKey0001* cho *defaultKey*.

Back to Certificate Authorities

CA Type X.509 CA CVC CA

Crypto Token

Signing Algorithm

Alternative Signing Algorithm

defaultKey

certSignKey

alternativeCertSignKey

crlSignKey Use same as Certificate Signing Key (certSignKey).

keyEncryptKey

Note: Only RSA or ECDH compatible ECC key algorithms (P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571) may be used. Also, the encryption key must be of the same curve as the signing key.

testKey

Key sequence format

Hình 40 Cấu hình Root CA -G1 với Crypto token

- Cập nhật thông tin CA và các điểm phân phối CRL như sau:
 - Subject DN: "CN = My PKI Root CA - G1, O = PTIT, C = VN".
 - Signed By: Self Signed.
 - Certificate Profile: Chọn MyPKIRootCAProfile.
 - Validity: 30 năm.
 - CRL Expire Period: 3 tháng.

CA Certificate Data	
Subject DN	CN=MyPKIRootCA-G1,O=PTIT,C=VN
Signed By	Self Signed
Certificate Profile	MyPKIRootCAProfile
Validity (*y *mo *d *h *m *s) or end date of the certificate	30y 25-03-18 02:35:34+00:00 y=365 days, mo=30 days
Subject Alternative Name	
Certificate Policy OID	
Use UTF-8 in policy notice text	<input checked="" type="checkbox"/> Use
PrintableString encoding in DN	<input type="checkbox"/> Use
LDAP DN order	<input type="checkbox"/> Use
Serial Number Octet Size	20
CRL Expire Period (*y *mo *d *h *m)	3mo y=365 days , mo=30 days
CRL Issue Interval (*y *mo *d *h *m)	0m y=365 days , mo=30 days
CRL Overlap Time (*y *mo *d *h *m)	0m y=365 days , mo=30 days
Delta CRL Period (*y *mo *d *h *m)	0m y=365 days , mo=30 days (0m, if no delta CRLs are issued)

Hình 41 Cấu hình thông tin CA và CRL

- Trong *Default CA defined validation data*, xác định các giá trị mặc định sẽ được sử dụng trong hồ sơ chứng chỉ do CA cấp:
 - *Default CRL Distribution Point*: <http://my.pki/crls/MyPKIRootCA-G1.crl>
 - *OCSP service Default URI*: <http://my.pki/ocsp>
 - *CA issuer Default URI*: <http://my.pki/certs/MyPKIRootCA-G1.crt>

Default CA defined validation data		Used as default values in certificate profiles using this CA
Default CRL Distribution Point	<input type="text" value="http://my.pki/crls/MyPKIRootCA-G1.crl"/> <small>(used in CRL, and as default value)</small>	<input type="button" value="Generate"/>
Default CRL Issuer	<input type="text"/>	<input type="button" value="Generate"/>
Default Freshest CRL Distribution Point	<input type="text"/> <small>(used in CRL, and as default value)</small>	<input type="button" value="Generate"/>
OCSP service Default URI	<input type="text" value="http://my.pki/ocsp"/>	<input type="button" value="Generate"/>
CA issuer Default URI	<input type="text" value="http://my.pki/certs/MyPKIRootCA-G1.crt"/>	

Hình 42 Cấu hình địa chỉ dẫn CRL và OCSP

- Nhập *Create* để tạo Root CA.

MyPKIRootCA-G1 đã tạo được hiển thị trong danh sách các CA.

Tạo Sub CA

- Trong trường *Add CA*, nhập tên *MyPKISubCA-G1* và nhấp *Create*.

Manage Certification Authorities

List of Certification Authorities

ManagementCA (Active)
MyFirstRootCA (Active)
MyPKIRootCA-G1 (Active)

Add CA

Hình 43 Tao Sub CA-G1

- Trong trang *Create CA*, cập nhật các thông tin sau:
 - Chọn crypto token Sub CA *MyPKISubCACryptoToken*.
 - Chọn *SHA256withECDSA* làm thuật toán ký.
 - Chọn *myPkiSubCaEncryptKey0001* cho *defaultKey*.

Back to Certificate Authorities

CA Type	<input checked="" type="checkbox"/> X.509 CA <input type="checkbox"/> CVC CA
Crypto Token	MyPKISubCACryptoToken <input style="width: 20px; height: 20px;" type="button" value="..."/>
	myPkiRSubCaEncryptKey0001 - RSA 4096 myPkiRSubCaSignKey0001 - ECDSA prime256v1 testKey - ECDSA prime256v1
Signing Algorithm	SHA256withECDSA <input style="width: 20px; height: 20px;" type="button" value="..."/>
Alternative Signing Algorithm	Select an algorithm to activate hybrid certificates. <input style="width: 20px; height: 20px;" type="button" value="..."/>
defaultKey	myPkiRSubCaEncryptKey0001 <input style="width: 20px; height: 20px;" type="button" value="..."/>
certSignKey	myPkiRSubCaSignKey0001 <input style="width: 20px; height: 20px;" type="button" value="..."/>
alternativeCertSignKey	<input style="width: 20px; height: 20px;" type="button" value="..."/>
crlSignKey	Use same as Certificate Signing Key (certSignKey).
keyEncryptKey	- Default key <input style="width: 20px; height: 20px;" type="button" value="..."/>
Note: Only RSA or ECDH compatible ECC key algorithms (P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571) may be used. Also, the encryption key must be of the same curve as the signing key.	
testKey	testKey <input style="width: 20px; height: 20px;" type="button" value="..."/>
Key sequence format	numeric [0-9] <input style="width: 20px; height: 20px;" type="button" value="..."/>
Key sequence	00000 <input style="width: 20px; height: 20px;" type="button" value="..."/>
Description	<input style="height: 40px; width: 100%;" type="text"/>

Hình 44 Cấu hình Sub Ca -G1 phù hợp với Crypto token

- Cập nhật thông tin CA và các điểm phân phối CRL như sau:

CA Certificate Data

Subject DN	CN=MyPKISubCA-G1,O=PTIT,C=VN <input style="width: 20px; height: 20px;" type="button" value="..."/>
Signed By	MyPKIRootCA-G1 <input style="width: 20px; height: 20px;" type="button" value="..."/>
Certificate Profile	MyPKISubCAProfile <input style="width: 20px; height: 20px;" type="button" value="..."/>
Validity (*y *mo *d *h *m *s) or end date of the certificate	15y <input style="width: 20px; height: 20px;" type="button" value="..."/>
Subject Alternative Name	<input type="text"/>
Certificate Policy OID	<input type="text"/>
Use UTF-8 in policy notice text	<input checked="" type="checkbox"/> Use <input style="width: 20px; height: 20px;" type="button" value="..."/>
PrintableString encoding in DN	<input type="checkbox"/> Use <input style="width: 20px; height: 20px;" type="button" value="..."/>
LDAP DN order	<input type="checkbox"/> Use <input style="width: 20px; height: 20px;" type="button" value="..."/>
Serial Number Octet Size	20 <input style="width: 20px; height: 20px;" type="button" value="..."/>
Name Constraints, Permitted	<input type="text"/>

Hình 45 Cấu hình CA và điểm phân phối CRL

- Trong Dữ liệu xác thực được xác định của CA mặc định, xác định các giá trị mặc định sẽ được sử dụng trong hồ sơ chứng chỉ do CA cấp:
 - Default CRL Distribution Point: <http://my.pki/crls/MyPKISubCA-G1.crl>

- *OCSP service Default URI*: http://my.pki/ocsp
- *CA issuer Default URI*: http://my.pki/certs/MyPKISubCA-G1.crt

Default CA defined validation data		Used as default values in certificate profiles using this CA
Default CRL Distribution Point	http://my.pki/crls/MyPKISubCA-G1.crl <small>(used in CRL, and as default value)</small>	Generate
Default CRL Issuer		Generate
Default Freshest CRL Distribution Point		Generate
OCSP service Default URI	http://my.pki/ocsp	Generate
CA issuer Default URI	http://my.pki/certs/MyPKISubCA-G1.crt	

Hình 46 Cấu hình đường dẫn CRL và OCSP

- Nhập **Create** để tạo Sub CA.
- MyPKISubCA-G1 đã tạo được hiển thị trong danh sách các CA.

Manage Certification Authorities

List of Certification Authorities
ManagementCA (Active)
MvFirstRootCA (Active)
MyPKIRootCA-G1 (Active)
MyPKISubCA-G1 (Active)

Hình 47 Kết quả khi tạo được Root CA-G1 và Sub CA-G1

Bây giờ bạn đã tạo một hệ thống phân cấp cơ sở hạ tầng khóa công khai (PKI) hai tầng với CA gốc và CA cấp dưới.

3.2.2.3 Cấp chứng chỉ máy chủ TLS với EJBCA

Trong hệ thống Public Key Infrastructure (PKI), việc cấp phát và quản lý chứng chỉ đóng vai trò quan trọng trong việc đảm bảo tính bảo mật và xác thực cho các kết nối mạng. Hướng dẫn này cung cấp quy trình chi tiết để tạo hồ sơ chứng chỉ và cấp chứng chỉ máy chủ TLS sử dụng EJBCA, một giải pháp quản lý chứng chỉ phổ biến trong môi trường PKI.

Chứng chỉ TLS (Transport Layer Security) là một thành phần quan trọng trong việc bảo vệ các kết nối giữa máy chủ và client, đặc biệt trong các giao thức như HTTPS. Việc cấu hình và quản lý chứng chỉ TLS chính xác không chỉ giúp đảm bảo sự bảo mật mà còn giúp duy trì độ tin cậy của hệ thống trong suốt quá trình hoạt động.

Thông qua các bước này, bạn sẽ nắm được cách cấu hình hệ thống EJBCA để cấp chứng chỉ TLS một cách hiệu quả, đồng thời bảo mật các kết nối máy chủ và đảm bảo tính toàn vẹn trong việc quản lý chứng chỉ.

(a) *Tạo hồ sơ chứng chỉ*

Bước đầu tiên là tạo một hồ sơ chứng chỉ cho các chứng chỉ TLS máy chủ. Hồ sơ chứng chỉ này xác định nội dung và các ràng buộc của chứng chỉ mới, ví dụ như các loại khóa được phép sử dụng và các phần mở rộng trong chứng chỉ.

Để tạo hồ sơ chứng chỉ cho TLS máy chủ, làm theo các bước sau:

- Trong EJBCA, trong *CA Functions*, bấm vào *Certificate Profiles*, trang *Quản lý hồ sơ chứng chỉ* hiển thị danh sách các cấu hình có sẵn.
- Nhấp vào *Clone* bên cạnh mẫu *SERVER* để sử dụng làm cơ sở tạo hồ sơ mới của bạn.
- Đặt tên cho hồ sơ chứng chỉ mới là *TLS Server Profile* và nhấp vào *Create from template*.

Manage Certificate Profiles

Clone

Template certificate profile SERVER
Name of new certificate profile

© 2002–2024. EJBCA® is a registered trademark.

Hình 48 Tạo hồ sơ chứng chỉ TLS Server

- Để chỉnh sửa giá trị hồ sơ sao cho phù hợp với nhu cầu của bạn, tìm hồ sơ *TLS Server Profile* vừa tạo trong danh sách và nhấp vào *Edit*.
- Trong trang chỉnh sửa, xác nhận loại là *End Entity* và cập nhật các thông số sau:

Available Key Algorithms	ECDSA RSA Ed25519 Ed448 FALCON-512 FALCON-1024 KYBER512 KYBER768 KYBER1024 DILITHIUM2 DILITHIUM3 DILITHIUM5
Available ECDSA curves	K-409 / sect409k1 K-571 / sect571k1 P-192 / prime192v1 / secp192r1 P-224 / secp224r1 P-256 / prime256v1 / secp256r1
Available Bit Lengths	No algorithm/curve with selectable key sizes selected.
Signature Algorithm	<input style="width: 100%;" type="button" value="Inherit from issuing CA"/>
Alternative Signature	<input type="checkbox"/> Use
Validity or end date of the certificate	<input type="text" value="1y"/> <small>ISO 8601 date: [yyyy-MM-dd HH:mm:ssZZ] '2025-03-18 11:57:45+00:00'</small>

Hình 49 Cấu hình thuật toán khóa và thời hạn chứng chỉ

- Phần *tiện ích mở rộng X.509v3* cho phép bạn xác định các tiện ích mở rộng được thêm vào chứng chỉ:
 - Xóa *Basic Constraints* vì phần này xác định đây là chứng chỉ của thực thể cuối chứ không phải chứng chỉ CA.
 - *Key Usage*: Đảm bảo rằng *Digital Signature* và *Key encipherment* được chọn.
 - *Extended Key Usage*: Đảm bảo rằng *Server Authentication* được chọn.
 - Xóa *Issuer Alternative Name* vì CA không có tên thay thế.

X.509v3 extensions		
Basic Constraints	<input type="checkbox"/> Use...	<input checked="" type="checkbox"/> Critical
Authority Key ID	<input checked="" type="checkbox"/> Use	
Subject Key ID	<input checked="" type="checkbox"/> Use	<input type="checkbox"/> Truncated KeyID (method 2 in RFC5280 which is uncommon, keep unchecked for most use cases)
X.509v3 extensions		
Key Usage	<input checked="" type="checkbox"/> Use... <input checked="" type="checkbox"/> Critical	<input type="checkbox"/> Forbid encryption usage for ECC keys
Key Usage:	<input checked="" type="checkbox"/> Digital Signature <input type="checkbox"/> Data encipherment <input type="checkbox"/> CRL sign <input type="checkbox"/> Non-repudiation <input type="checkbox"/> Key agreement <input type="checkbox"/> Encipher only <input checked="" type="checkbox"/> Key encipherment <input type="checkbox"/> Key certificate sign <input type="checkbox"/> Decipher only	
Extended Key Usage	<input checked="" type="checkbox"/> Use... <input type="checkbox"/> Critical	
	MS EFS Recovery MS Encrypted File System (EFS) MS Individual Code Signing MS Smart Card Logon OCSP Signer PDF Signing PIV Card Authentication RFC9336 Document Signing SCVP Client SCVP Server	
Certificate Policies	<input type="checkbox"/> Use... <input type="checkbox"/> Critical	
X.509v3 extensions		
Names		
Subject Alternative Name	<input checked="" type="checkbox"/> Use... <input type="checkbox"/> Critical	<input checked="" type="checkbox"/> Search enabled (search enabled SAN use more storage)
Issuer Alternative Name	<input type="checkbox"/> Use... <input type="checkbox"/> Critical	
Subject Directory Attributes	<input type="checkbox"/> Use	

Hình 50 Cấu hình X509v3 extenstions

- Dưới X.509v3 extensions - Validation data, cập nhật các thông số sau:
 - o Kích hoạt *CRL Distribution Points* để cho phép xác minh sau này.
 - o Kích hoạt *Use CA defined CRL Distribution Point* để sử dụng giá trị đã được định cấu hình sẵn trong CA.
 - o Kích hoạt *Authority Information Access* và các URI được định nghĩa trong cài đặt CA của bạn:
 - Kích hoạt *Use CA defined OCSP locator* cho nơi dịch vụ OCSP có sẵn.
 - Kích hoạt *Use CA defined CA issuer* cho nơi có thể truy xuất chứng chỉ CA phát hành.

CRL Distribution Points	<input checked="" type="checkbox"/> Use... <input type="checkbox"/> Critical
Use CA defined CRL Distribution Point	<input checked="" type="checkbox"/> Use...
CRL Distribution Point URI	<input type="text" value="http://localhost:80/ejbca/publicweb/webdist/certdist?cmd=crl&iss"/>
CRL Issuer	<input type="text" value="CN=TestCA,O=AnaTom,C=SE"/>
Freshest CRL (a.k.a. Delta CRL DP)	<input type="checkbox"/> Use...
Authority Information Access	<input checked="" type="checkbox"/> Use...
Use CA defined OCSP locator	<input checked="" type="checkbox"/> Use...
OCSP Service Locator URI	<input type="text"/>
Use CA defined CA issuer	<input checked="" type="checkbox"/> Use...
CA issuer URI	<input type="text"/> <input type="button" value="Add"/>
Private Key Usage Period	<input type="checkbox"/> Start offset... <input type="text"/> (*y *mo *d *h *m *s) <input type="checkbox"/> Period length... <input type="text"/> (*y *mo *d *h *m *s)

Hình 51 Cấu hình điểm CRL và OCSP với TLS Server

- Dưới *Other Data*, cập nhật các thông số sau:
 - Xóa *LDAP DN order* để tắt việc sắp xếp DN theo LDAP và sử dụng sắp xếp chuẩn X.509 thay vào đó.
 - Đối với *Available CAs*, chọn *MyPKISubCA-G1*.

Other Data	
LDAP DN order	<input type="checkbox"/> Use
Custom Subject DN Order	<input type="checkbox"/> Use... <input type="checkbox"/> Apply LDAP DN order settingValue <input type="text"/> (comma separated)
CN postfix	<input type="checkbox"/> Add... Value <input type="text"/> (text appended after first CN field)
Subset of Subject DN	<input type="checkbox"/> Restrict...
Subset of Subject Alt. Name	<input type="checkbox"/> Restrict...
Available CAs	<input type="checkbox"/> Any CA <input type="checkbox"/> ManagementCA <input type="checkbox"/> MyPKIRootCA-G1 <input checked="" type="checkbox"/> MyPKISubCA-G1
Publishers	<input type="text"/>
Single Active Certificate Constraint	<input type="checkbox"/> Use
Account Binding Namespace	<input type="text"/>
<input type="button" value="Save"/> <input type="button" value="Cancel"/>	

Hình 52 Cấu hình người cấp chứng chỉ

- Nhập Save để lưu hồ sơ chứng chỉ.

Hồ sơ **TLS Server Profile** mới tạo sẽ hiển thị trong danh sách hồ sơ chứng chỉ.

(b) *Tạo hồ sơ thực thể cuối (End Entity Profile)*

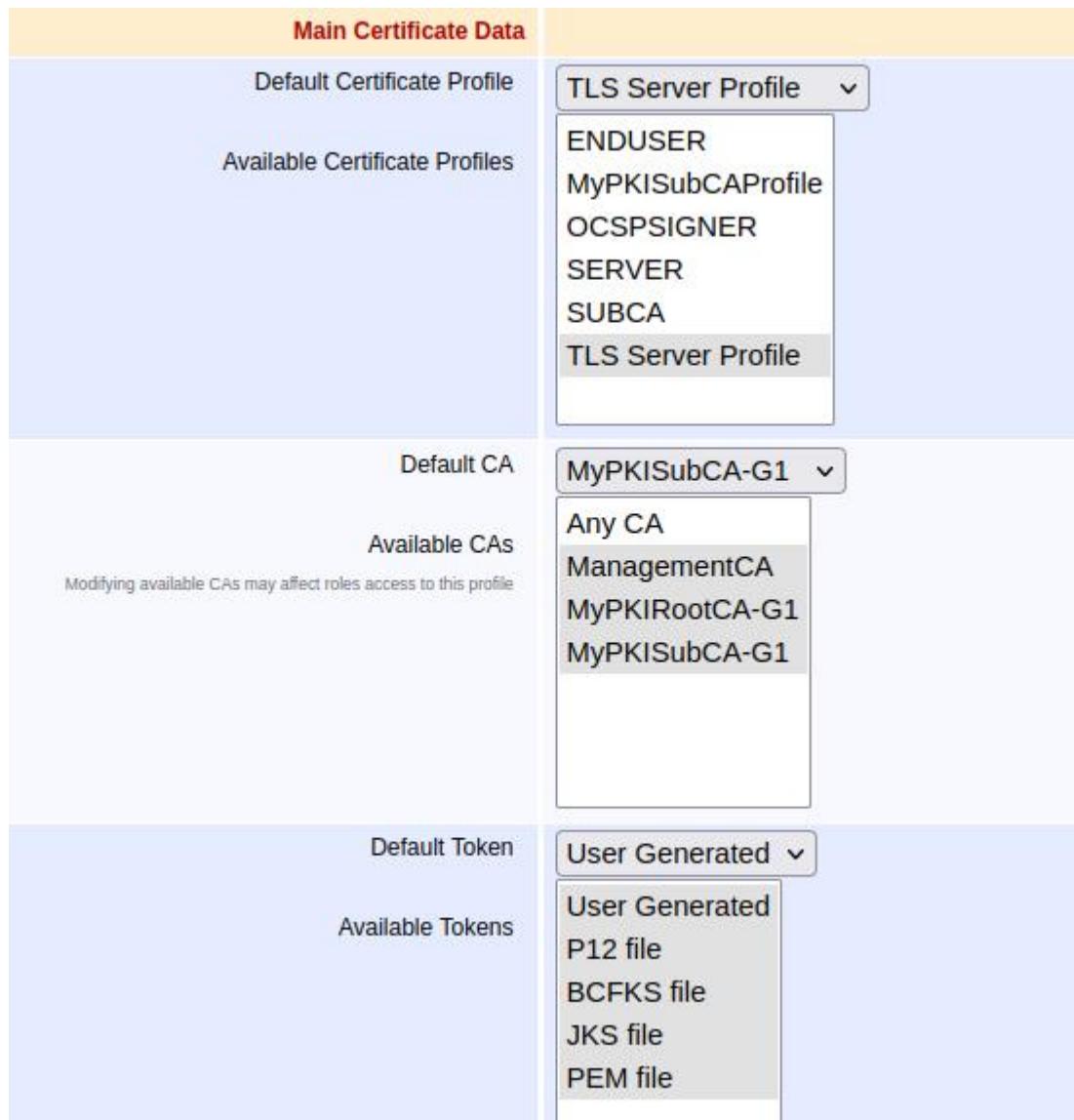
- Trong trường *Add Profile*, thêm tên cho hồ sơ mới, ví dụ *TLS Server Profile*, và nhấp vào *Add profile*.
- Chọn hồ sơ *TLS Server Profile* mới tạo và nhấp vào *Edit End Entity Profile* để cập nhật hồ sơ.

- Chính sửa hồ sơ và cập nhật các thông số sau:
 - Xóa **End Entity E-mail** để không lưu bất kỳ địa chỉ email nào.
 - Dưới **Subject DN Attributes**, xác định các thông tin: CN (bắt buộc), O ,C
- Trong danh sách **Subject Alternative Name**, chọn **DNS Name** và nhấp **Add**. Chọn **Use entity CN field** để thêm một tên DNS giống với trường tên chung của bạn. Thêm các tên DNS tùy chọn để cho phép nhiều hơn một tên DNS trong chứng chỉ.

Other Subject Attributes	
Subject Alternative Name	<input type="button" value="DNS Name"/> <input type="button" value="Add"/>
DNS Name	<input checked="" type="checkbox"/> Use entity CN field <input type="text"/> <input type="checkbox"/> Required <input checked="" type="checkbox"/> Modifiable <input type="checkbox"/> Validation <input type="text"/>
DNS Name	<input type="checkbox"/> Use entity CN field <input type="text"/> <input type="checkbox"/> Required <input checked="" type="checkbox"/> Modifiable <input type="checkbox"/> Validation <input type="text"/>
DNS Name	<input type="checkbox"/> Use entity CN field <input type="text"/> <input type="checkbox"/> Required <input checked="" type="checkbox"/> Modifiable <input type="checkbox"/> Validation <input type="text"/>
DNS Name	<input type="checkbox"/> Use entity CN field <input type="text"/> <input type="checkbox"/> Required <input checked="" type="checkbox"/> Modifiable <input type="checkbox"/> Validation <input type="text"/>

Hình 53 Cấu hình DNS cho TLS Server

- Trong *Main Certificate Data*, bạn có thể xác định hồ sơ mặc định cho việc cấp chứng chỉ:
 - Đối với *Default Certificate Profile*, chọn *TLS Server Profile* đã tạo trong phần a.
 - Đối với *Default CAs*, chọn *MyPKISubCA-G1* để hạn chế hồ sơ này chỉ có thể sử dụng với Sub CA của bạn.



Hình 54 Cấu hình chứng chỉ nhận, người cấp chứng chỉ và định dạng nhận chứng chỉ

- Nhập **Save** để lưu hồ sơ thực thể cuối.

3.2.2.4 Phân quyền trong EJBCA CE

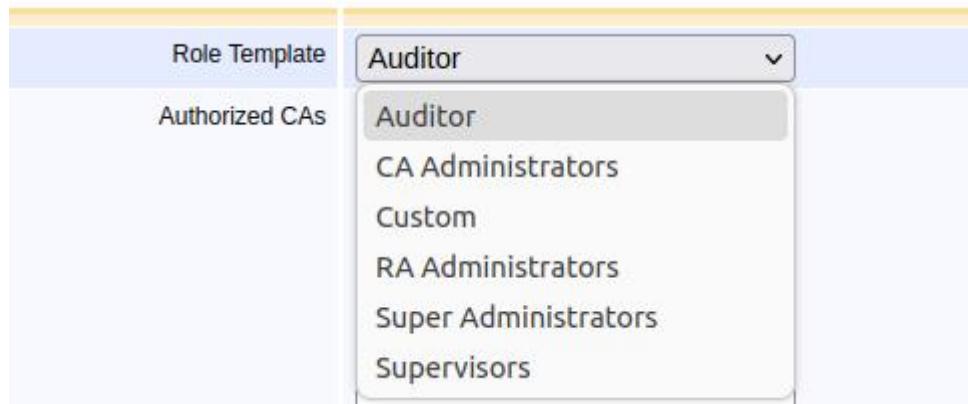
Trong hệ thống PKI hỗ trợ bởi EJBCA-CE, ta có thể phân quyền cho các thiết bị đầu cuối, được thiết kế để quản lý quyền truy cập hệ thống một cách linh hoạt cũng như nâng cao tính bảo mật của hệ thống PKI

EJBCA-CE phân quyền người dùng với cấu trúc RBAC (Role-Based Access Control), trong đó mỗi người dùng hoặc nhóm người dùng sẽ được gán với một vai trò (Roles) và được cấp các quyền (Access Rules) nhất định với hệ thống PKI.

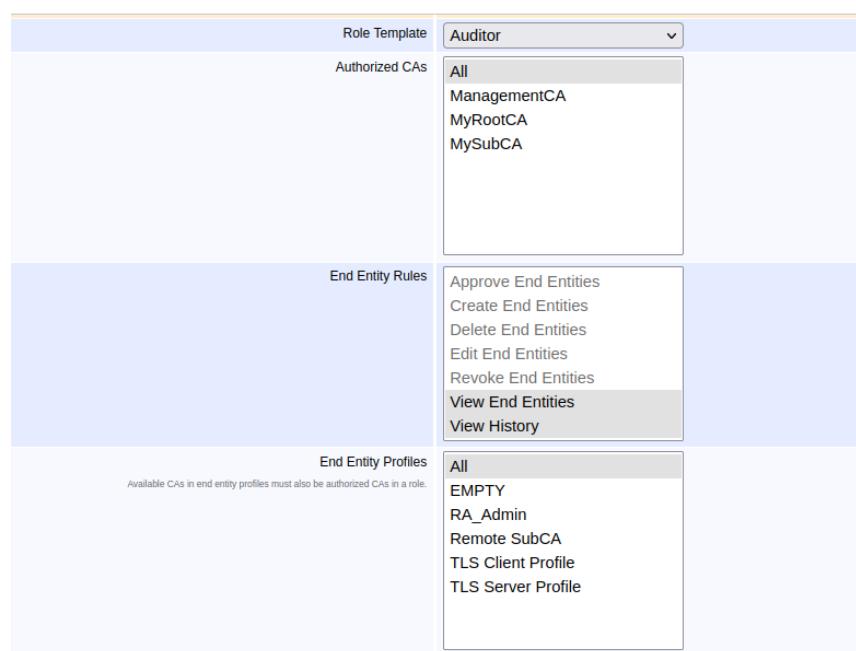
EJBCA-CE cung cấp một số vai trò quản trị mặc định như:

- Auditor (Kiểm toán viên): Có quyền truy cập đầy đủ mọi thông tin trong hệ thống EJBCA để giám sát hệ thống. Tuy nhiên, vai trò này chỉ có quyền đọc chứ không có quyền cấp hoặc phê duyệt chứng chỉ.

- CA Administrators: Quản lý CA, bao gồm việc cấu hình và phát hành chứng chỉ, nhưng không xử lý yêu cầu cấp chứng chỉ từ client.
- RA Administrators: Phê duyệt và xử lý các yêu cầu cấp chứng chỉ từ client.
- Super Administrators: Vai trò có toàn quyền hệ thống, gồm cả quản lý CA và RA.
- Supervisors (Người giám sát): Quản lý mọi hoạt động của CA, tuy nhiên chỉ có quyền đọc chứ không có quyền trực tiếp cấp chứng chỉ



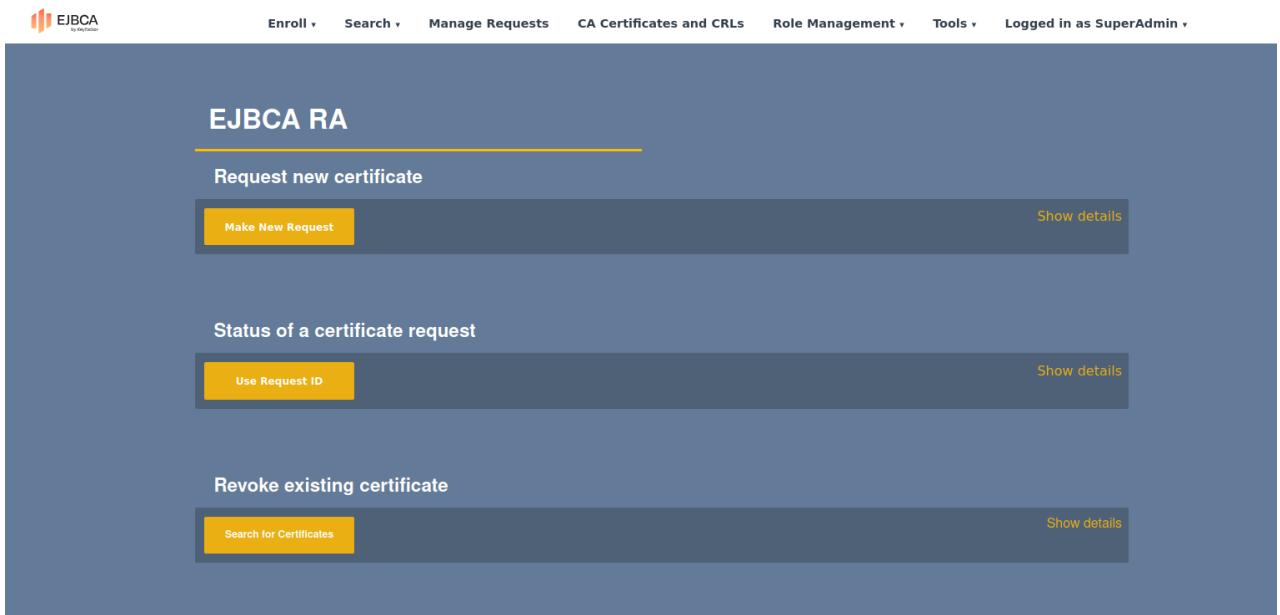
Hình 55 Các roles mặc định trong EJBCA



Hình 56 Một số quyền được thiết lập trong EJBCA

3.2.2.5 Giao diện quản lý RA

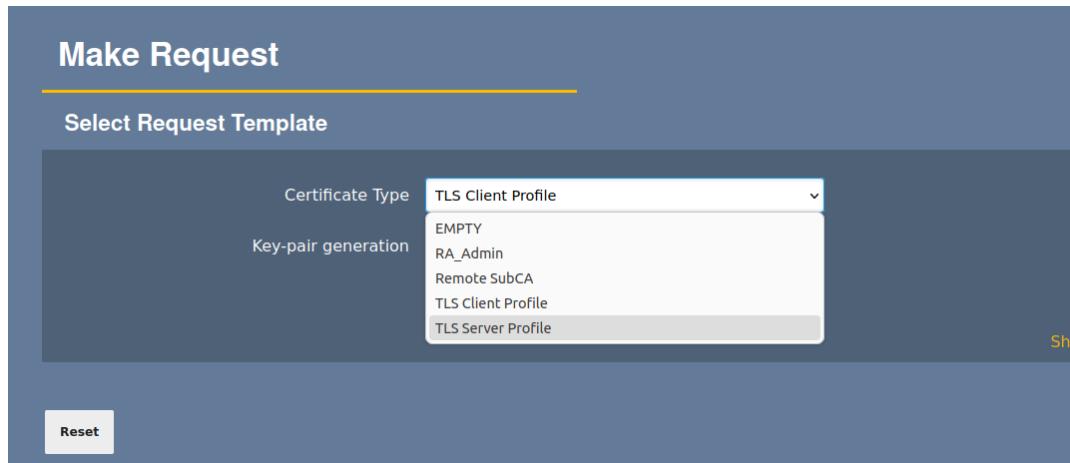
Với một tài khoản được cấp quyền quản lý RA, ta làm việc trên một giao diện GUI



Hình 57 Giao diện quản lý RA

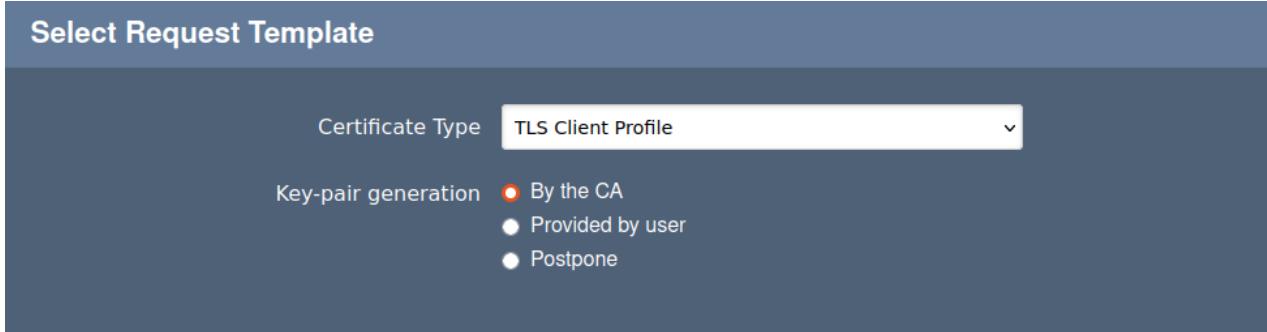
Tại giao diện này, ta có thể tạo chứng chỉ mới cho client, kiểm tra các chứng chỉ hiện có, các thiết bị đầu cuối được xác thực trong hệ thống PKI, kiểm tra các Roles và nhóm người dùng được cấu hình trong hệ thống.

Tại giao diện tạo một chứng chỉ mới, ta cần chọn chứng chỉ này dùng để xác thực thực thể đầu cuối nào được cấu hình trước bởi CA.



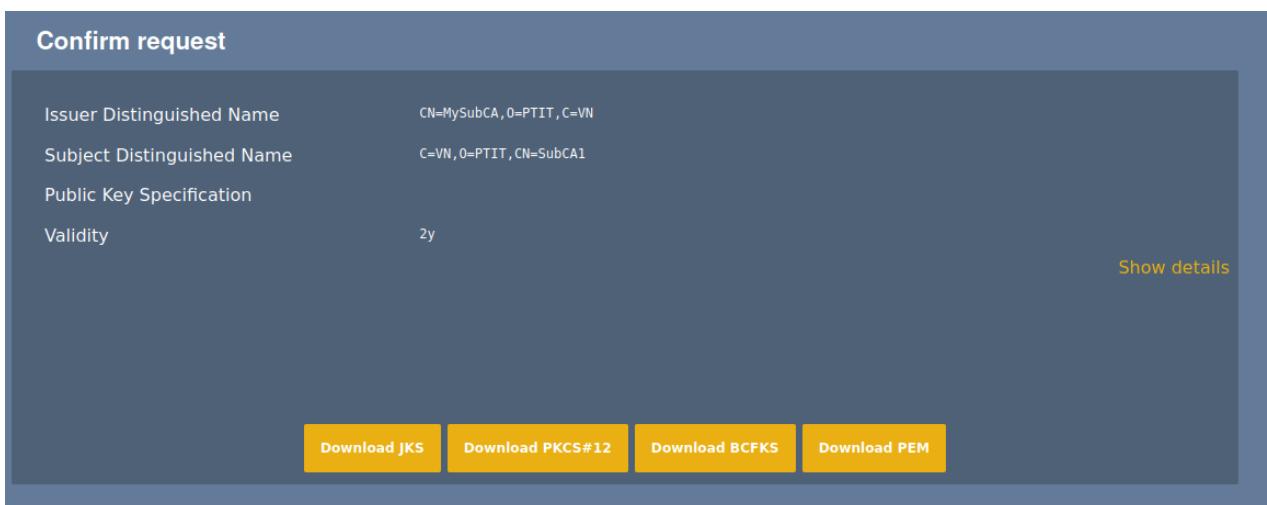
Hình 58 Chọn profile cho chứng chỉ mới

Tiếp theo ta cần quyết định xem cặp khóa sẽ được CA tạo ra hay do user tạo sẵn private key với file .csr (Certificate Signing Request) gửi cho RA để tạo khóa công khai đính kèm chứng chỉ.



Hình 59 Chọn cách tạo cặp khóa

Sau đó ta nhập vào các trường thông tin cần thiết và xác nhận lại các thông tin của chứng chỉ và tải về chứng chỉ được tạo. (***Lưu ý:** Các trường CN – Common name và username không được trùng lặp giữa các thực thể khác được cấp chứng chỉ trong hệ thống)



Hình 60 Xác nhận lại thông tin chứng chỉ

3.2.3 Demo cấp chứng chỉ cho SubCA

Tại RootCA, tạo một profile mới để cấu hình cho SubCA

Manage Certificate Profiles

List of Certificate Profiles

Name	Actions
ENDUSER	View Edit Delete Rename Clone Export
OCSPSIGNER	View Edit Delete Rename Clone Export
ROOTCA	View Edit Delete Rename Clone Export
SERVER	View Edit Delete Rename Clone Export
SUBCA	View Edit Delete Rename Clone Export
MyRootCAProfile	View Edit Delete Rename Clone Export
MySubCAProfile	View Edit Delete Rename Clone Export
RA_Admin	View Edit Delete Rename Clone Export
TLS Client Profile	View Edit Delete Rename Clone Export
TLS Server Profile	View Edit Delete Rename Clone Export
	Add

Import/Export

Hình 61 Danh sách các profile được CA cấu hình

Các bước cấu hình cho SubCA được phân tích rõ tại *phần 3.2.2.2 (a)*. Sau đó, nếu muốn cho một user khác được cấu hình trên hệ thống EJBCA như một SubCA, ta cần tạo định nghĩa 1 thiết bị đầu cuối (End Entity Profiles)

Hình 62 Danh sách các hồ sơ định nghĩa thiết bị đầu cuối

Chọn **Edit End Entity Profiles** để định nghĩa cho hồ sơ SubCA, với template được định sẵn bởi EJBCA, ta cần chú ý đến phần sau:

Hình 63 Giao diện định nghĩa hồ sơ thiết bị đầu cuối

Ở đây với mục đầu tiên, ta cần định hướng thiết bị đầu cuối đến CA Profile được tạo bởi CA, ở đây là **MySubCAProfile**

Tiếp theo, ở mục Available CAs, ta định nghĩa CA nào đang được hoạt động sẽ cung cấp chứng chỉ cho hồ sơ thiết bị đầu cuối này. Các CA này được hướng dẫn cấu hình tại mục 3.2.2.2 (b) và 3.2.2.2 (c). Ở đây, ta cấu hình SubCA sẽ được cấp chứng chỉ bởi RootCA.

Cuối cùng định nghĩa các dạng token có thể được sử dụng, ở đây ta dùng token do user tạo hoặc dưới định dạng PKCS#12

Tiếp theo ta sang trang web cho RA để tạo chứng chỉ cho user. Ta chọn profile CA là Remote SubCA vừa định nghĩa và chọn CA cấp chứng chỉ cho SubCA này.

EJBCA's RA GUI

Make New Request

Use Request ID
Use Username

st

Select Request Template

Certificate Type: Remote SubCA

CA: ManagementCA (default)

Key-pair generation: By the CA

Show details

Provide request info

Required Subject DN Attributes

Common Name (CN) *: SubCA1

Organization (O) *: PTIT

Country [ISO 3166] (C) = VN

Hình 64 Tạo chứng chỉ mới cho SubCA

Common Name (CN) *: SubCA1

Organization (O) *: PTIT

Country [ISO 3166] (C) = VN

Optional Subject Alternative Name Attributes

DNS Name: Use data from Common Name (CN) field

DNS Name:

Show more optional fields

Provide User Credentials

Username *: sub_ca1

Enrollment code *:

Confirm enrollment code *:

Email:

Hình 65 Điện thông tin cho chứng chỉ SubCA

Ở đây ta cần nhập miền Common Name và username để phục vụ cho việc phân quyền người dùng và Enrollment code để khóa file chứng chỉ mới tạo.

Tiếp theo ta tạo nhóm user SubCA, thêm người dùng mới cho nhóm người dùng nhận dạng bằng CN và CA cấp chứng chỉ cho nó. Ở đây là SubCA1.

Members

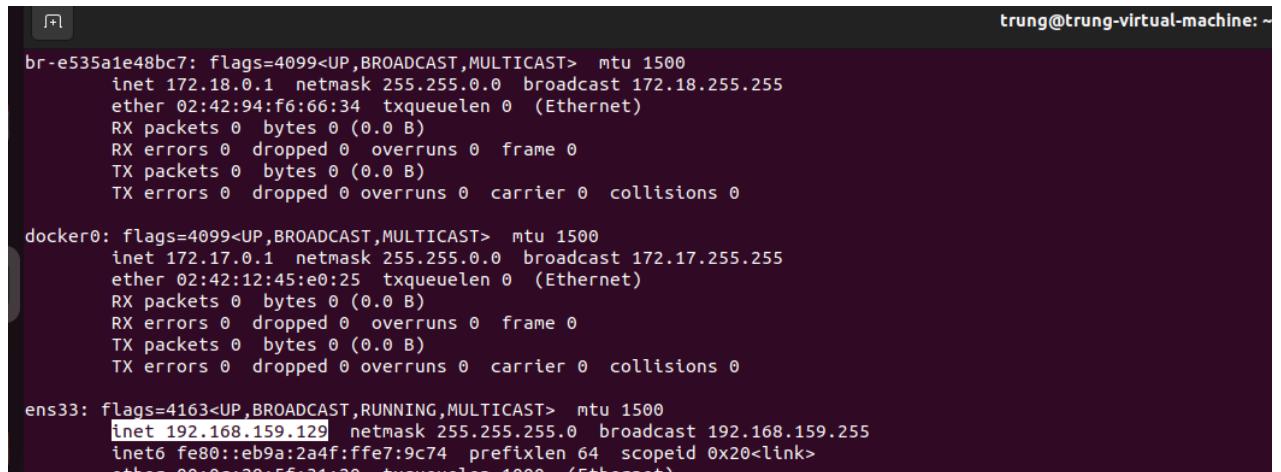
Role : SubCA

Match with	CA	Match Operator	Match Value
X509: Certificate serial number (Recommended)	ManagementCA		
X509: CN, Common name	ManagementCA	Equal, case sens.	SubCA1

© 2002–2024. EJBCA® is a registered trademark.

Hình 66 Nhóm người dùng SubCA

Lấy ip của máy chủ chạy dịch vụ EJBCA để máy khác trong mạng LAN có thể truy cập và thao tác trên hệ thống.



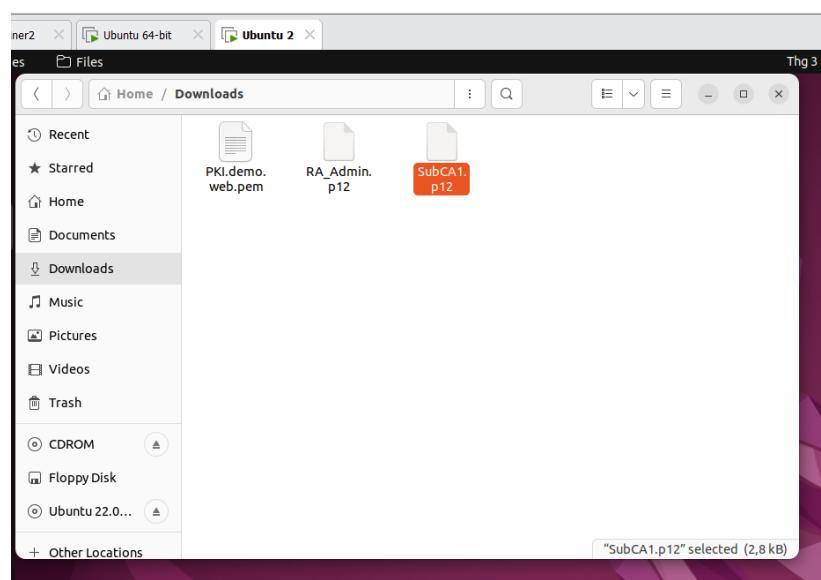
```
trung@trung-virtual-machine: ~
br-e535a1e48bc7: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.18.0.1 netmask 255.255.0.0 broadcast 172.18.255.255
        ether 02:42:94:f6:66:34 txqueuelen 0 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
        ether 02:42:12:45:e0:25 txqueuelen 0 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.159.129 netmask 255.255.255.0 broadcast 192.168.159.255
        inet6 fe80::eb9a:2a4f:ffe7:9c74 prefixlen 64 scopelid 0x20<link>
            ether 00:0c:29:5f:31:20 txqueuelen 1000 (Ethernet)
```

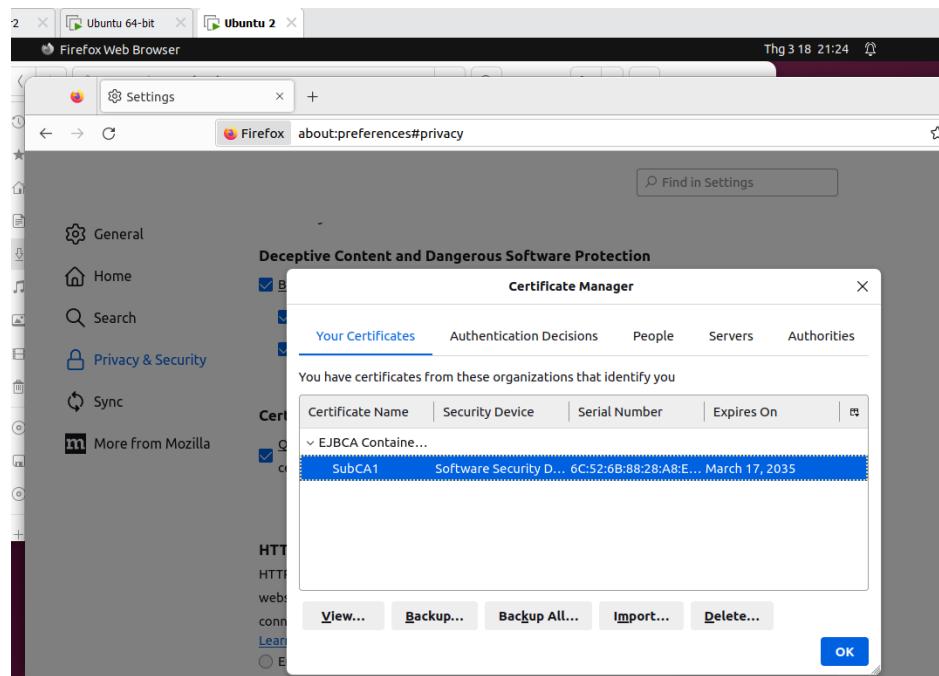
Hình 67 IP máy chủ EJBCA

Tại máy đầu cuối khác để làm user SubCA, ta lấy được file chứng chỉ xác thực lấy được từ RA.



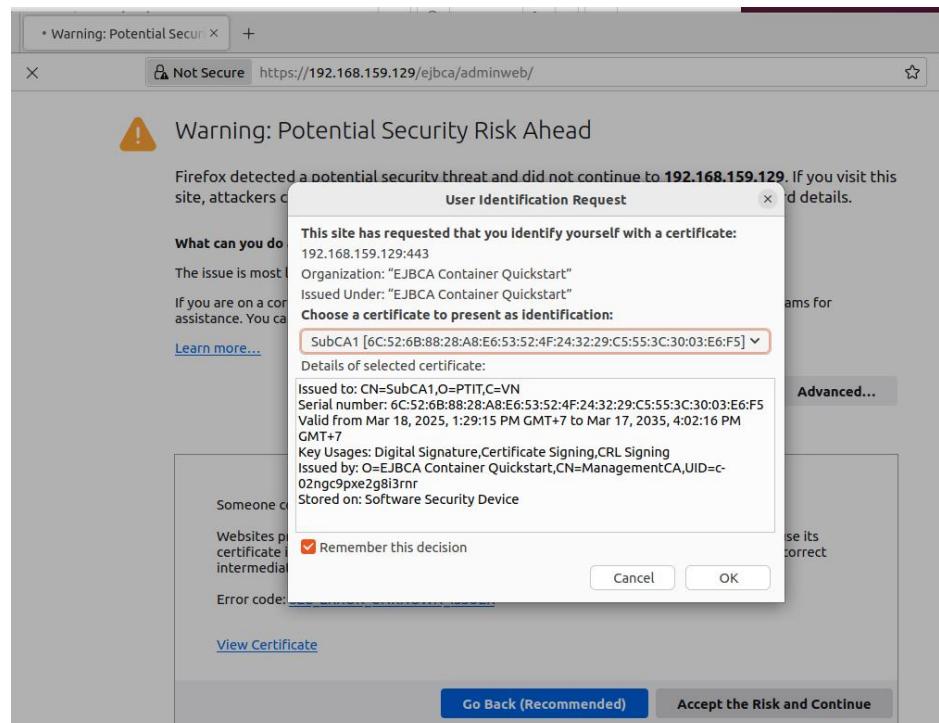
Hình 68 File chứng chỉ của SubCA1

Tiếp theo, import file chứng chỉ vào trình duyệt, nhập mật khẩu là enrollment code đã thiết lập ở RA Web



Hình 69 Thêm chứng chỉ của SubCA1 vào trình duyệt

Tiếp theo truy cập vào địa chỉ IP của máy chủ EJBCA, dùng chứng chỉ vừa thêm vào trình duyệt để xác thực, lấy được quyền SubCA và hệ thống.



Hình 70 Dùng chứng chỉ xác thực người dùng EJBCA

Ta vào được hệ thống EJBCA với username là SubCA1

The screenshot shows the EJBCA Administration interface in a browser window. The title bar says "EJBCA Administration". The URL is "https://192.168.159.129/ejbca/adminweb/". The page header includes the EJBCA logo and navigation links for Home, CA Functions, RA Functions, VA Functions, and Supervision Functions. Below the header, it displays the version "Version : EJBCA 9.0.0 Community (7fb810e779ded2d8152e89a6d47e83d5fdcf8e4a)", a welcome message "Welcome SubCA1 to EJBCA Administration.", and node information "Node hostname ejbca-node1" and "Server time 2025-03-18 14:29:08+00:00". Two tables are present: "CA Status" and "Publisher Queue Status". The "CA Status" table has two rows: ManagementCA and MySubCA, both with green checkmarks in all columns. The "Publisher Queue Status" table has one row: "No publishers defined." A copyright notice at the bottom right reads "© 2002–2024. EJBCA® is a registered trademark."

Hình 71 Giao diện EJBCA trên SubCA1

3.2.4 Cấp quyền TLS server cho bên thứ 3

Trước tiên, ta cấu hình một webserver sử dụng dịch vụ apache2. Đầu tiên ta tạo một file **index.html** thiết kế giao diện web cơ bản trong **/var/www/html/**

The screenshot shows a web browser window titled "Trang Web Cơ Bản" with the URL "localhost". The page content is a simple website with three main sections: "Trang Chủ", "Giới Thiệu", and "Liên Hệ". Each section contains a brief description. The "Trang Chủ" section says "Đây là trang web mẫu của tôi.". The "Giới Thiệu" section says "Đây là phần giới thiệu về tôi và công ty của tôi.". The "Liên Hệ" section says "Để liên hệ, vui lòng gửi email tới contact@example.com".

Hình 72 Giao diện web trên Webserver

Tiếp theo ta định nghĩa thông tin sẽ được ghi trên chứng chỉ của Webserver và xác định phương thức mã hóa trong file **tls_server_req.cnf**

```
1 [req]
2 default_md = sha256
3 prompt = no
4 distinguished_name = dn
5 req_extensions = v3_req
6
7 [dn]
8 CN = PKI.demo.web
9 O = PTIT
10 C = VN
11
12 [v3_req]
13 keyUsage = critical, digitalSignature, keyEncipherment
14 extendedKeyUsage = serverAuth
15 subjectAltName = @alt_names
16
17 [alt_names]
18 DNS.1 = PKI.demo.web
19 DNS.2 = 192.169.159.136
20 IP.1 = 192.168.159.136
```

Hình 73 File định nghĩa thông tin Webserver

Sau đó dùng openssl để khởi tạo private key và file yêu cầu cấp chứng chỉ .csr với đầu vào là file định nghĩa trên, sử dụng thuật toán mã hóa ECDSA prime256v1 để trùng khớp với thuật toán mã hóa được định nghĩa trong profile CA

```
student@LabtainerVMware:~/tls_req$ cat gen_key.txt
$ openssl ecparam -genkey -name prime256v1 -out tls_server.key

$ openssl req -new -key tls_server.key -config tls_cert_req.cnf
student@LabtainerVMware:~/tls_req$
```

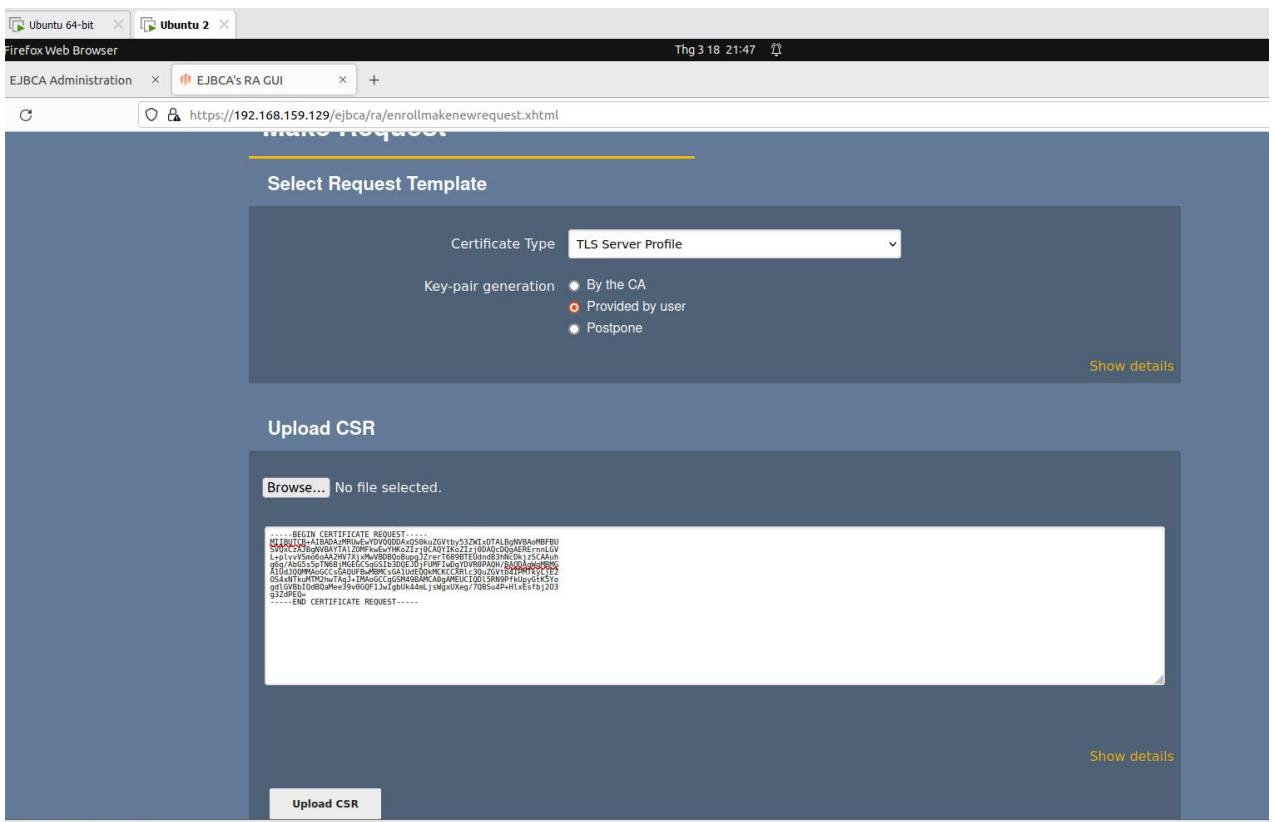
Hình 74 Lệnh khởi tạo private key

Sau khi khởi tạo, ta có được 1 file private key và một file .csr để gửi cho RA tạo chứng chỉ mới

```
student@LabtainerVMware:~/tls_req$ ls
gen_key.txt  PKI.demo.web.pem  tls_server.csr  tls_server.key  tls_server_req.cnf
student@LabtainerVMware:~/tls_req$
```

Hình 75 File được tạo ra từ file định nghĩa Webserver

Tại SubCA, ta tạo chứng chỉ mới với profile TLS Server và lựa chọn cặp khóa tạo bởi người dùng, gán nội dung file tls_server.csr vào



Hình 76 Khởi tạo chứng chỉ mới cho Webserver tại RA Web

Các thông tin được định nghĩa trước sẽ được tự động ghi vào chứng chỉ, ta cần gán cho thực thể này một username riêng biệt

Provide User Credentials	
Username *	pki.demo.web
The username 'pki.demo.web' already exists	
Confirm request	
Issuer Distinguished Name	CN=MySubCA, O=PTIT, C=VN
Subject Distinguished Name	C=VN, O=PTIT, CN=PKI.demo.web
Subject Alternative Name	DNSNAME=test.demo, DNSNAME=192.169.159.136, DNSNAME=PKI.demo.web
Public Key Specification	ECDSA prime256v1
Validity	2y

Hình 77 Thông tin chứng chỉ Webserver

Sau đó ta tải file chứng chỉ về dưới định dạng PEM full chain

```
$ openssl req -new -key tls_server.key -signing tls_cert_req.pem  
student@LabtainerVMware:~/tls_req$ ls  
gen_key.txt  PKI.demo.web.pem  tls_server.csr  tls_server.key  tls_server_req.cnf  
student@LabtainerVMware:~/tls_req$
```

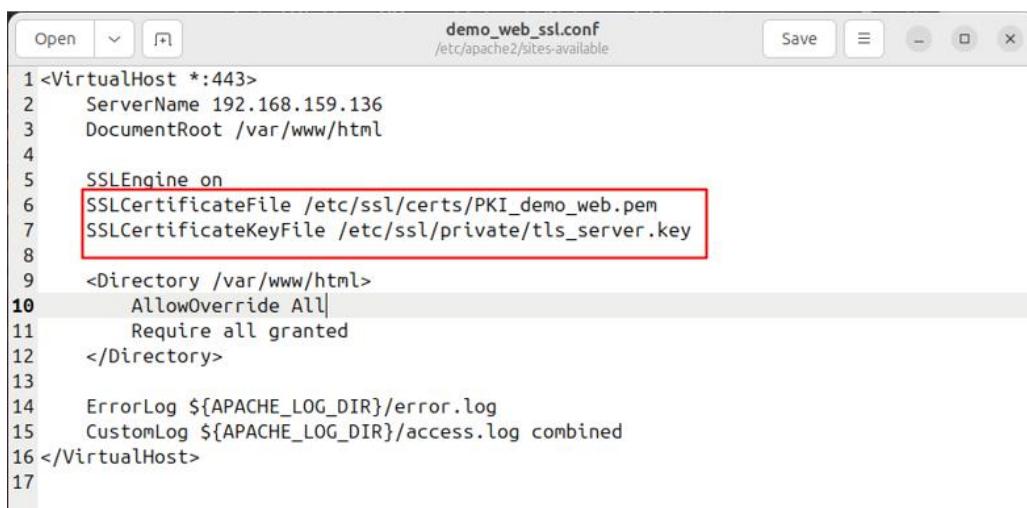
Hình 78 File chứng chỉ được tạo

Kích hoạt module ssl để webserver trên apache2

```
student@LabtainerVMware:~/tls_req$ sudo a2enmod ssl  
[sudo] password for student:  
Considering dependency mime for ssl:  
Module mime already enabled  
Considering dependency socache_shmcb for ssl:  
Module socache_shmcb already enabled  
Module ssl already enabled  
student@LabtainerVMware:~/tls_req$ sudo systemctl restart apache2  
student@LabtainerVMware:~/tls_req$
```

Hình 79 Kích hoạt module SSL của apache2

Tại /etc/apache2/site-available/ tạo một file config cấu hình SSL cho apache



```
Open  Save  demo_web_ssl.conf  /etc/apache2/sites-available  
1 <VirtualHost *:443>  
2   ServerName 192.168.159.136  
3   DocumentRoot /var/www/html  
4  
5   SSLEngine on  
6   SSLCertificateFile /etc/ssl/certs/PKI_demo_web.pem  
7   SSLCertificateKeyFile /etc/ssl/private/tls_server.key  
8  
9   <Directory /var/www/html>  
10    AllowOverride All  
11    Require all granted  
12  </Directory>  
13  
14  ErrorLog ${APACHE_LOG_DIR}/error.log  
15  CustomLog ${APACHE_LOG_DIR}/access.log combined  
16 </VirtualHost>  
17
```

Hình 80 File config SSL cho Webserver

Sau đó kích hoạt file config và khởi động lại dịch vụ apache2

```
student@LabtainerVMware:/etc/apache2/sites-available$ sudo a2ensite demo_web_ssl.conf  
Site demo_web_ssl already enabled  
student@LabtainerVMware:/etc/apache2/sites-available$ sudo systemctl restart apache2  
Failed to restart apache2.service: Unit apache2.service not found.  
student@LabtainerVMware:/etc/apache2/sites-available$ sudo systemctl restart apache2  
student@LabtainerVMware:/etc/apache2/sites-available$
```

Hình 81 Kích hoạt file config SSL

Ta lấy IP của máy chủ Webserver để truy cập vào bằng máy khác trong mạng LAN

```

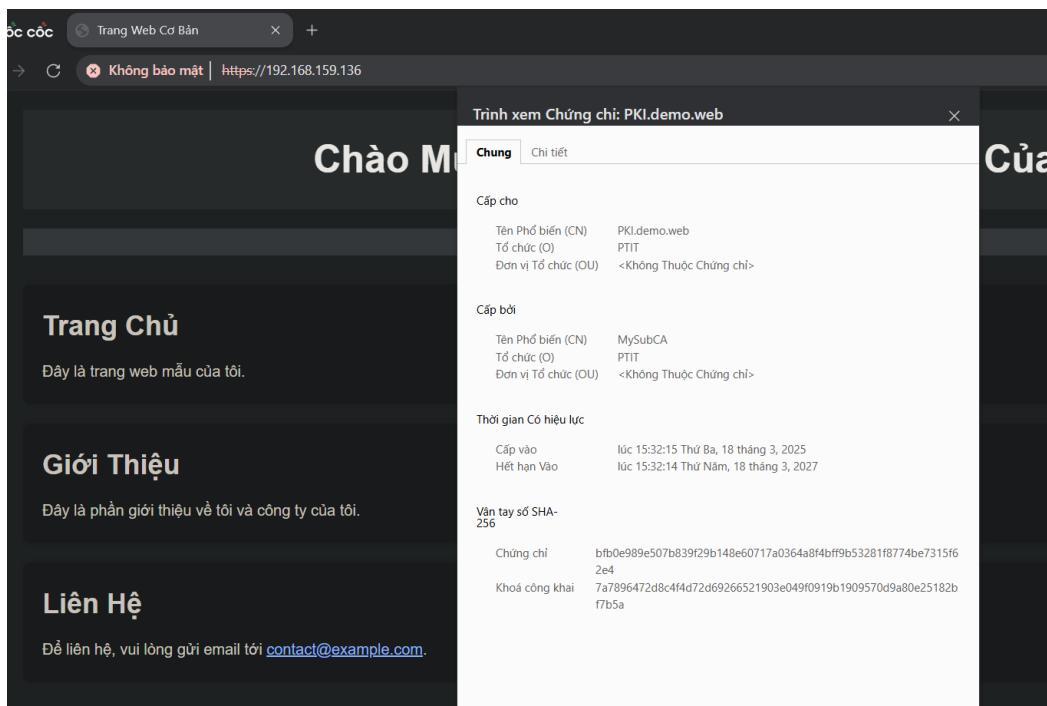
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 19 overruns 0 carrier 0 collisions 0

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST
      inet 192.168.159.136 netmask 255.255.255.0
      inet6 fe80::20c:29ff:fe06:b2f6 prefixlen 64
        ether 00:0c:29:06:b2:f6 txqueuelen 1000
        RX packets 6283 bytes 4663272 (4.6 MB)

```

Hình 82 IP máy chủ Webserver

Truy cập vào máy chủ Webserver qua IP trên và ta nhận thấy Webserver đã được cấp chứng chỉ



Hình 83 Webserver đã được cấp chứng chỉ

3.3 Kết chương

Chương này nói tổng quan về hệ thống EJBCA CE, mô tả việc triển khai và thử nghiệm hệ thống EJBCA CE qua các quy trình như cài đặt, cấu hình. Ngoài ra chúng ta cũng đã thử nghiệm các chức năng của hệ thống như tạo CA, phân quyền CA, cấp quyền, cấp chứng chỉ, cấp chứng chỉ TLS Server cho bên thứ ba.

KẾT LUẬN

Các kết quả đạt được (nêu các kết quả đã đạt được của BTL)

- Nhóm đã hoàn thành nghiên cứu về hạ tầng khóa công khai (PKI), chứng chỉ số và chữ ký số.
- Đã thực hiện thành công việc cài đặt, cấu hình và thử nghiệm hệ thống quản lý chứng chỉ số EJBCA CE.
- Triển khai các chức năng quản lý chứng chỉ, cấp chứng chỉ, phân quyền và quản lý khóa công khai trong hệ thống.
- Hệ thống EJBCA CE đã được thử nghiệm và chứng minh khả năng vận hành hiệu quả trong việc bảo mật giao dịch và trao đổi thông tin.
- Các kết quả đạt được từ việc triển khai hệ thống EJBCA CE đã cho thấy tính linh hoạt và hiệu quả của công cụ này trong việc xây dựng hệ thống bảo mật PKI cho các tổ chức.
- Nhóm cũng nhận thấy các yếu tố quan trọng trong việc đảm bảo tính toàn vẹn, bảo mật và xác thực trong môi trường mạng, như việc áp dụng các thuật toán mã hóa mạnh mẽ và các cơ chế bảo mật khác.

Hướng phát triển (nêu hướng phát triển, bổ sung, nghiên cứu tiếp của BTL)

Đề tài này có thể được mở rộng theo các hướng sau:

- Tìm hiểu thêm về ứng dụng của hệ thống EJBCA CE như xác thực người dùng, kết hợp với các hệ thống khác như SignServer để xác thực dữ liệu hơn

TÀI LIỆU THAM KHẢO

- [1] <https://hub.docker.com/r/keyfactor/ejbcache>
- [2] <https://viblo.asia/p/phan-2-gioi-thieu-ve-ejbcava-cach-cai-dat-ejbcata-tren-windows-GrLZDvEV5k0>
- [3] <https://www.ejbcna.org/>
- [4] [Top 5 Public Key Infrastructure \(PKI\) Pitfalls and How to Overcome Them - Spiceworks](#)
- [5] [Top 12 PKI Risks That Keep Security Professionals Up At Night](#)
- [6] [PKI Infrastructure: Attack, Detect & Defend | Curios](#)
- [7] [Vulnerability | PKI Consortium](#)
- [8] [PKI Management And Its Mistakes | Encryption Consulting](#)
- [9] <https://filelist.tudelft.nl/TBM/Over%20faculteit/Afdelingen/Engineering%20Systems%20and%20Services/People/Professors%20emeriti/Jan%20van%20den%20Berg/MasterPhdThesis/stefan.pdf>
- [10] <https://www.scribd.com/document/303188283/digital-signature-algorithm>
- [11] https://www.cryptomathic.com/hubfs/Documents/EBooks/Digital_Signatures_for_Dummies.pdf
- [12] <https://www.redbooks.ibm.com/redbooks/pdfs/sg248336.pdf>
- [13] <https://viblo.asia/p/co-ban-ve-chu-ky-so-chung-chi-so-va-pki-OeVKBoM2ZkW>
- [14] <https://lists.debian.org/debian-security-announce/2008/msg00152.html>
- [15] <https://letsencrypt.org/docs/dst-root-ca-x3-expiration-september-2021/>
- [16] <https://www.youtube.com/watch?v=5OqgYSXWYQM>
- [17] <https://threatpost.com/final-report-diginotar-hack-shows-total-compromise-ca-servers-103112/77170/>
- [18] <https://www.thesslstore.com/blog/pki-uses-applications-examples/>