Security & Privacy

TD2 (Password Management)

Esilv 2025

arnaud.nauwynck@gmail.com

Outline (= Same as Course)

Http Authorization

Transport: Need For confidentiality

Backend-end: storing password in Database ?

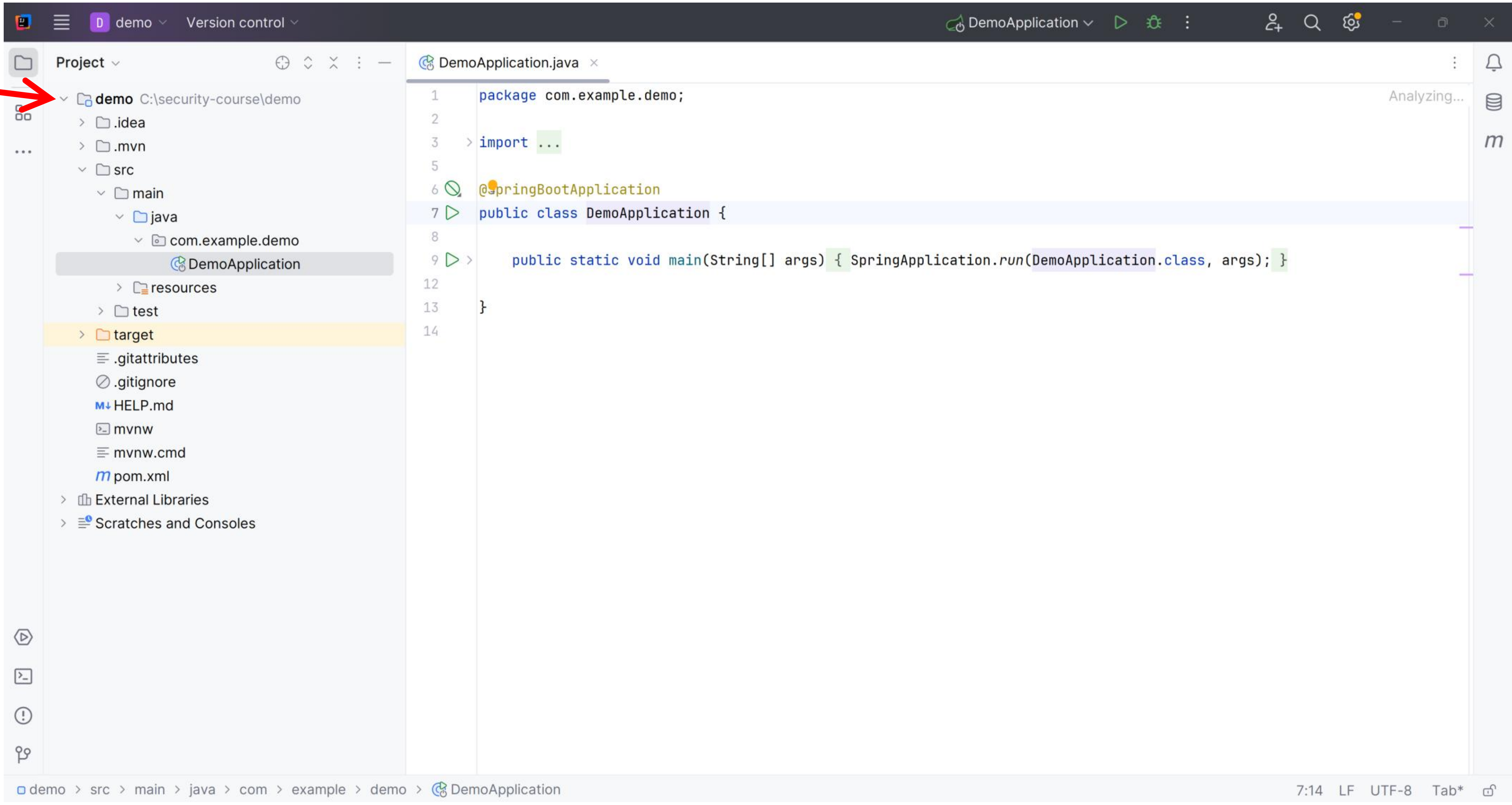
Cryptographic "Hash" functions, Salt

Who is "John The Ripper" ?

2FA & MFA (Multi Factor Authentication)

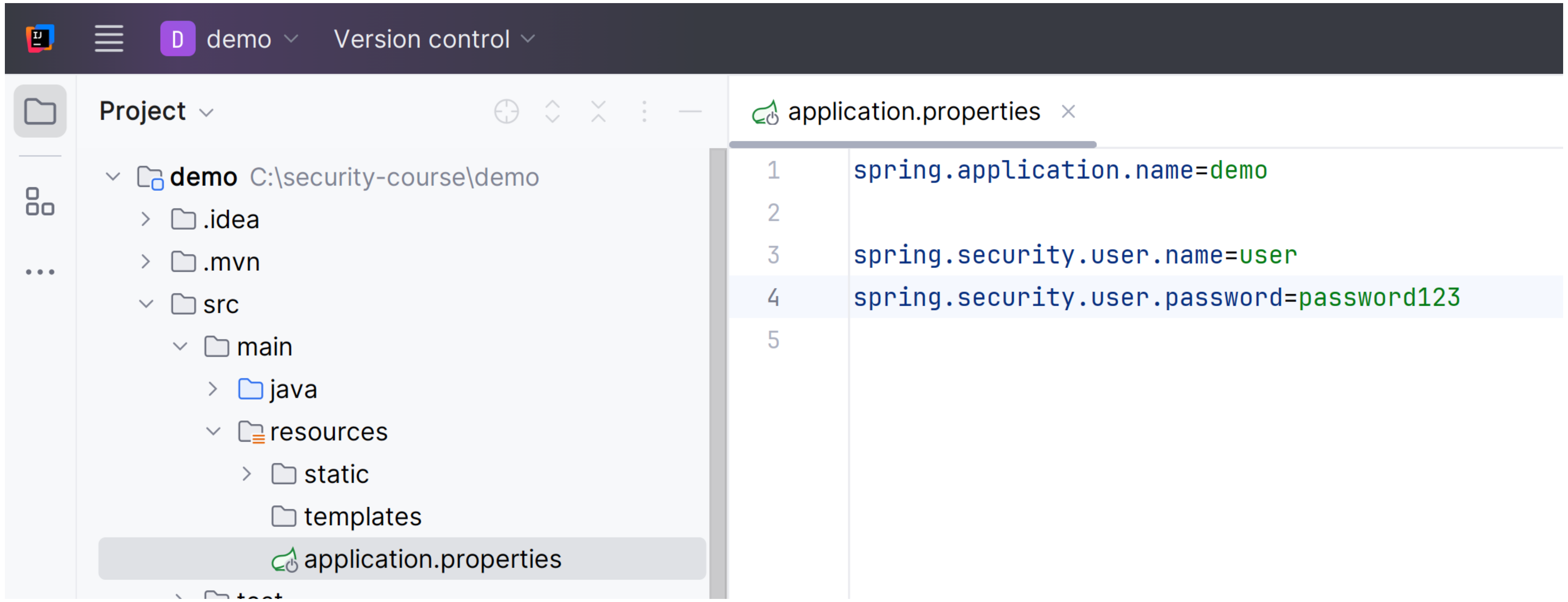
Outline

Pre-Requisite Step1: relaunch your IntelliJ IDE reopen project c:\security-course\demo



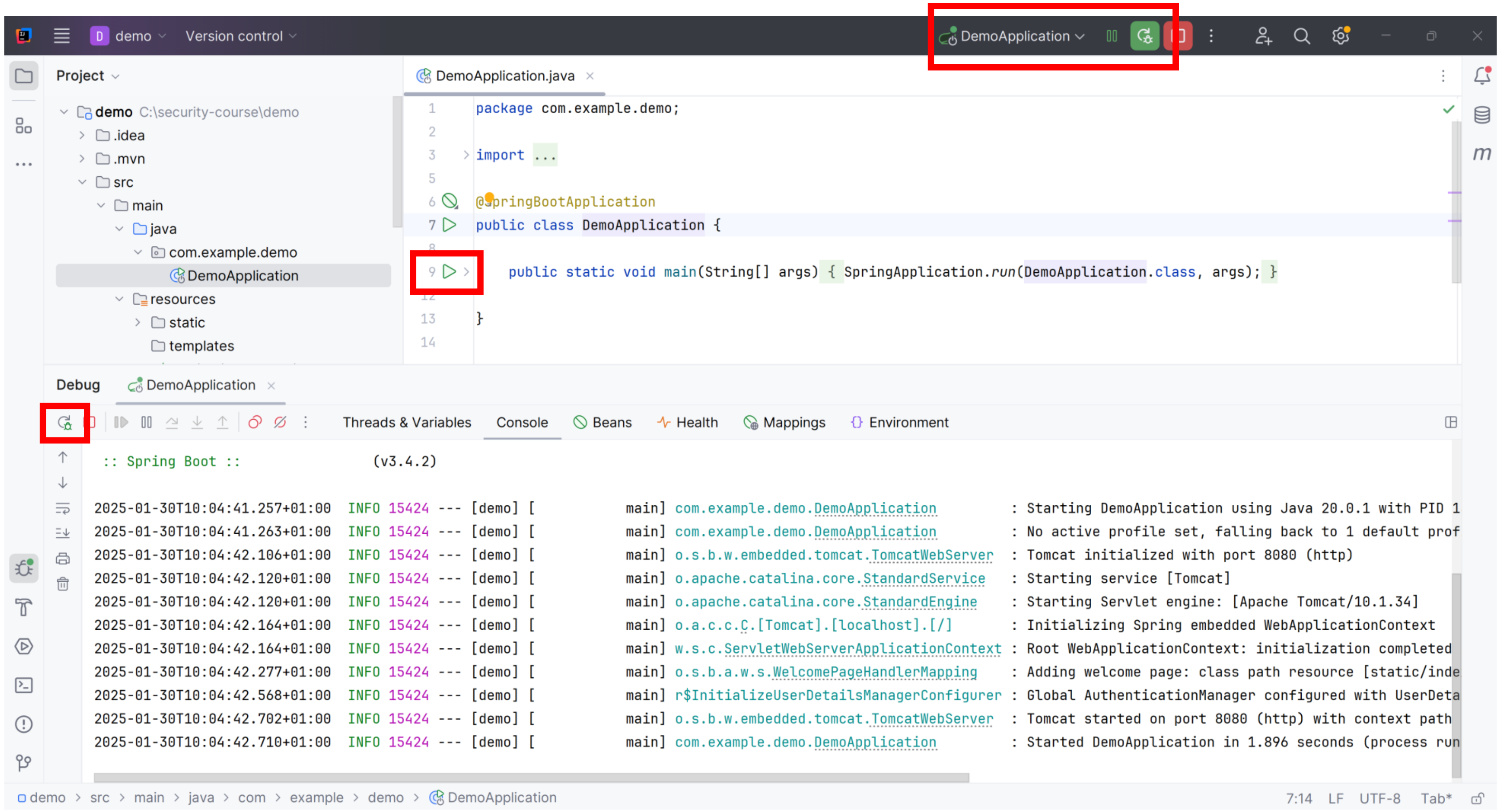
```
package com.example.demo;

import ...

@SpringBootApplication
public class DemoApplication {

    public static void main(String[] args) { SpringApplication.run(DemoApplication.class, args); }

}
```

Pre-Requisite Step 2: check file src/main/resources/application.properties for "user" / "password123"



```
spring.application.name=demo

spring.security.user.name=user
spring.security.user.password=password123
```

Pre-Requisite Step 3: relaunch webapp server



```java
package com.example.demo;

import ...

@SpringBootApplication
public class DemoApplication {

    public static void main(String[] args) { SpringApplication.run(DemoApplication.class, args); }

}
```

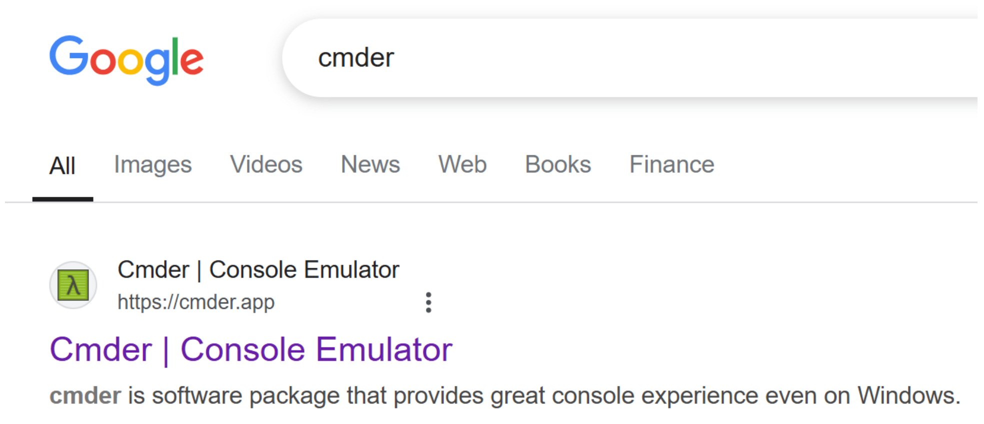
```
 :: Spring Boot ::                (v3.4.2)

2025-01-30T10:04:41.257+01:00  INFO 15424 --- [demo] [           main] com.example.demo.DemoApplication         : Starting DemoApplication using Java 20.0.1 with PID 1
2025-01-30T10:04:41.263+01:00  INFO 15424 --- [demo] [           main] com.example.demo.DemoApplication         : No active profile set, falling back to 1 default prof
2025-01-30T10:04:42.106+01:00  INFO 15424 --- [demo] [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (http)
2025-01-30T10:04:42.120+01:00  INFO 15424 --- [demo] [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-01-30T10:04:42.120+01:00  INFO 15424 --- [demo] [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.34]
2025-01-30T10:04:42.164+01:00  INFO 15424 --- [demo] [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2025-01-30T10:04:42.164+01:00  INFO 15424 --- [demo] [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed
2025-01-30T10:04:42.277+01:00  INFO 15424 --- [demo] [           main] o.s.b.a.w.s.WelcomePageHandlerMapping    : Adding welcome page: class path resource [static/inde
2025-01-30T10:04:42.568+01:00  INFO 15424 --- [demo] [           main] r$InitializeUserDetailsManagerConfigurer : Global AuthenticationManager configured with UserDeta
2025-01-30T10:04:42.702+01:00  INFO 15424 --- [demo] [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8080 (http) with context path
2025-01-30T10:04:42.710+01:00  INFO 15424 --- [demo] [           main] com.example.demo.DemoApplication         : Started DemoApplication in 1.896 seconds (process run
```

Pre-Requisite : Install (unix) shell tools "curl" and "base64"

on linux => built-in tools on system

on mac => same (or use "brew install" ?)

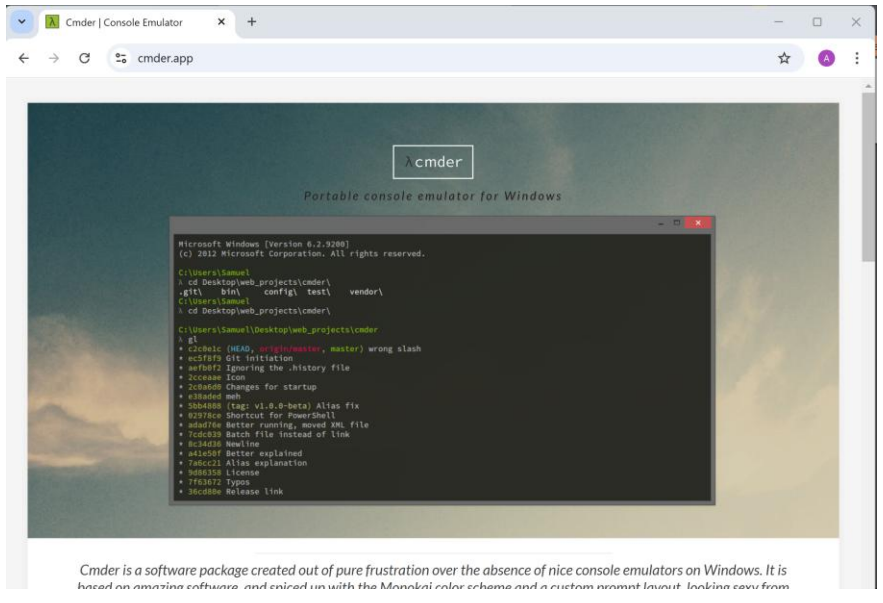
on windows => recommended solution: install "cmder" **https://cmder.app/**



Cmder | Console Emulator
https://cmder.app

Cmder | Console Emulator

cmder is software package that provides great console experience even on Windows.

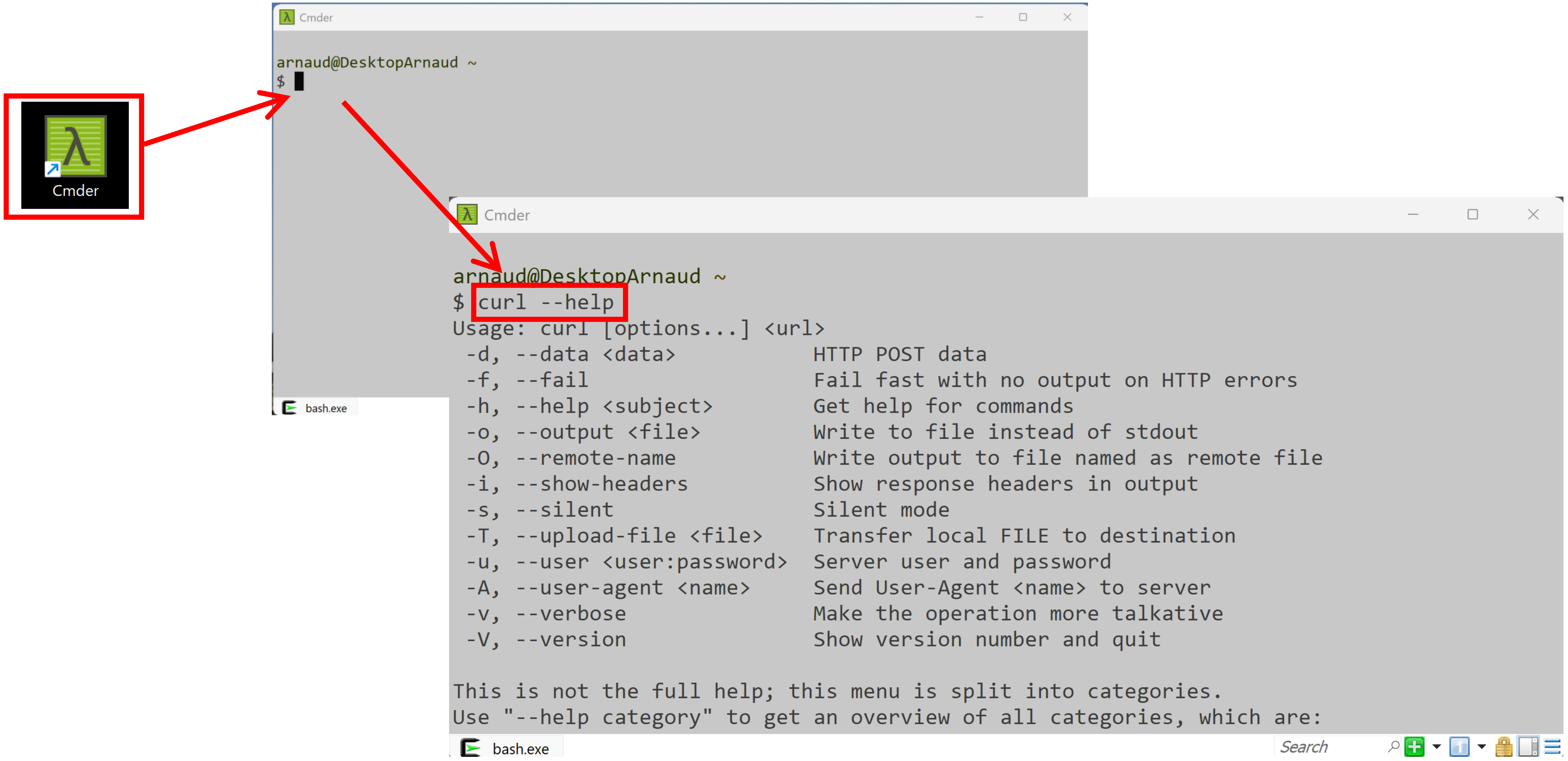


Portable console emulator for Windows

```
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.

C:\Users\Samuel
λ cd Desktop\web_projects\cmder\
.git\    bin\     config\  test\    vendor\
C:\Users\Samuel
λ cd Desktop\web_projects\cmder\

C:\Users\Samuel\Desktop\web_projects\cmder
λ gl
* c2c0e1c (HEAD, origin/master, master) wrong slash
* ec5f8f9 Git initiation
* aefb0f2 Ignoring the .history file
* 2cceaae Icon
* 2c0a6d8 Changes for startup
* e38aded meh
* 5bb4888 (tag: v1.0.0-beta) Alias fix
* 02978ce Shortcut for PowerShell
* adad76e Better running, moved XML file
* 7cdc039 Batch file instead of link
* 8c34d36 Newline
* a41e50f Better explained
* 7a6cc21 Alias explanation
* 9d86358 License
* 7f63672 Typos
* 36cd80e Release link
```

Cmder is a software package created out of pure frustration over the absence of nice console emulators on Windows. It is based on amazing software, and spiced up with the Monokai color scheme and a custom prompt layout, looking sexy from

Launch cmder console, test "curl --help"



```
arnaud@DesktopArnaud ~
$
```

```
arnaud@DesktopArnaud ~
$ curl --help
Usage: curl [options...] <url>
 -d, --data <data>           HTTP POST data
 -f, --fail                  Fail fast with no output on HTTP errors
 -h, --help <subject>        Get help for commands
 -o, --output <file>         Write to file instead of stdout
 -O, --remote-name           Write output to file named as remote file
 -i, --show-headers          Show response headers in output
 -s, --silent                Silent mode
 -T, --upload-file <file>    Transfer local FILE to destination
 -u, --user <user:password>  Server user and password
 -A, --user-agent <name>     Send User-Agent <name> to server
 -v, --verbose               Make the operation more talkative
 -V, --version               Show version number and quit

This is not the full help; this menu is split into categories.
Use "--help category" to get an overview of all categories, which are:
```

curl main arguments

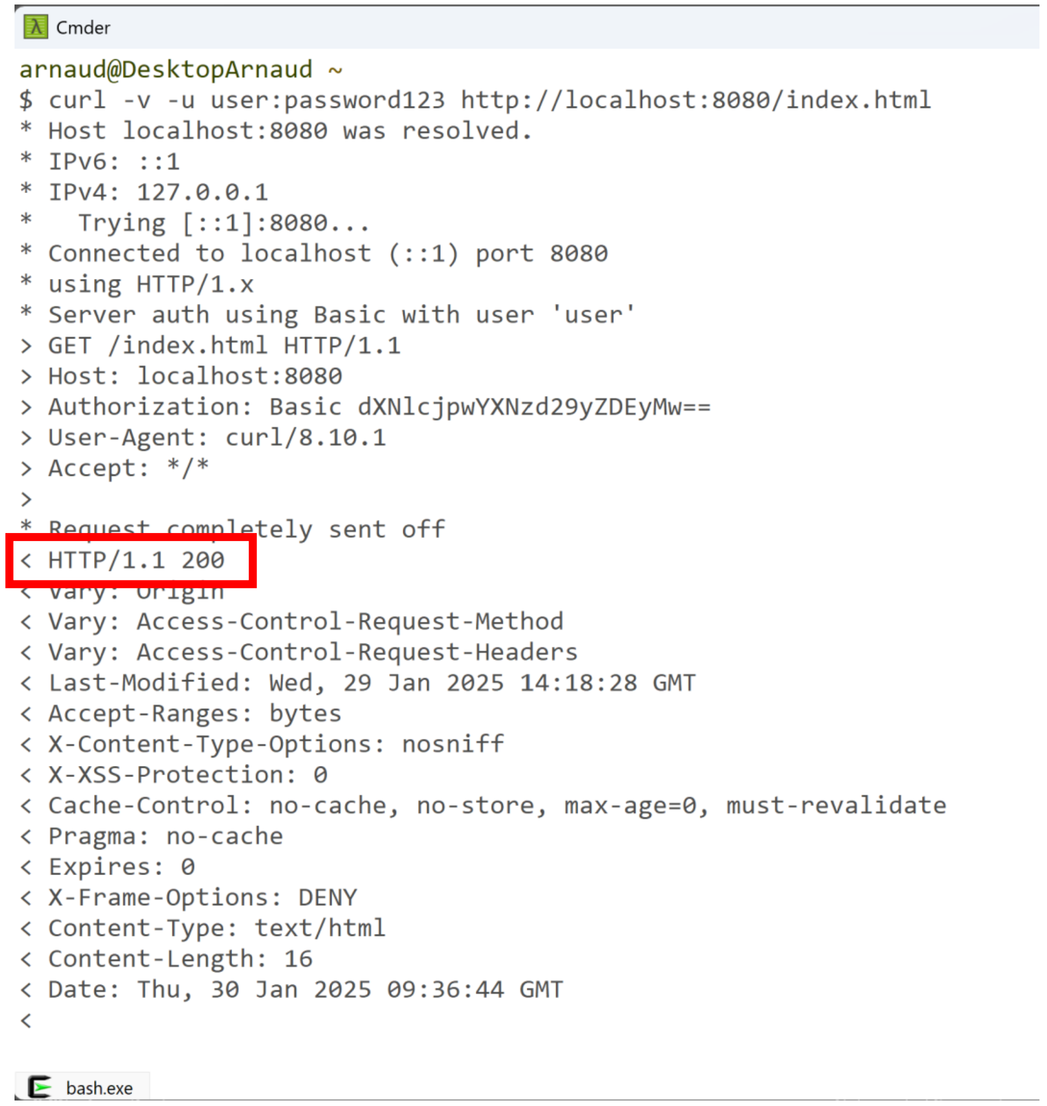
curl -v	<= for verbose
-k	<= accept all https certificate
-u "user:password"	<= basic user authentication
-x POST, PUT, DELETE	<= http verb
-H "header:value"	<= http header
url	<= http url
-d	<= http request body data

Test different curl requests.. http://localhost:8080/index.html

```
curl -v http://localhost:8080/index.html

curl -v -u user:password123 http://localhost:8080/index.html

curl -v -u user:password123 -H "accept: text/html"  http://localhost:8080/index.html

curl -v -H "Cookie: JSESSIONID=..." \
        -H "accept: text/html"  http://localhost:8080/index.html
```

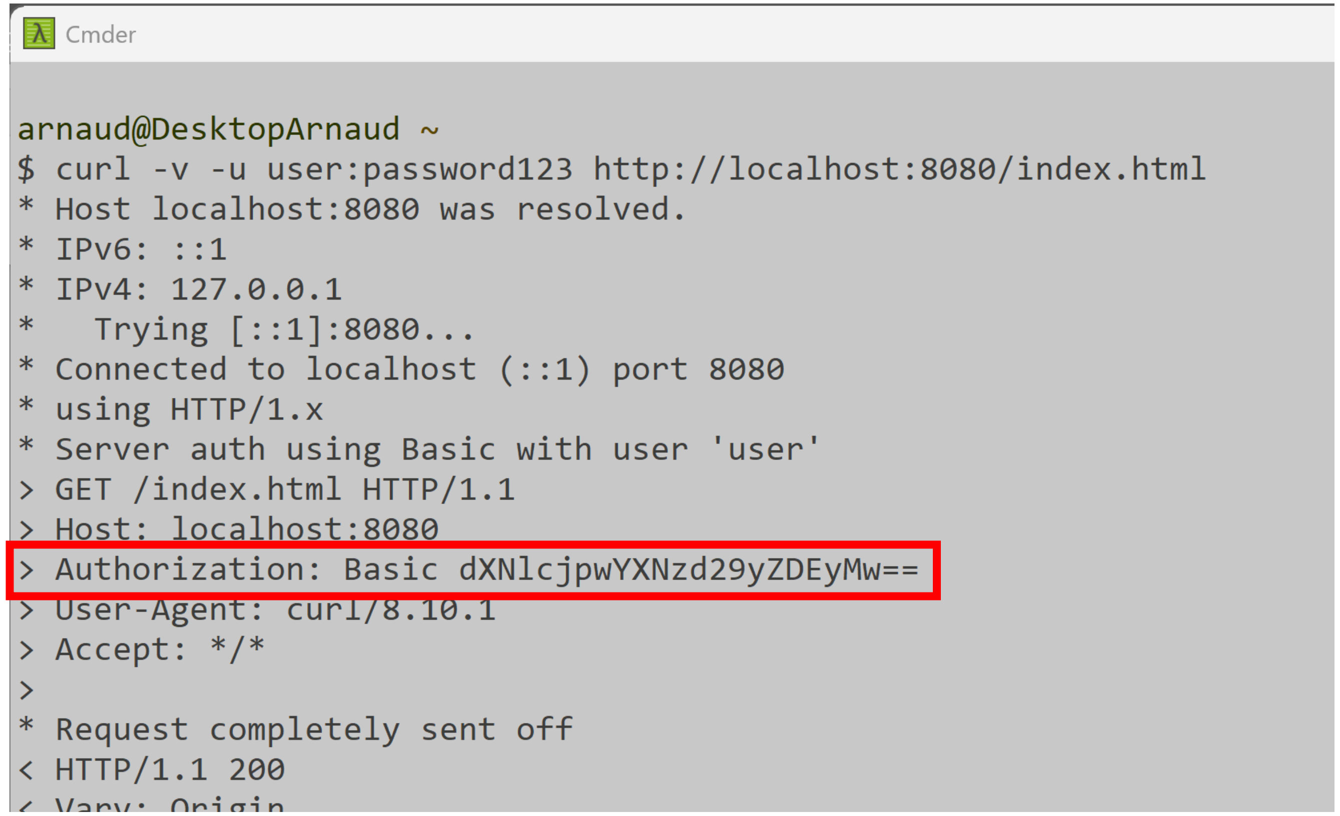
Explain different Http response status 200, 301, 400, 403, 404, ...

example with http **200**



```
arnaud@DesktopArnaud ~
$ curl -v -u user:password123 http://localhost:8080/index.html
* Host localhost:8080 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
*   Trying [::1]:8080...
* Connected to localhost (::1) port 8080
* using HTTP/1.x
* Server auth using Basic with user 'user'
> GET /index.html HTTP/1.1
> Host: localhost:8080
> Authorization: Basic dXNlcjpwYXNzd29yZDEyMw==
> User-Agent: curl/8.10.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200
< Vary: Origin
< Vary: Access-Control-Request-Method
< Vary: Access-Control-Request-Headers
< Last-Modified: Wed, 29 Jan 2025 14:18:28 GMT
< Accept-Ranges: bytes
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 0
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
< Content-Type: text/html
< Content-Length: 16
< Date: Thu, 30 Jan 2025 09:36:44 GMT
<
```

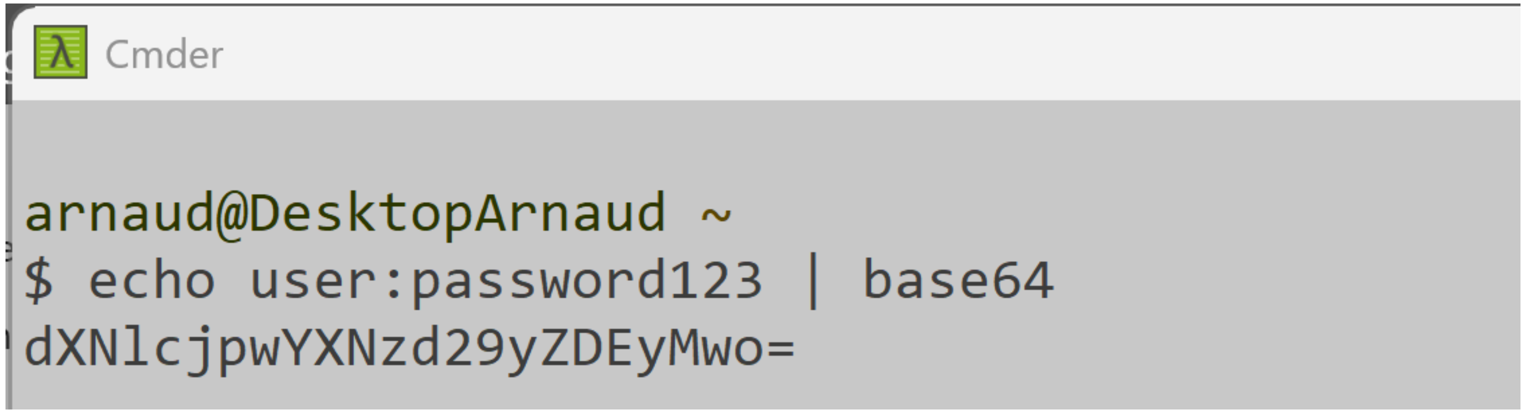
curl -v -u "user:password123"
=> see Http Header "Authorization: Basic ..."



```
arnaud@DesktopArnaud ~
$ curl -v -u user:password123 http://localhost:8080/index.html
* Host localhost:8080 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
*   Trying [::1]:8080...
* Connected to localhost (::1) port 8080
* using HTTP/1.x
* Server auth using Basic with user 'user'
> GET /index.html HTTP/1.1
> Host: localhost:8080
> Authorization: Basic dXNlcjpwYXNzd29yZDEyMw==
> User-Agent: curl/8.10.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200
< Vary: Origin
```

Encode / Decode base64 password, using "base64 -d"

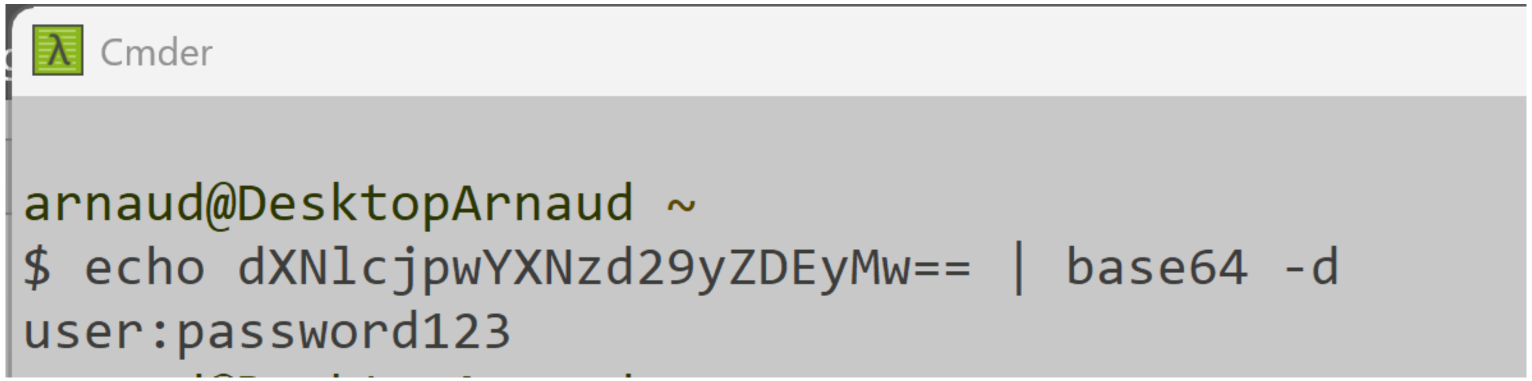
echo user:password123 | base64



Cmder

```
arnaud@DesktopArnaud ~
$ echo user:password123 | base64
dXNlcjpwYXNzd29yZDEyMwo=
```

echo dXNlcjpwYXNzd29yZDEyMw== | base64 -d



Cmder

```
arnaud@DesktopArnaud ~
$ echo dXNlcjpwYXNzd29yZDEyMw== | base64 -d
user:password123
```

Test curl Basic Authorization Header (base64)

```
curl -v -H "Authorization: Basic dXNlcjpwYXNzd29yZDEyMw==" \
        -H "accept: text/html"  http://localhost:8080/index.html
```

Remember ...
When taking Http Request screenshots, or sharing screens to have IT support
... Do not reveal your "base64 encoded" password !!!

Outline

Http Authorization

Transport: Need For confidentiality

Backend-end: storing password in Database ?

Cryptographic "Hash" functions, Salt

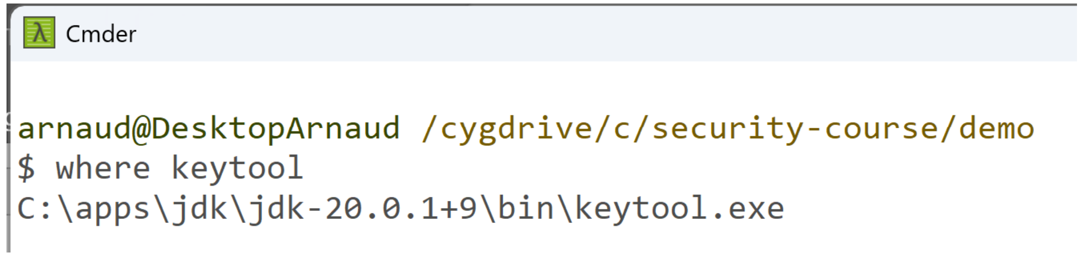
Who is "John The Ripper" ?

2FA & MFA (Multi Factor Authentication)

Generate your Https self-signed certificate

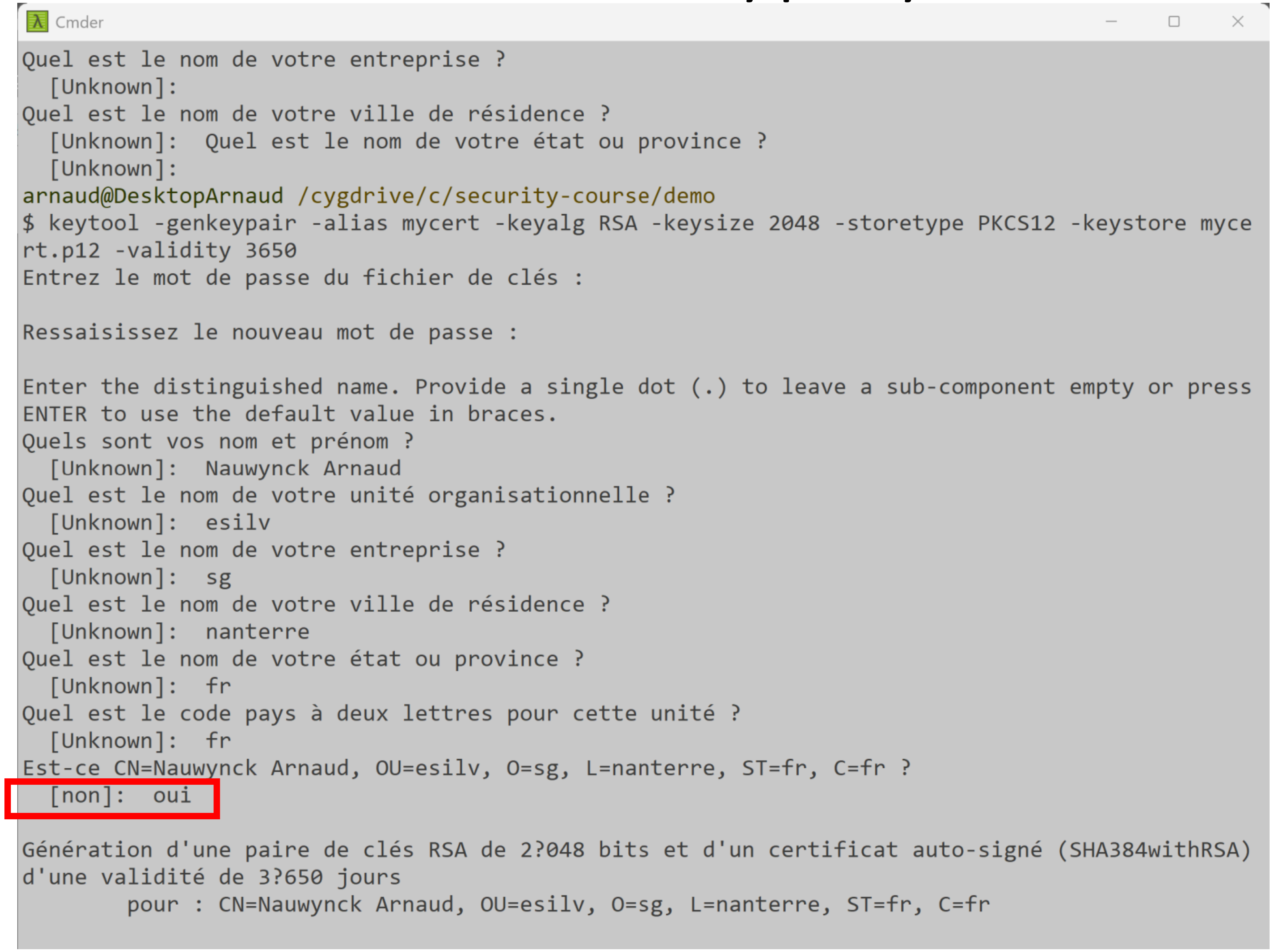
keytool -genkeypair -alias mycert -keyalg RSA -keysize 2048 -storetype PKCS12 -keystore mycert.p12 -validity 3650

Troubleshooting ?
"keytool" is a tool in your jdk/bin directory
check "where keytool" => it should be found in your "PATH" environment variable
check "echo $PATH"
... to fix, use windows "env" to edit your environement variables, or export PATH="$PATH:$JAVA_HOME/bin"



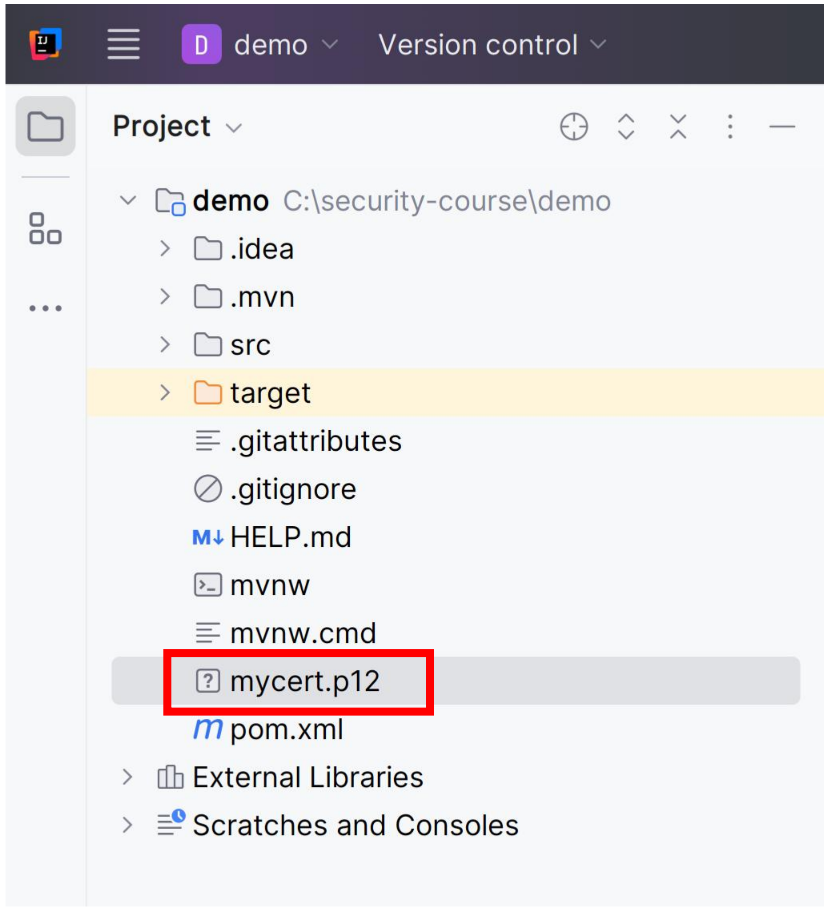
```
arnaud@DesktopArnaud /cygdrive/c/security-course/demo
$ where keytool
C:\apps\jdk\jdk-20.0.1+9\bin\keytool.exe
```

interactive answer to prompt (any is ok) lastly "Is it CN=.... ? " [no]:   type yes !!

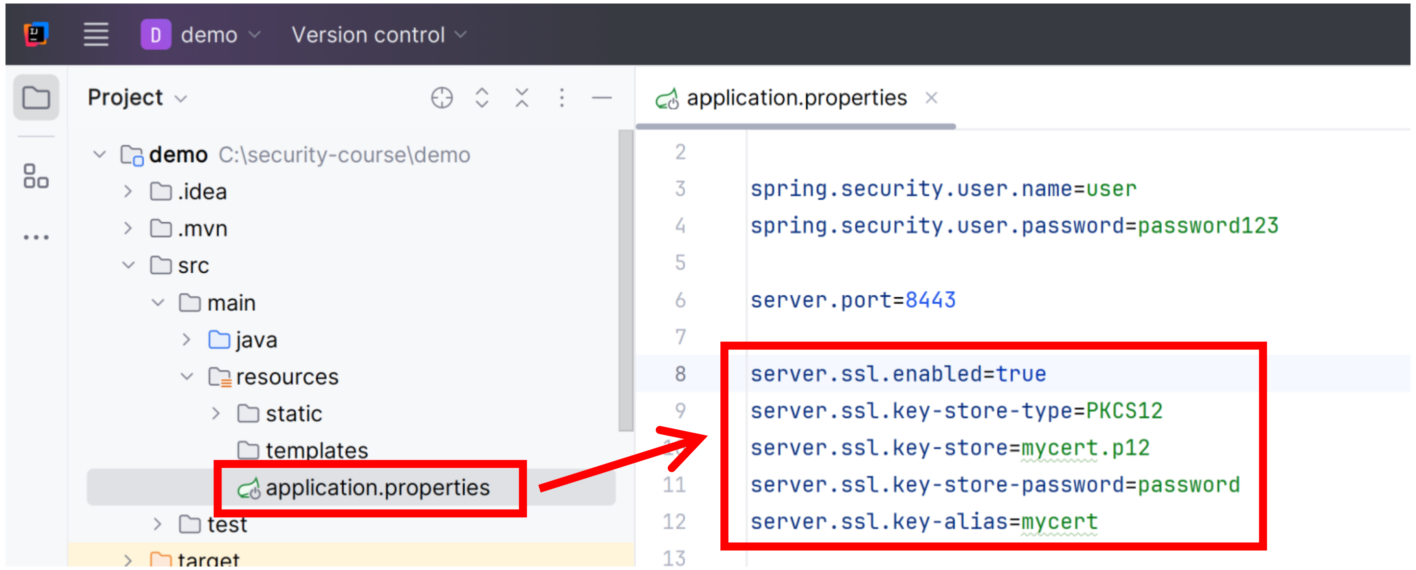


```
Quel est le nom de votre entreprise ?
  [Unknown]:
Quel est le nom de votre ville de résidence ?
  [Unknown]:  Quel est le nom de votre état ou province ?
  [Unknown]:
arnaud@DesktopArnaud /cygdrive/c/security-course/demo
$ keytool -genkeypair -alias mycert -keyalg RSA -keysize 2048 -storetype PKCS12 -keystore myce
rt.p12 -validity 3650
Entrez le mot de passe du fichier de clés :

Ressaisissez le nouveau mot de passe :

Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press
ENTER to use the default value in braces.
Quels sont vos nom et prénom ?
  [Unknown]:  Nauwynck Arnaud
Quel est le nom de votre unité organisationnelle ?
  [Unknown]:  esilv
Quel est le nom de votre entreprise ?
  [Unknown]:  sg
Quel est le nom de votre ville de résidence ?
  [Unknown]:  nanterre
Quel est le nom de votre état ou province ?
  [Unknown]:  fr
Quel est le code pays à deux lettres pour cette unité ?
  [Unknown]:  fr
Est-ce CN=Nauwynck Arnaud, OU=esilv, O=sg, L=nanterre, ST=fr, C=fr ?
  [non]:  oui

Génération d'une paire de clés RSA de 2?048 bits et d'un certificat auto-signé (SHA384withRSA)
d'une validité de 3?650 jours
        pour : CN=Nauwynck Arnaud, OU=esilv, O=sg, L=nanterre, ST=fr, C=fr
```

Check created file "mycert.p12" in project (or move it in your project)



Edit src/main/resources/application.properties to configure SSL

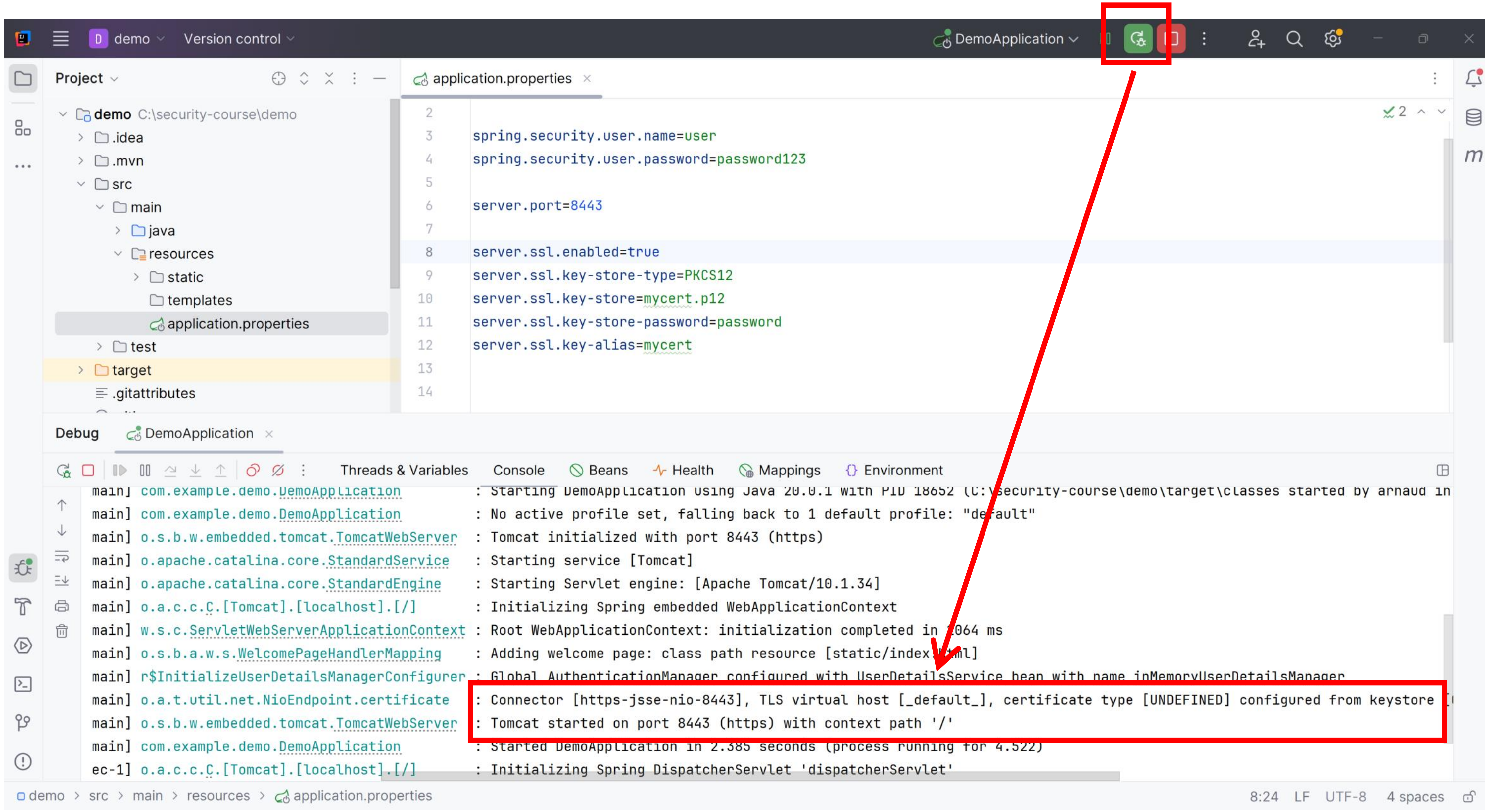


```
spring.security.user.name=user
spring.security.user.password=password123

server.port=8443

server.ssl.enabled=true
server.ssl.key-store-type=PKCS12
server.ssl.key-store=mycert.p12
server.ssl.key-store-password=password
server.ssl.key-alias=mycert
```

server.port=8443

server.ssl.enabled=true
server.ssl.key-store-type=PKCS12
server.ssl.key-store=mycert.p12
server.ssl.key-store-password=password
server.ssl.key-alias=mycert

Relaunch server, check new console log



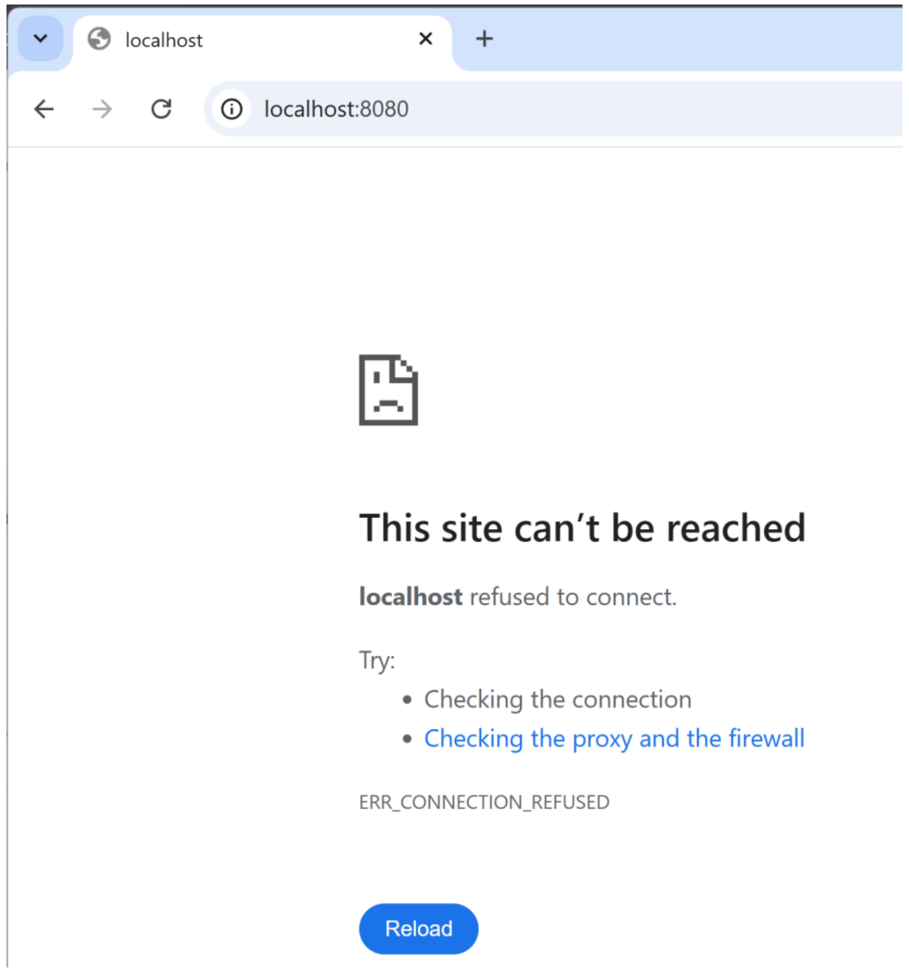
```
spring.security.user.name=user
spring.security.user.password=password123

server.port=8443

server.ssl.enabled=true
server.ssl.key-store-type=PKCS12
server.ssl.key-store=mycert.p12
server.ssl.key-store-password=password
server.ssl.key-alias=mycert
```

```
main] com.example.demo.DemoApplication            : Starting DemoApplication using Java 20.0.1 with PID 18652 (C:\security-course\demo\target\classes started by arnaud in
main] com.example.demo.DemoApplication            : No active profile set, falling back to 1 default profile: "default"
main] o.s.b.w.embedded.tomcat.TomcatWebServer     : Tomcat initialized with port 8443 (https)
main] o.apache.catalina.core.StandardService      : Starting service [Tomcat]
main] o.apache.catalina.core.StandardEngine       : Starting Servlet engine: [Apache Tomcat/10.1.34]
main] o.a.c.c.C.[Tomcat].[localhost].[/]          : Initializing Spring embedded WebApplicationContext
main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1064 ms
main] o.s.b.a.w.s.WelcomePageHandlerMapping       : Adding welcome page: class path resource [static/index.html]
main] r$InitializeUserDetailsManagerConfigurer : Global AuthenticationManager configured with UserDetailsService bean with name inMemoryUserDetailsManager
main] o.a.t.util.net.NioEndpoint.certificate      : Connector [https-jsse-nio-8443], TLS virtual host [_default_], certificate type [UNDEFINED] configured from keystore [
main] o.s.b.w.embedded.tomcat.TomcatWebServer     : Tomcat started on port 8443 (https) with context path '/'
main] com.example.demo.DemoApplication            : Started DemoApplication in 2.385 seconds (process running for 4.522)
ec-1] o.a.c.c.C.[Tomcat].[localhost].[/]          : Initializing Spring DispatcherServlet 'dispatcherServlet'
```

port changed 8080 (default for http) -> 8443 (default for https)

on previous port 8080
=> "ERROR Connection refused"

OK, Normal !!



This site can’t be reached

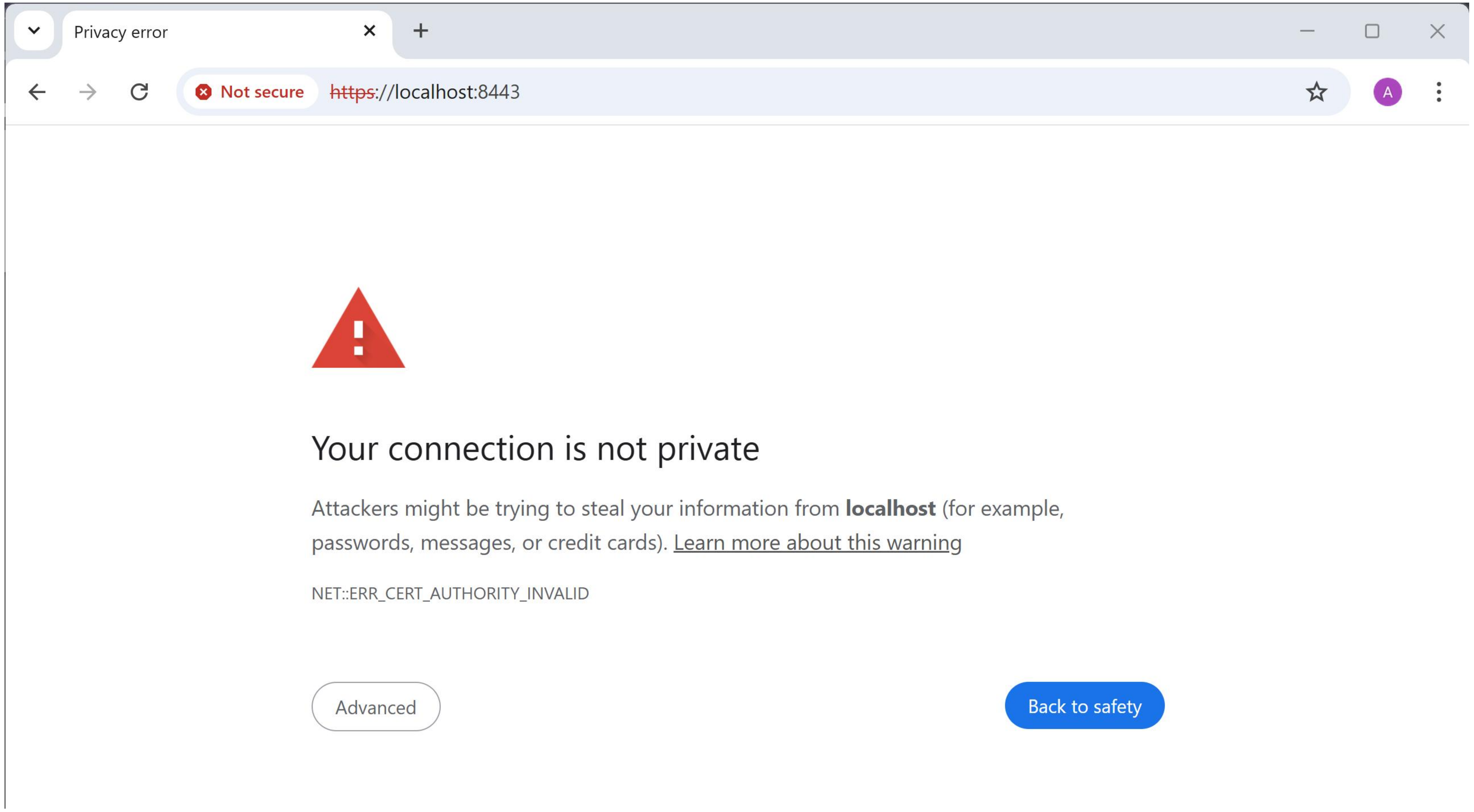
localhost refused to connect.

Try:

- Checking the connection
- Checking the proxy and the firewall

ERR_CONNECTION_REFUSED

https://localhost:8443 (self-signed certificate) CERT_AUTHORITY_INVALID ... OK



Your connection is not private

Attackers might be trying to steal your information from **localhost** (for example, passwords, messages, or credit cards). Learn more about this warning

NET::ERR_CERT_AUTHORITY_INVALID

Why ?
… because of self-signed certificate
NOT built-in recognized in your Chrome

Alternatives ?

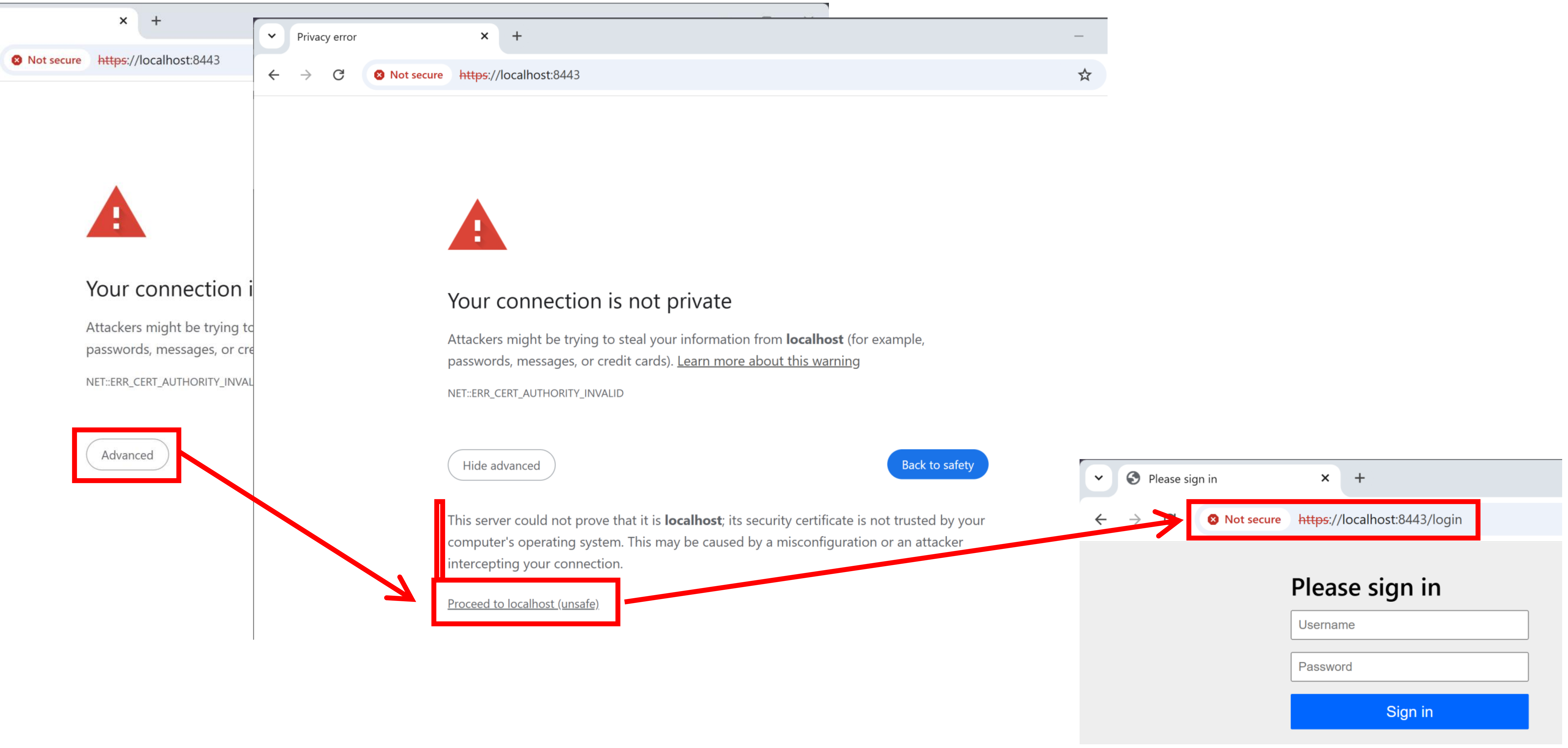
=> pay 200 EURO/year to have a valid certificate ?

=> use GO-DADDY   or other signing Authority on internet (recognized by Chrome)

=> recompile chrome to know "yourself" as valid authority

=> click on "ADVANCED" button to accept self-signed certificate

Advanced > Proceed to (unsafe)



Privacy error

Not secure https://localhost:8443

Your connection is not private

Attackers might be trying to steal your information from **localhost** (for example, passwords, messages, or credit cards). Learn more about this warning

NET::ERR_CERT_AUTHORITY_INVALID

This server could not prove that it is **localhost**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

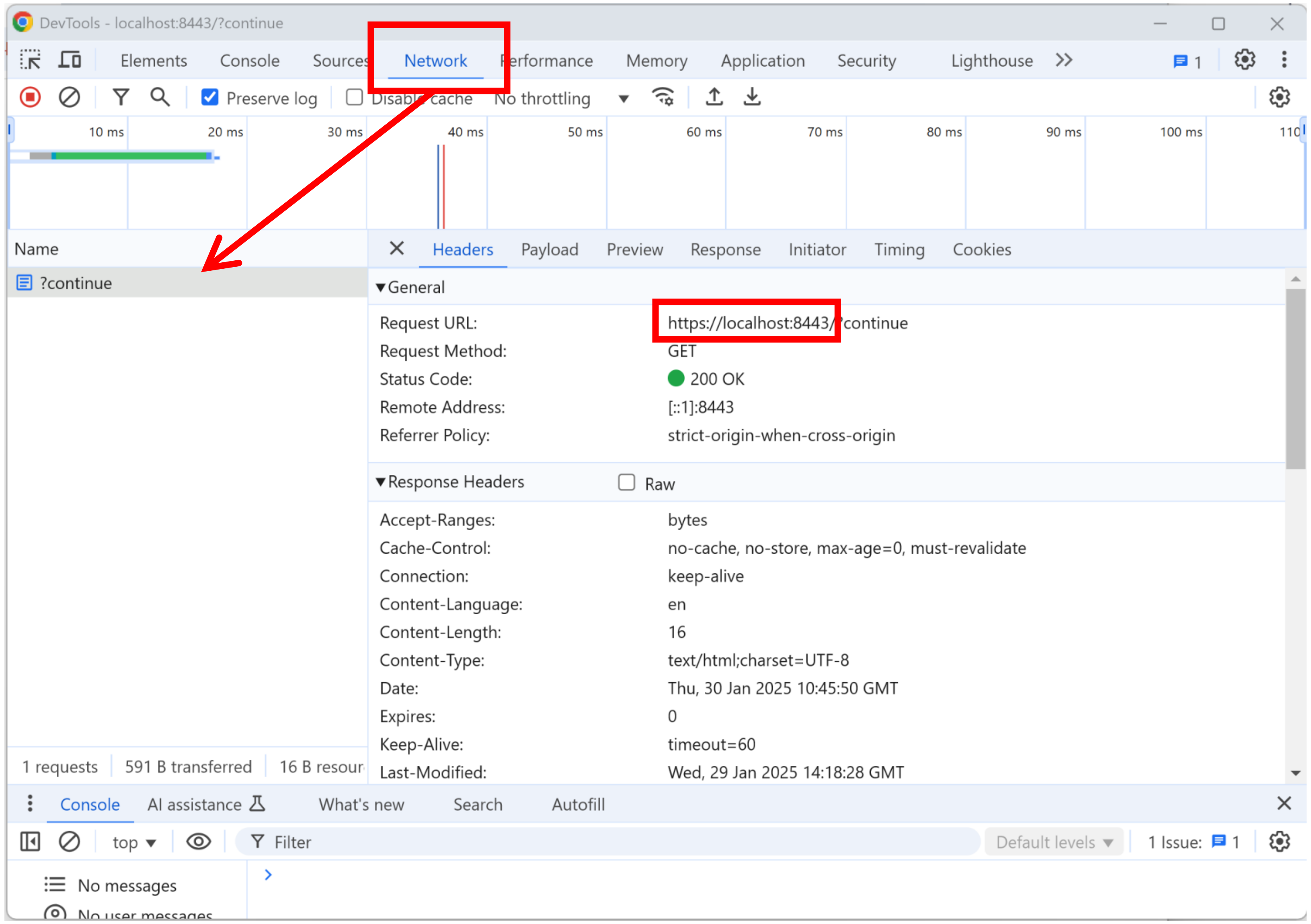
Proceed to localhost (unsafe)

Please sign in

Not secure https://localhost:8443/login

Please sign in

Chrome DevTools Network requests ... same on "https://"



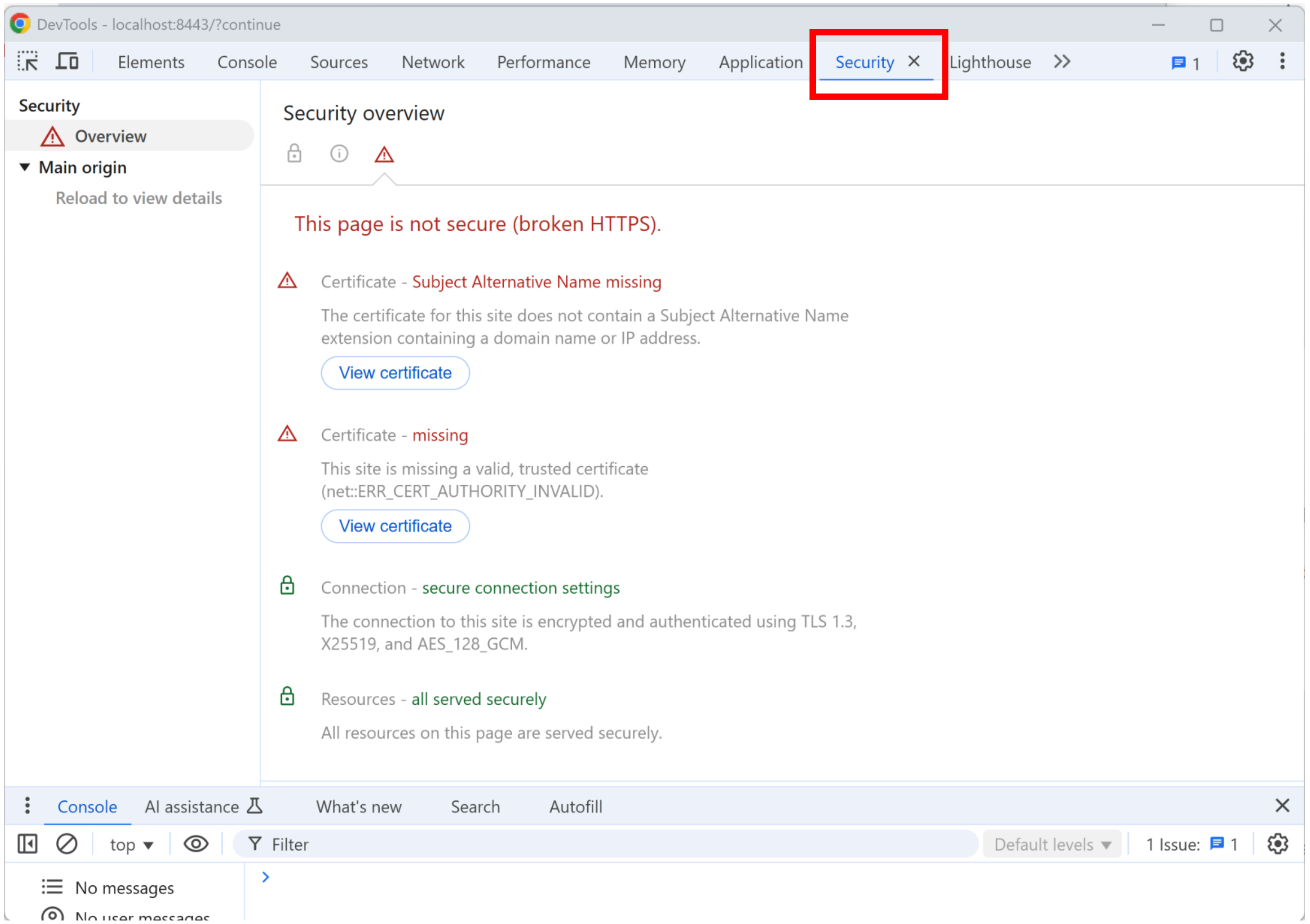
General

Request URL:	https://localhost:8443/?continue
Request Method:	GET
Status Code:	● 200 OK
Remote Address:	[::1]:8443
Referrer Policy:	strict-origin-when-cross-origin

Response Headers   Raw

Accept-Ranges:	bytes
Cache-Control:	no-cache, no-store, max-age=0, must-revalidate
Connection:	keep-alive
Content-Language:	en
Content-Length:	16
Content-Type:	text/html;charset=UTF-8
Date:	Thu, 30 Jan 2025 10:45:50 GMT
Expires:	0
Keep-Alive:	timeout=60
Last-Modified:	Wed, 29 Jan 2025 14:18:28 GMT

Chrome DevTools "Security" tab



DevTools - localhost:8443/?continue

Elements Console Sources Network Performance Memory Application **Security** Lighthouse

Security

Overview

Main origin

Reload to view details

Security overview

This page is not secure (broken HTTPS).

Certificate - Subject Alternative Name missing

The certificate for this site does not contain a Subject Alternative Name extension containing a domain name or IP address.

View certificate

Certificate - missing

This site is missing a valid, trusted certificate (net::ERR_CERT_AUTHORITY_INVALID).

View certificate

Connection - secure connection settings

The connection to this site is encrypted and authenticated using TLS 1.3, X25519, and AES_128_GCM.

Resources - all served securely

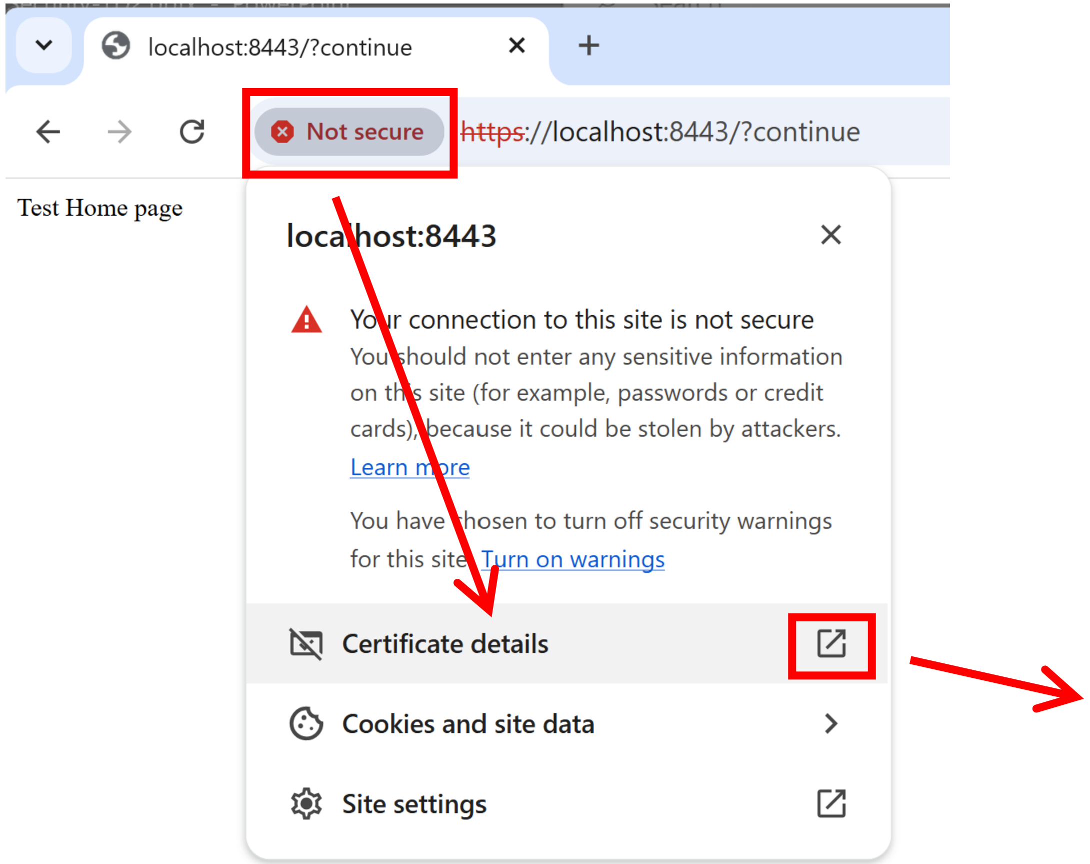
All resources on this page are served securely.

Console AI assistance What's new Search Autofill

top Filter Default levels 1 Issue: 1

No messages

See Also Chrome -> Certificate Details



localhost:8443/?continue

Not secure ~~https~~://localhost:8443/?continue

Test Home page

localhost:8443

Your connection to this site is not secure

You should not enter any sensitive information on this site (for example, passwords or credit cards), because it could be stolen by attackers. Learn more

You have chosen to turn off security warnings for this site. Turn on warnings

Suggestion ...
open view on a real web-site, "https://www.google.fr"
to see a valid signed certificate

Outline

- Http Authorization
- Transport: Need For confidentiality
- ➡ Backend-end: storing password in Database ?
- Cryptographic "Hash" functions, Salt
- Who is "John The Ripper" ?
- 2FA & MFA (Multi Factor Authentication)

Outline

Outline

Outline

Http Authorization

Transport: Need For confidentiality

➡ Backend-end: storing password in Database ?

Cryptographic "Hash" functions, Salt

Who is "John The Ripper" ?

2FA & MFA (Multi Factor Authentication)