Applied!

Data & Network Security

Behnam Amiri

ans.dailysec.ir

aNetSec.github.io

Spring 2025

Cryptographic Key Management

Cryptographic key management

- Cryptographic key algorithms depends on the protection of the cryptographic keys.
- All keys need to be protected against modification.
- Secret and private keys need to be protected against disclosure
- Cryptographic key management is the process of administering or managing cryptographic keys for a cryptographic system.
- It involves the generation, creation, protection, storage, exchange, replacement.

Symmetric Key Distribution Using Symmetric Encryption

1. A can select a key and physically deliver it to B.

2. A third party can select the key and physically deliver it to A & B.

3. If A & B have previously and recently used a key, one party can transmit the new key to the other, encrypted using the old key.

4. If A & B each has an encrypted connection to a third party C, C can deliver a key on the encrypted links to A and B.

Symmetric Key Distribution Using Asymmetric Encryption

1. A generates a public/private key pair {PUa, PRa} and transmits a message to B consisting of PUa and an identifier of A, IDA.

2. B generates a secret key, Ks, and transmits it to A, which is encrypted with A’s public key.

3. A computes D(PRa, E(PUa, Ks)) to recover the secret key. Because only A can decrypt the message, only A and B will know the identity of Ks.

4. A discards PUa and PRa and B discards PUa.

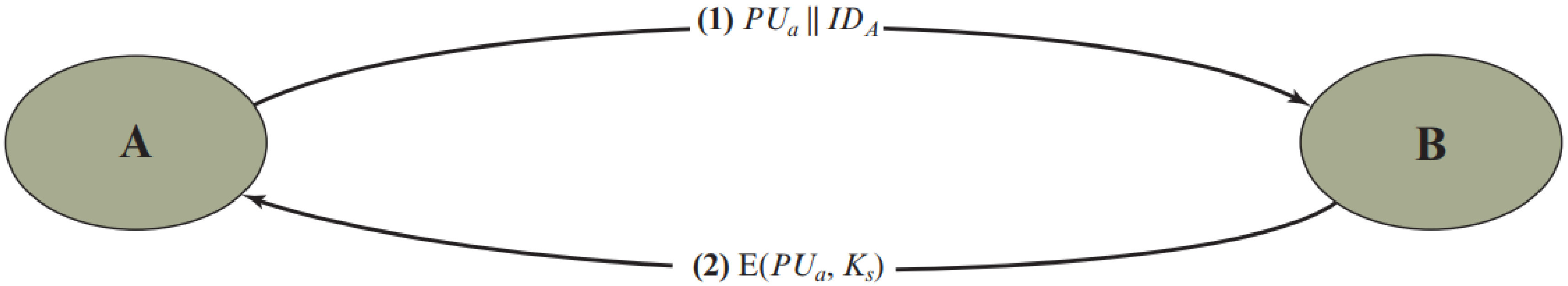


Figure 15.3 Simple Use of Public-Key Encryption to Establish a Session Key

Another MitM Attack

1. A generates a public/private key pair {*PUa*, *PRa*} and transmits a message for B consisting of *PUa* and an identifier of A, *IDA*.

2. D intercepts the message, creates its own public/private key pair {*PUd*, *PRd*} and transmits *PUd* } *IDA* to B.

3. B generates a secret key, *Ks*, and transmits E(*PUd*, *Ks*).

4. D intercepts the message and learns *Ks* by computing D(*PRd*, E(*PUd*, *Ks*)).

5. D transmits E(*PUa*, *Ks*) to A.

The result is that both A and B know *Ks* and are unaware that *Ks* has also been revealed to D .

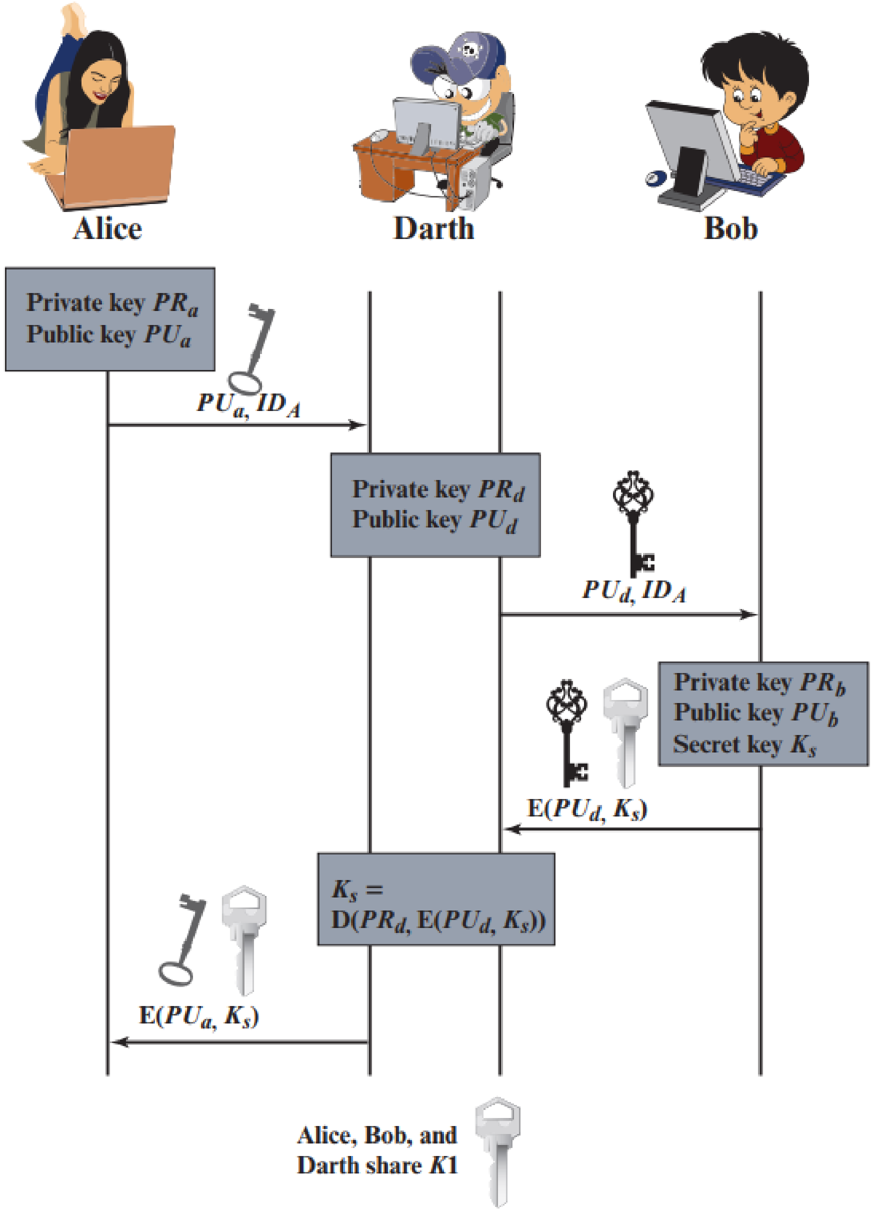


Figure 15.4 Another Man-in-the-Middle Attack

Secret Key Distribution with Confidentiality and Authentication

1. A uses B’s public key to encrypt a message to B containing an identifier of A(IDA) and a nonce ($N1$), which is used to identify this transaction uniquely.

2. B sends a message to A encrypted with PUa and containing A’s nonce ($N1$) as well as a new nonce generated by B ($N2$). Because only B could have decrypted message (1), the presence of $N1$ in message (2) assures A that the correspondent is B.

3. A returns $N2$, encrypted using B’s public key, to assure B that its correspondent is A.

4. A selects a secret key Ks and sends M = E(PUb, E(PRa, Ks)) to B. Encryption of this message with B’s public key ensures that only B can read it; encryption with A’s private key ensures that only A could have sent it.

5. B computes D(PUa, D(PRb, M)) to recover the secret key.

The result is that this scheme ensures both confidentiality and authentication in the exchange of a secret key.

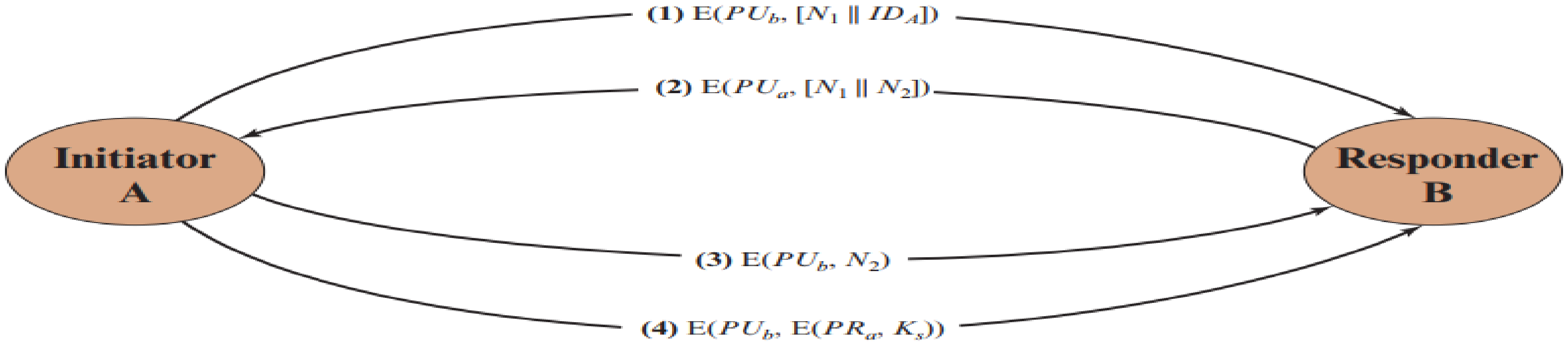

(1) $E(PU_b, [N_1 \| ID_A])$
(2) $E(PU_a, [N_1 \| N_2])$
Initiator A
Responder B
(3) $E(PU_b, N_2)$
(4) $E(PU_b, E(PR_a, K_s))$

Figure 15.5 Public-Key Distribution of Secret Keys

Distribution Of Public Keys

- Public announcement
- Publicly available directory
- Public-key authority
- Public-key certificate

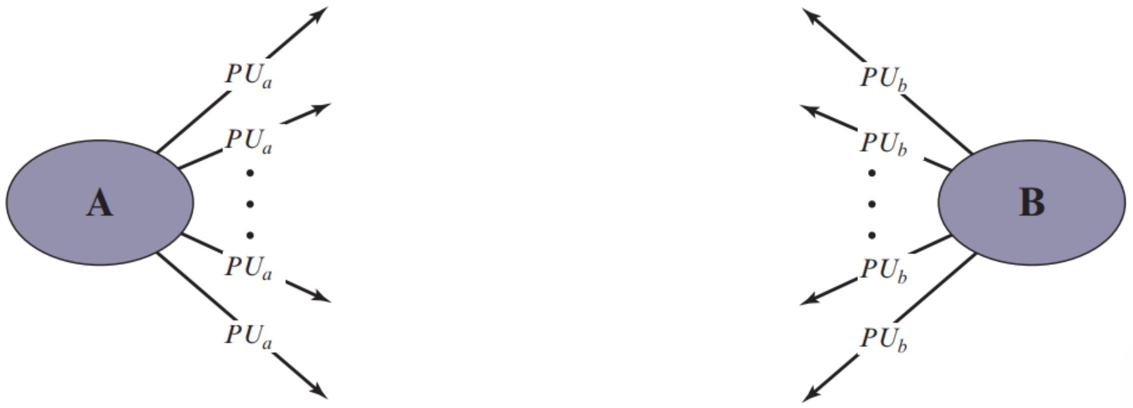


Figure 15.6 Uncontrolled Public-Key Distribution

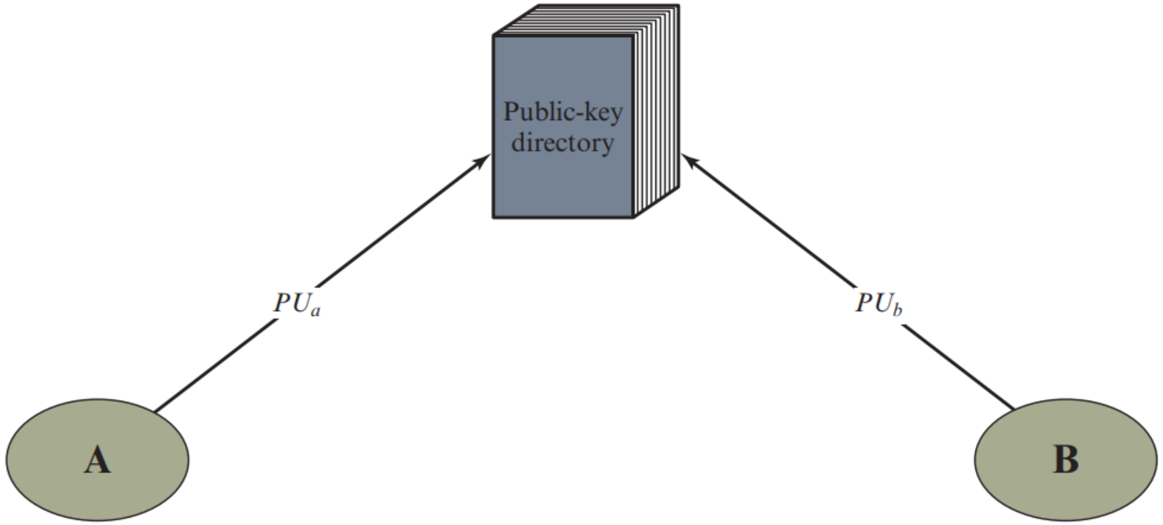


Figure 15.7 Public-Key Publication

X.509

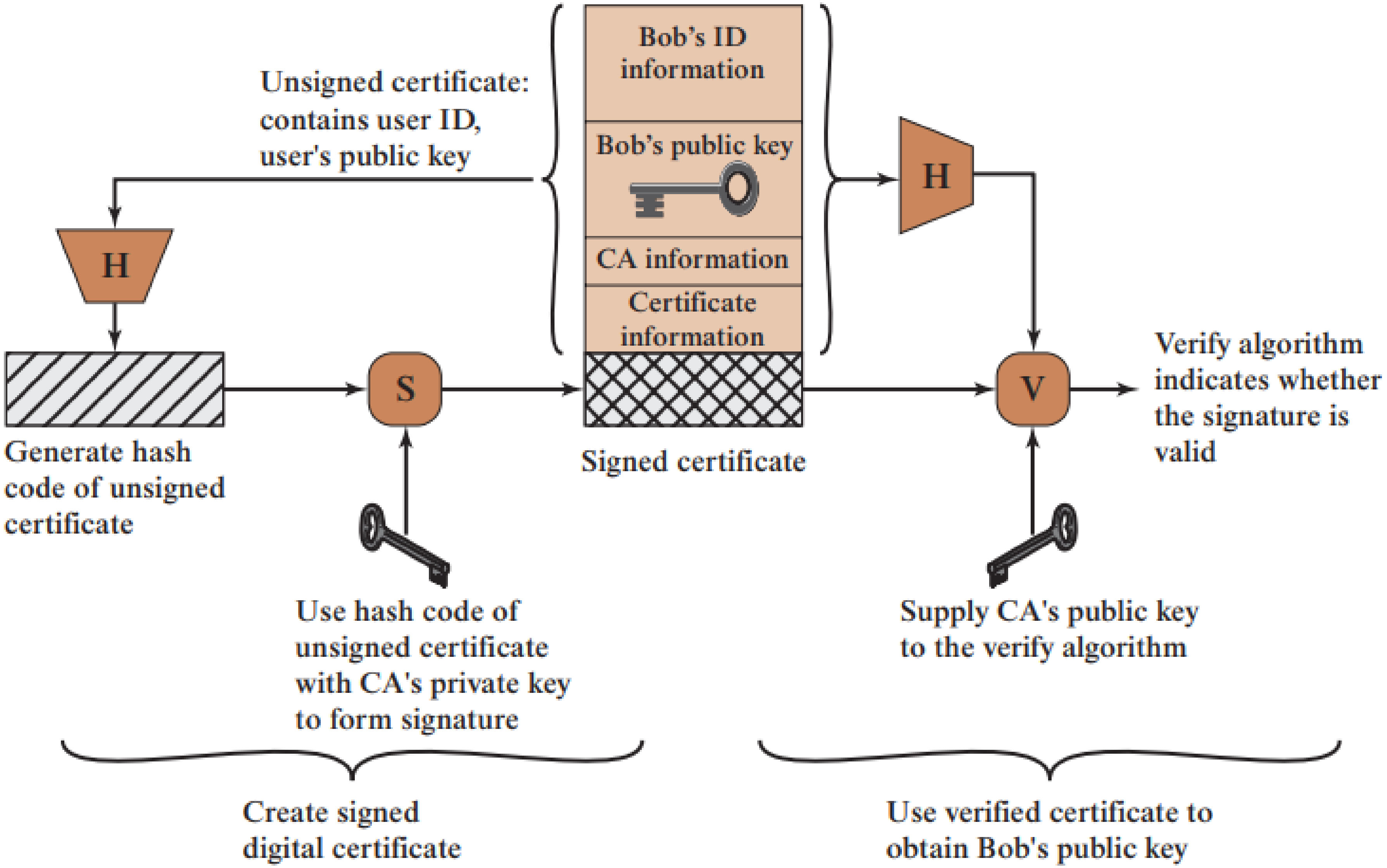
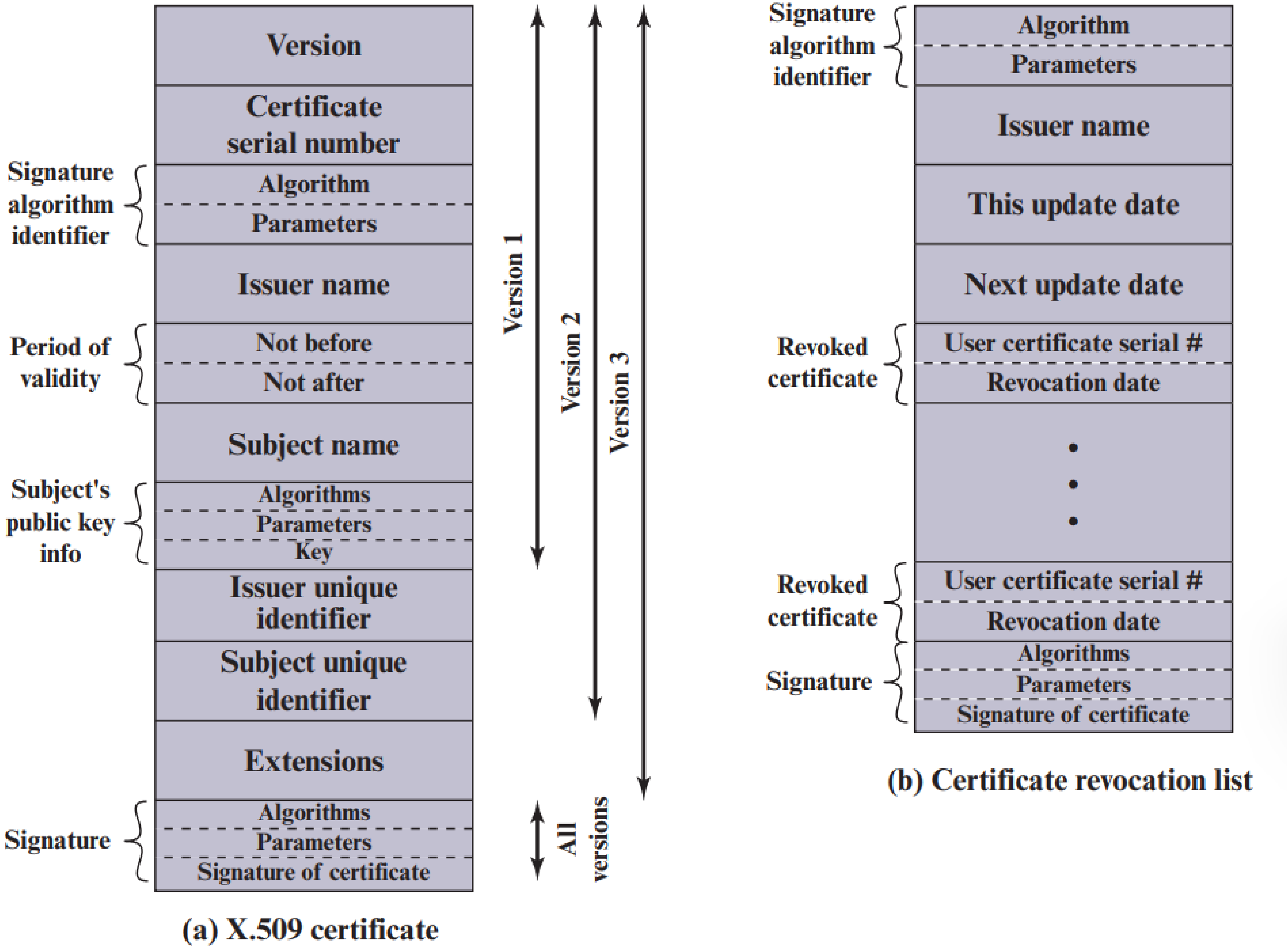

Unsigned certificate:
contains user ID,
user's public key
Bob's ID information
Bob's public key
CA information
Certificate information
H
H
S
V
Generate hash code of unsigned certificate
Signed certificate
Verify algorithm indicates whether the signature is valid
Use hash code of unsigned certificate with CA's private key to form signature
Supply CA's public key to the verify algorithm
Create signed digital certificate
Use verified certificate to obtain Bob's public key

Figure 15.10 X.509 Public-Key Certificate Use

X.509

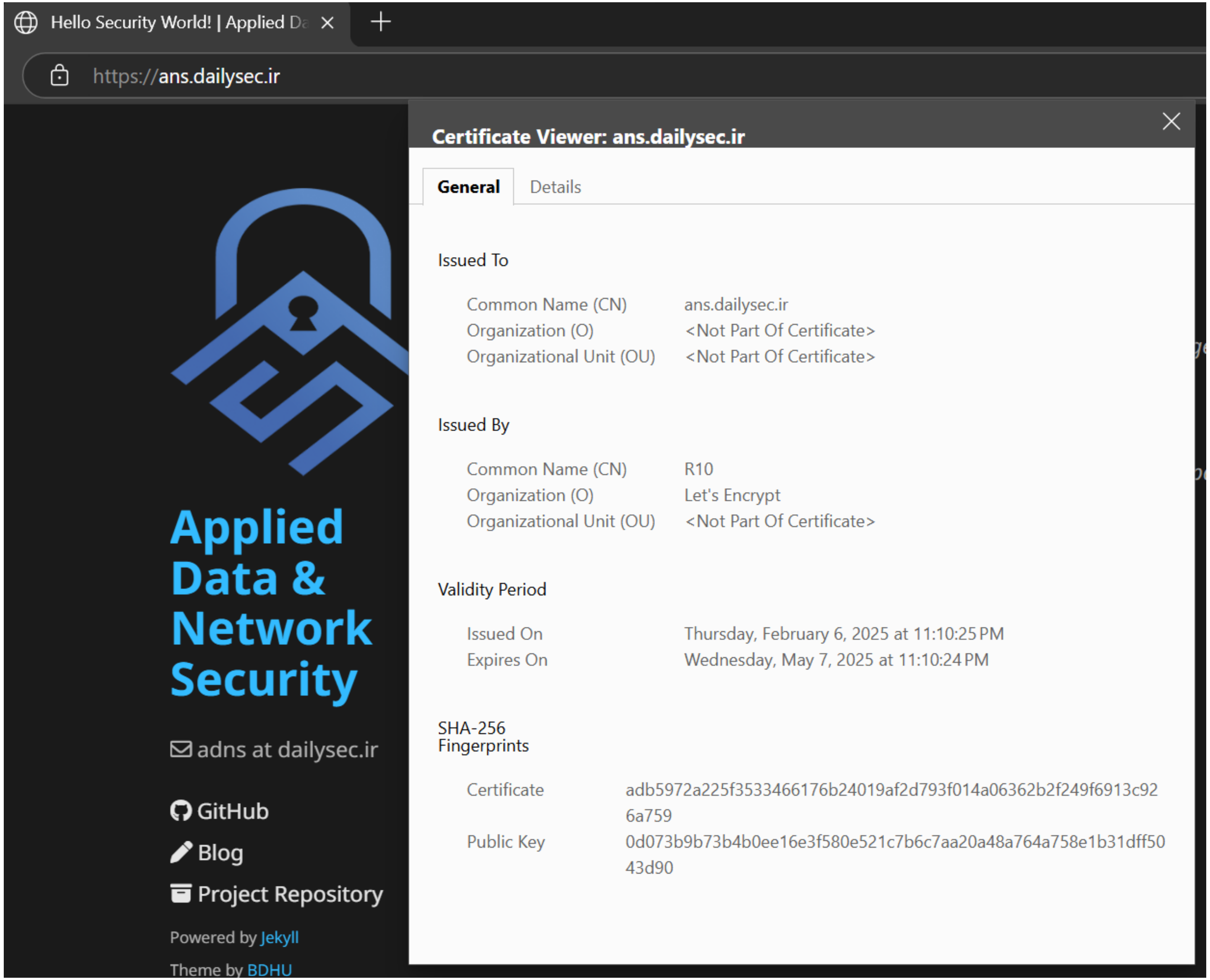


(a) X.509 certificate

(b) Certificate revocation list

Figure 15.11 X.509 Formats

Real Example



Applied Data & Network Security

adns at dailysec.ir

GitHub

Blog

Project Repository

Powered by Jekyll

Theme by BDHU

Certificate Viewer: ans.dailysec.ir

Issued To

Common Name (CN)	ans.dailysec.ir
Organization (O)	<Not Part Of Certificate>
Organizational Unit (OU)	<Not Part Of Certificate>

Issued By

Common Name (CN)	R10
Organization (O)	Let's Encrypt
Organizational Unit (OU)	<Not Part Of Certificate>

Validity Period

Issued On	Thursday, February 6, 2025 at 11:10:25 PM
Expires On	Wednesday, May 7, 2025 at 11:10:24 PM

SHA-256 Fingerprints

Certificate	adb5972a225f3533466176b24019af2d793f014a06362b2f249f6913c926a759
Public Key	0d073b9b73b4b0ee16e3f580e521c7b6c7aa20a48a764a758e1b31dff5043d90

Public-key Infrastructure

- NIST SP 800-32 (Introduction to Public Key Technology and the Federal PKI Infrastructure) defines a public-key infrastructure (PKI)

1. Any participant can read a certificate to determine the name and public key of the certificate’s owner.

2. Any participant can verify that the certificate originated from the certificate authority and is not counterfeit.

3. Only the certificate authority can create and update certificates.

4. Any participant can verify the currency of the certificate.

PKI Scenario

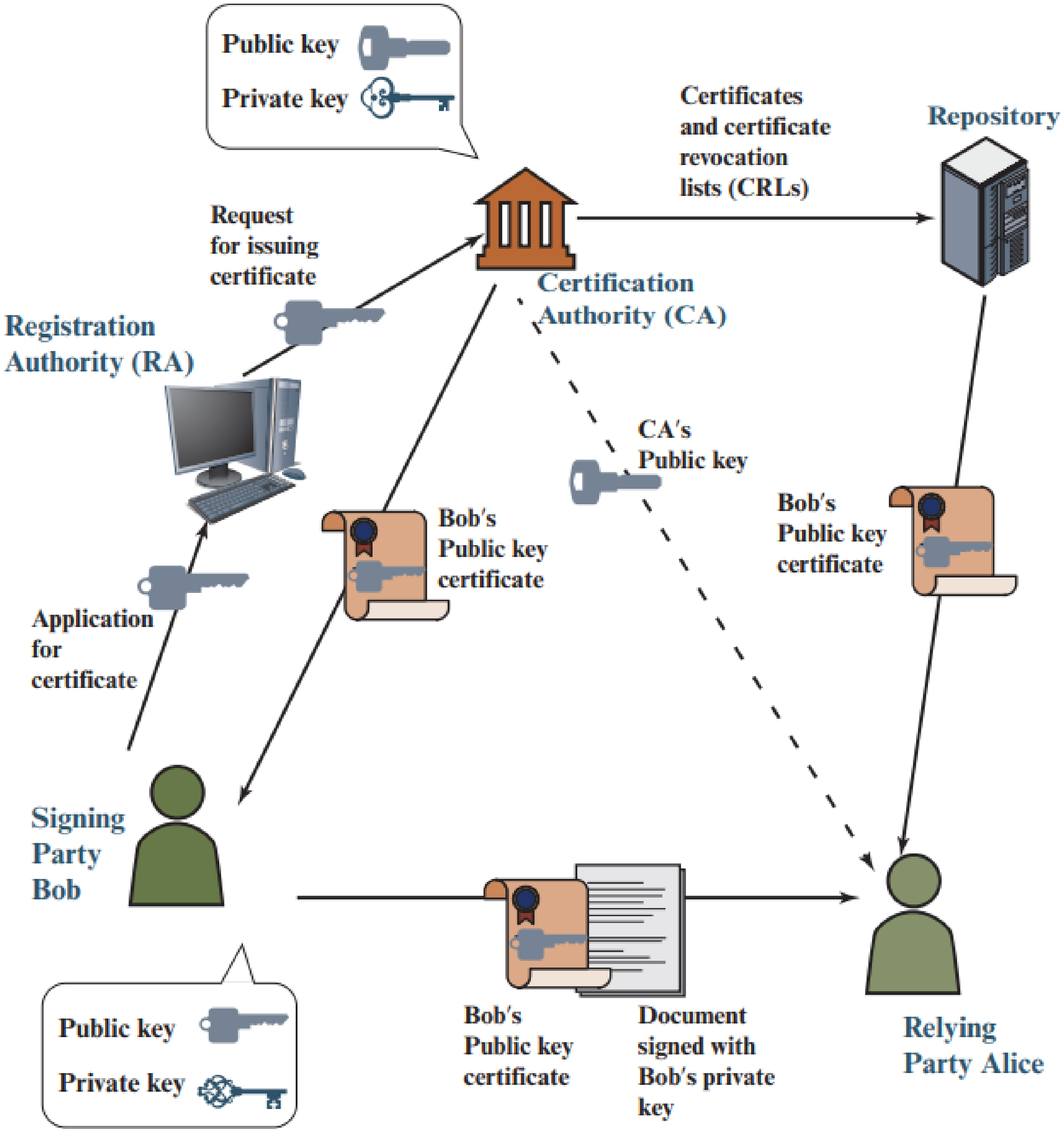
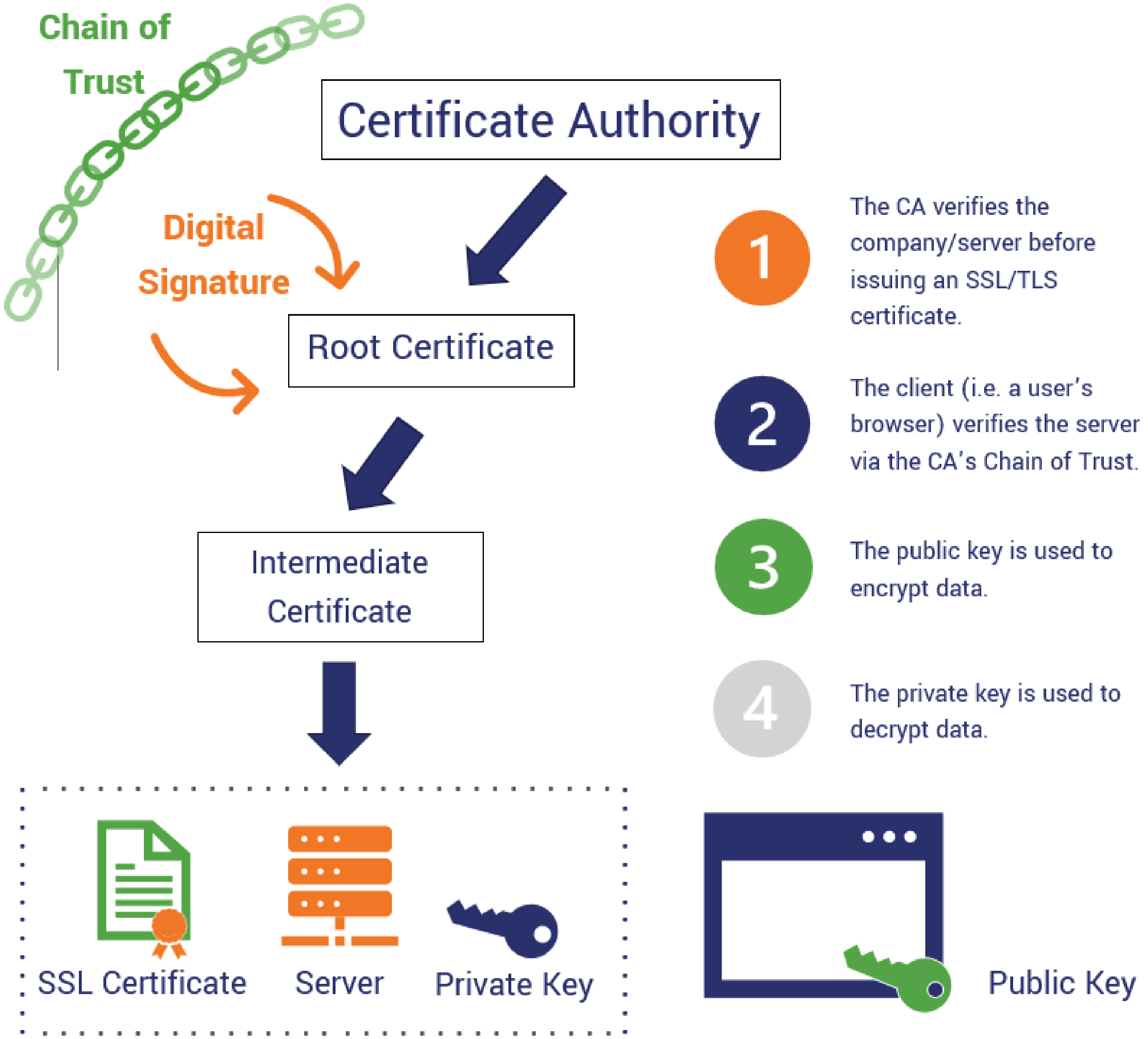

Public key
Private key
Certificates and certificate revocation lists (CRLs)
Repository
Request for issuing certificate
Certification Authority (CA)
Registration Authority (RA)
CA's Public key
Bob's Public key certificate
Bob's Public key certificate
Application for certificate
Signing Party Bob
Public key
Private key
Bob's Public key certificate
Document signed with Bob's private key
Relying Party Alice

Figure 15.13 PKI Scenario

Key Management in Action

SSL



Chain of Trust

Certificate Authority

Digital Signature

Root Certificate

Intermediate Certificate

SSL Certificate

Server

Private Key

Public Key

1 The CA verifies the company/server before issuing an SSL/TLS certificate.

2 The client (i.e. a user's browser) verifies the server via the CA's Chain of Trust.

3 The public key is used to encrypt data.

4 The private key is used to decrypt data.

Get SSL Certificate for a website

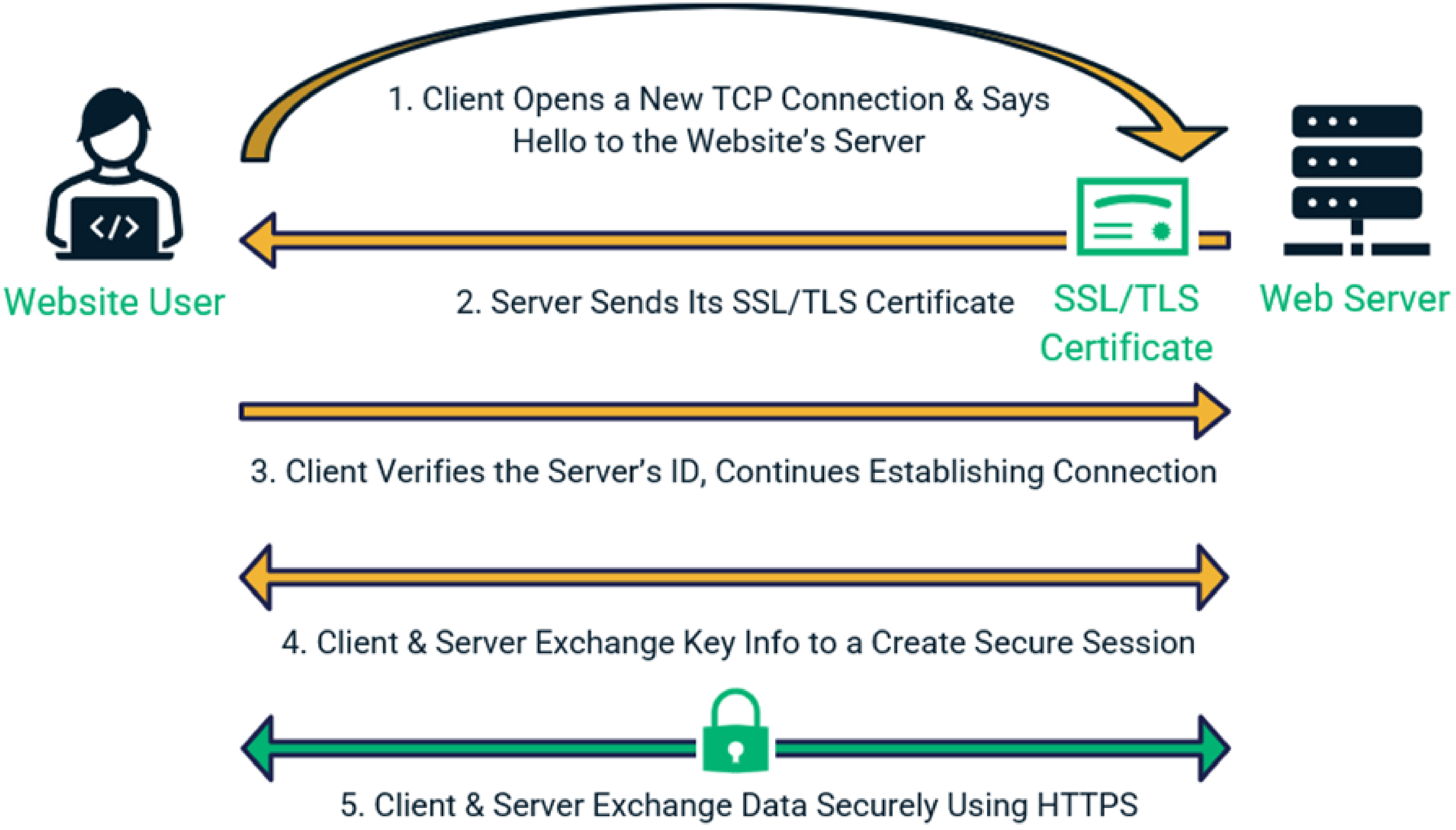
1. Setup a WebServer. e.g: Apache, IIS, Nginx, …
2. Generate self sign cert and test server config
3. Create CSR
4. Give CSR to RA
   - Free RA: Let’s Encrypt, sslforfree, zerossl, …
   - Paid RA: Comodo, Namecheap, Certum, …
5. Get sign cert files from RA
6. Install cert on server
7. Renew after expiration
   - Free certs about 180 days
   - Paid certs 365 days

Get SSL Certificate for website - Tips

- Let’s Encrypt issues certificates through an ACME protocol.
- Free and paid have same functionality.
- Certificate security depends on web server config not cert!
- Free management is harder because of short lifetime
- In no-Internet environments we must use paid certificates.

Let’s Encrypt

How PKI Enables Secure Website Connections



1. Client Opens a New TCP Connection & Says Hello to the Website's Server

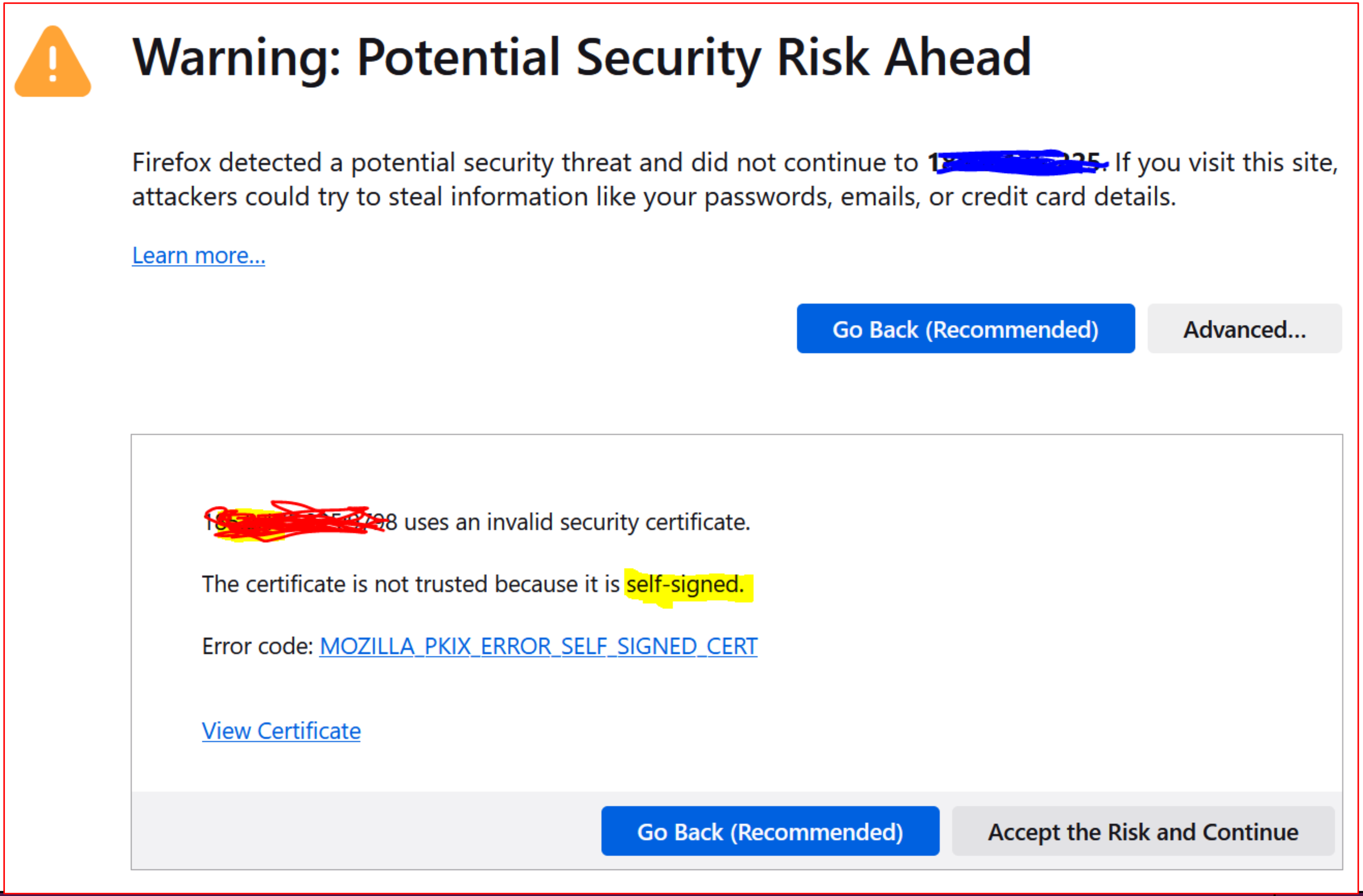
2. Server Sends Its SSL/TLS Certificate

3. Client Verifies the Server's ID, Continues Establishing Connection

4. Client & Server Exchange Key Info to a Create Secure Session

5. Client & Server Exchange Data Securely Using HTTPS

Example



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to [redacted] If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

Learn more...

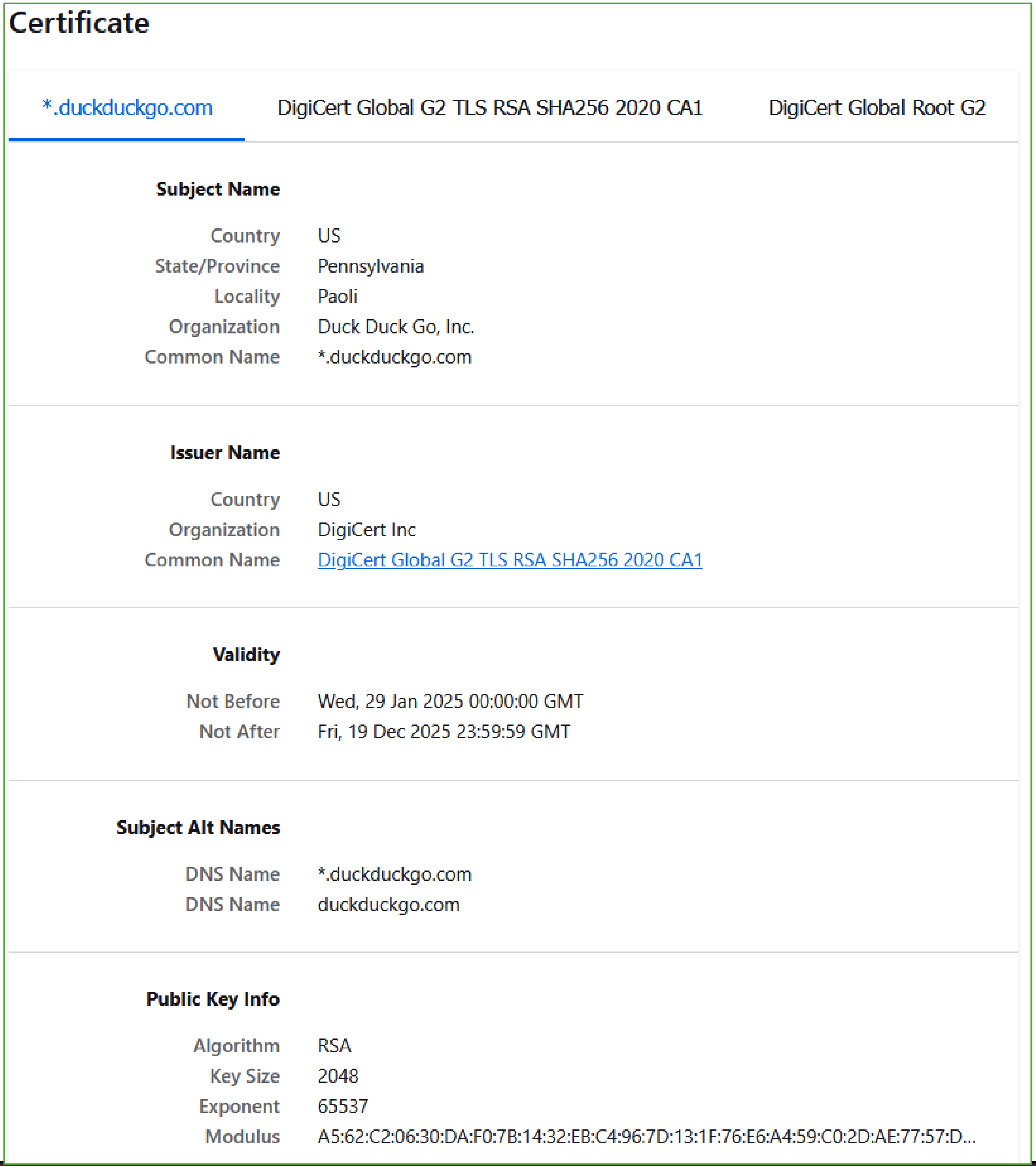
[redacted] uses an invalid security certificate.

The certificate is not trusted because it is self-signed.

Error code: MOZILLA_PKIX_ERROR_SELF_SIGNED_CERT

View Certificate

Example



Certificate

*.duckduckgo.com | DigiCert Global G2 TLS RSA SHA256 2020 CA1 | DigiCert Global Root G2

Subject Name

Country	US
State/Province	Pennsylvania
Locality	Paoli
Organization	Duck Duck Go, Inc.
Common Name	*.duckduckgo.com

Issuer Name

Country	US
Organization	DigiCert Inc
Common Name	DigiCert Global G2 TLS RSA SHA256 2020 CA1

Validity

Not Before	Wed, 29 Jan 2025 00:00:00 GMT
Not After	Fri, 19 Dec 2025 23:59:59 GMT

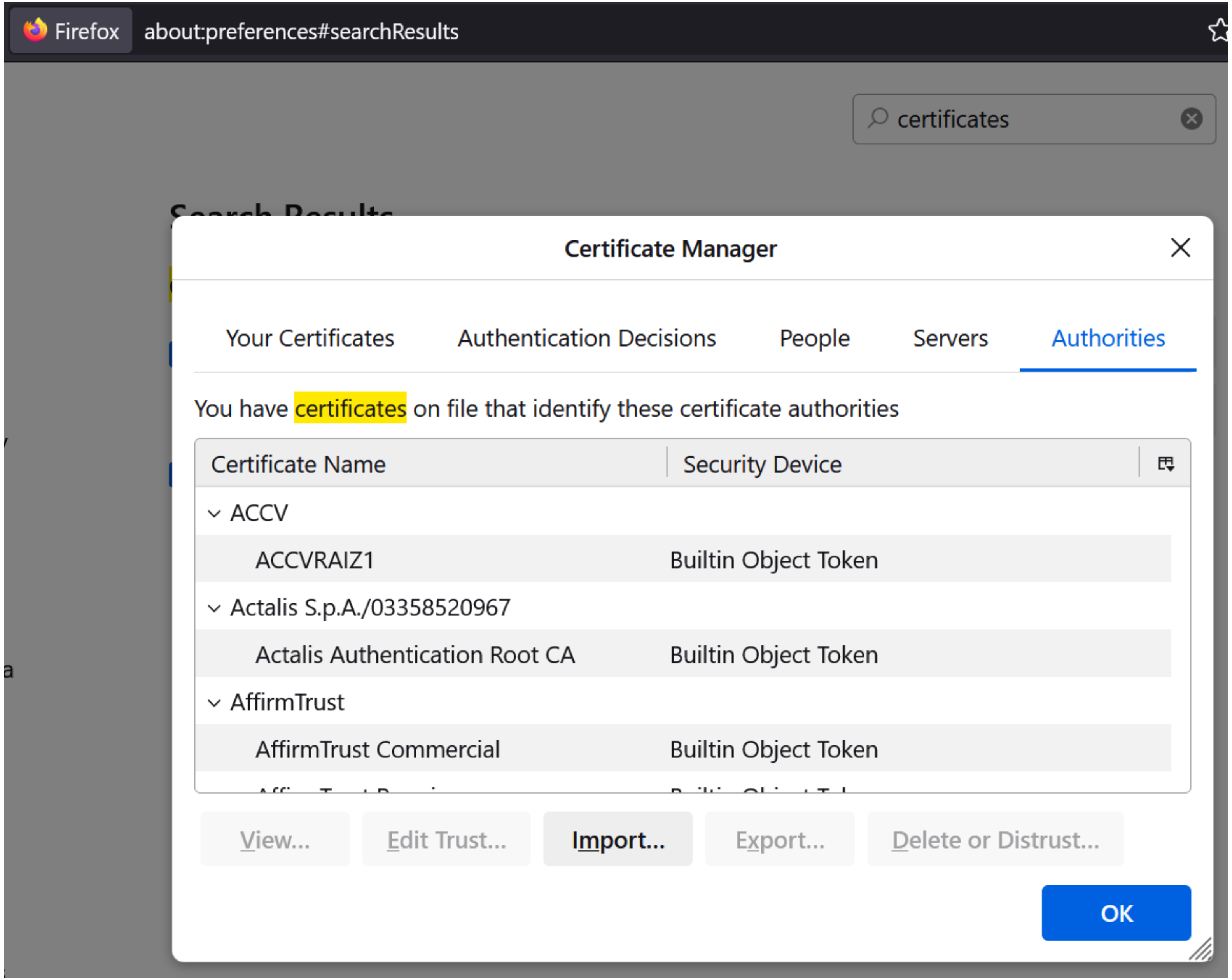
Subject Alt Names

DNS Name	*.duckduckgo.com
DNS Name	duckduckgo.com

Public Key Info

Algorithm	RSA
Key Size	2048
Exponent	65537
Modulus	A5:62:C2:06:30:DA:F0:7B:14:32:EB:C4:96:7D:13:1F:76:E6:A4:59:C0:2D:AE:77:57:D...

Example

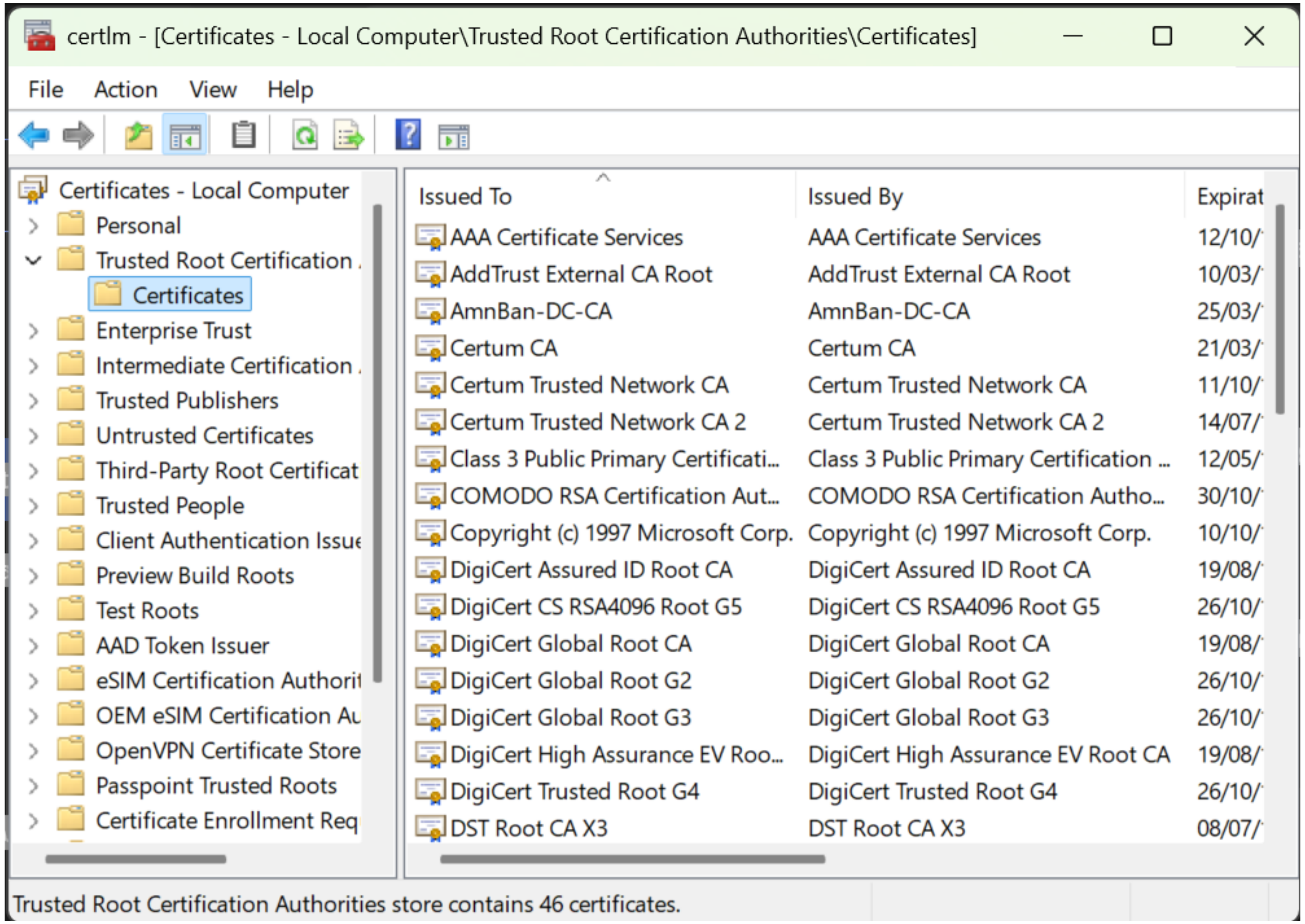


You have certificates on file that identify these certificate authorities

Certificate Name	Security Device
⌄ ACCV	
ACCVRAIZ1	Builtin Object Token
⌄ Actalis S.p.A./03358520967	
Actalis Authentication Root CA	Builtin Object Token
⌄ AffirmTrust	
AffirmTrust Commercial	Builtin Object Token

Example

- Windows certificate store

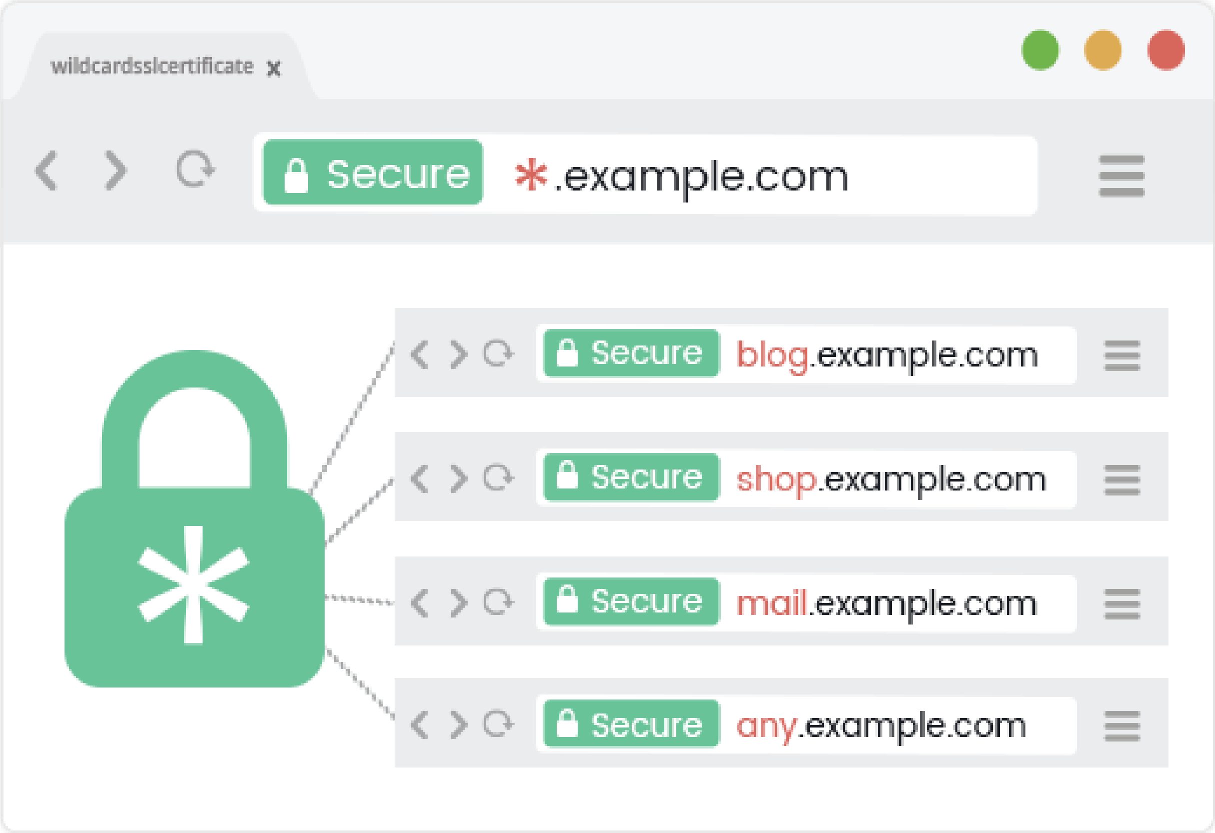

certlm - [Certificates - Local Computer\Trusted Root Certification Authorities\Certificates]
File
Action
View
Help
Certificates - Local Computer
Personal
Trusted Root Certification .
Certificates
Enterprise Trust
Intermediate Certification .
Trusted Publishers
Untrusted Certificates
Third-Party Root Certificat
Trusted People
Client Authentication Issue
Preview Build Roots
Test Roots
AAD Token Issuer
eSIM Certification Authorit
OEM eSIM Certification Au
OpenVPN Certificate Store
Passpoint Trusted Roots
Certificate Enrollment Req

Issued To	Issued By	Expirat
AAA Certificate Services	AAA Certificate Services	12/10/
AddTrust External CA Root	AddTrust External CA Root	10/03/
AmnBan-DC-CA	AmnBan-DC-CA	25/03/
Certum CA	Certum CA	21/03/
Certum Trusted Network CA	Certum Trusted Network CA	11/10/
Certum Trusted Network CA 2	Certum Trusted Network CA 2	14/07/
Class 3 Public Primary Certificati...	Class 3 Public Primary Certification ...	12/05/
COMODO RSA Certification Aut...	COMODO RSA Certification Autho...	30/10/
Copyright (c) 1997 Microsoft Corp.	Copyright (c) 1997 Microsoft Corp.	10/10/
DigiCert Assured ID Root CA	DigiCert Assured ID Root CA	19/08/
DigiCert CS RSA4096 Root G5	DigiCert CS RSA4096 Root G5	26/10/
DigiCert Global Root CA	DigiCert Global Root CA	19/08/
DigiCert Global Root G2	DigiCert Global Root G2	26/10/
DigiCert Global Root G3	DigiCert Global Root G3	26/10/
DigiCert High Assurance EV Roo...	DigiCert High Assurance EV Root CA	19/08/
DigiCert Trusted Root G4	DigiCert Trusted Root G4	26/10/
DST Root CA X3	DST Root CA X3	08/07/

Trusted Root Certification Authorities store contains 46 certificates.

Wildcard vs Single cert

- Single is for 1 domain
    - mydomain.com
- Wildcard is for all subdomains
    - news.kish.ac.ir
    - lt.kish.ac.ir
    - mail.kish.ac.ir



Other usage

- We can use valid certificate for
  - Email
  - Database connection
  - Remote desktop
  - And my other
  - User Authentication