



# Hacker-Workshop

---

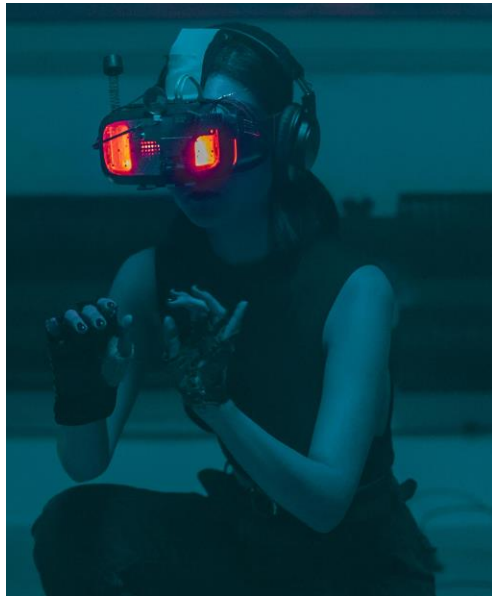
I dag skal vi lære teknikker til  
at hacke og at jage hackere.

**KEA**

**KØBENHAVNS ERHVERVSAKADEMI**

# Undervisere

- Kevin – Underviser på IT-Teknolog i Programmering
- Natasha – Underviser på IT-Teknolog i Indlejrede systemer og på  
Kompetence i DFIR – Digital Forensics og Incident Response



# Agenda

- 0900 – Velkomst og praktisk
- 0915 - Intro til forensics teknikker
- 0945 - Pause
- 1000 - Intro til offensive hacker teknikker
- 1030 - Øvelser
- 1145 - Afslutning





# Velkommen

- Kaffe/Te
- Toiletter
- Spørgsmål undervejs
- Uddannelser på KEA Competence
  - Kommer til sidst



# Forensics

- Find ud af hvad der er sket på en computer
  - Herunder slettede filer og browser historik
- Kig på maskinkoden for et program
- Forensics er en paraply term for følgende:
  - Host forensics – Kigge på en eller flere computere
  - Log analyse – Kigge på logs fra forskelligt udstyr og cloudservices for at danne overblik over en bruger eller computers handlinger
  - Reversing – Analyse af malware for at finde flere spor at gå efter i forensics arbejdet



# Autopsy – et sagsværktøj



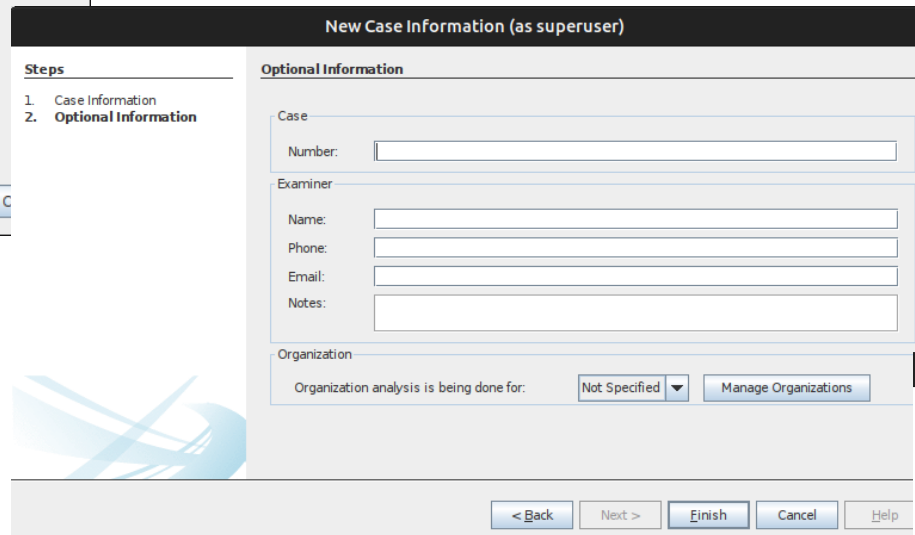
- Autopsy kan holde styr på diskimages indsamlet for analysearbejde
- Derfor skal man indtaste nogle oplysninger
- Autopsy bygger på SleuthKit
- Autopsy kan tilbyde automatisk analyse af diskimages
  - Analyse af browser
  - Finde filer i rå data (disk carving)
  - Noget analyse tager lang tid, andre kort tid
  - Godt start tip: Fravælg alt, vælg recent Activity

# Brug af Autopsy



- Start en ny sag
- Tilføje beviser / disk-images
- Analyse af filer

# Autopsy – Start en ny sag



New Case Information (as superuser)

**Steps**

1. Case Information
2. **Optional Information**

**Optional Information**

Case

Number:

Examiner

Name:

Phone:

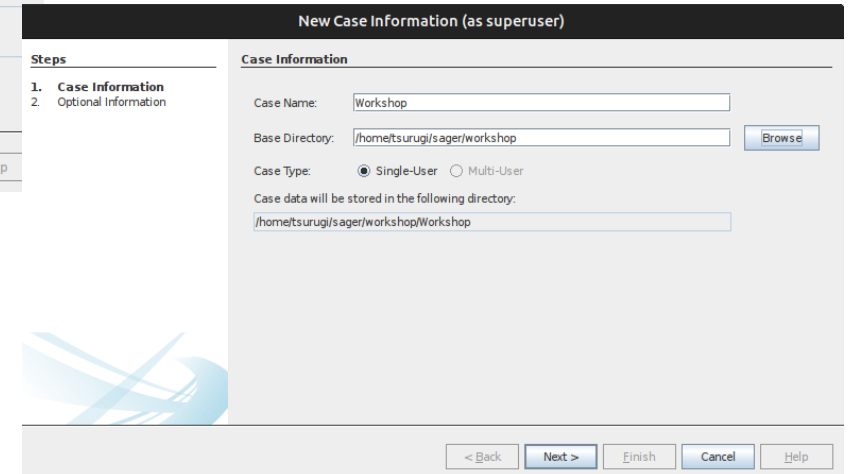
Email:

Notes:

Organization

Organization analysis is being done for:

< Back Next > Finish Cancel Help



New Case Information (as superuser)

**Steps**

1. **Case Information**
2. Optional Information

**Case Information**

Case Name:

Base Directory:

Case Type: ☒ Single-User ☐ Multi-User

Case data will be stored in the following directory:

< Back Next > Finish Cancel Help





# Autopsy - Tilføje beviser / Disk images

The image displays a sequence of five screenshots from the Autopsy Digital Forensics application, illustrating the 'Add Data Source' process as a superuser.

- Screenshot 1: Add Data Source (as superuser) - Select Host**
  - Steps:**
    1. Select Host
    2. Select Data Source Type
    3. Select Data Source
    4. Configure Ingest
    5. Add Data Source
  - Select Host**

Hosts are used to organize data sources and other data.

☒ Generate new host name based on data source name

☐ Specify new host name

☐ Use existing host
- Screenshot 2: Add Data Source (as superuser) - Select Data Source**
  - Steps:**
    1. Select Host
    2. Select Data Source Type
    3. Select Data Source
    4. Configure Ingest
    5. Add Data Source
  - Select Data Source**

Path:

☐ Ignore orphan files in FAT file systems

Time zone:

Sector size:

Hash Values (optional):

MDS:

SHA-1:

SHA-256:

NOTE: These values will not be validated when the data source is added.
- Screenshot 3: Add Data Source (as superuser) - Select Data Source Type**
  - Steps:**
    1. Select Host
    2. Select Data Source Type
    3. Select Data Source
    4. Configure Ingest
    5. Add Data Source
  - Select Data Source Type**
    - ☒ Disk Image or VM File
    - ☐ Local Disk
    - ☐ Logical Files
    - ☐ Unallocated Space Image File
    - ☐ Autopsy Logical Imager Results
    - ☐ XRY Text Export
- Screenshot 4: Add Data Source (as superuser) - Configure Ingest**
  - Steps:**
    1. Select Host
    2. Select Data Source Type
    3. Select Data Source
    4. Configure Ingest
    5. Add Data Source
  - Configure Ingest**

Run ingest modules on:

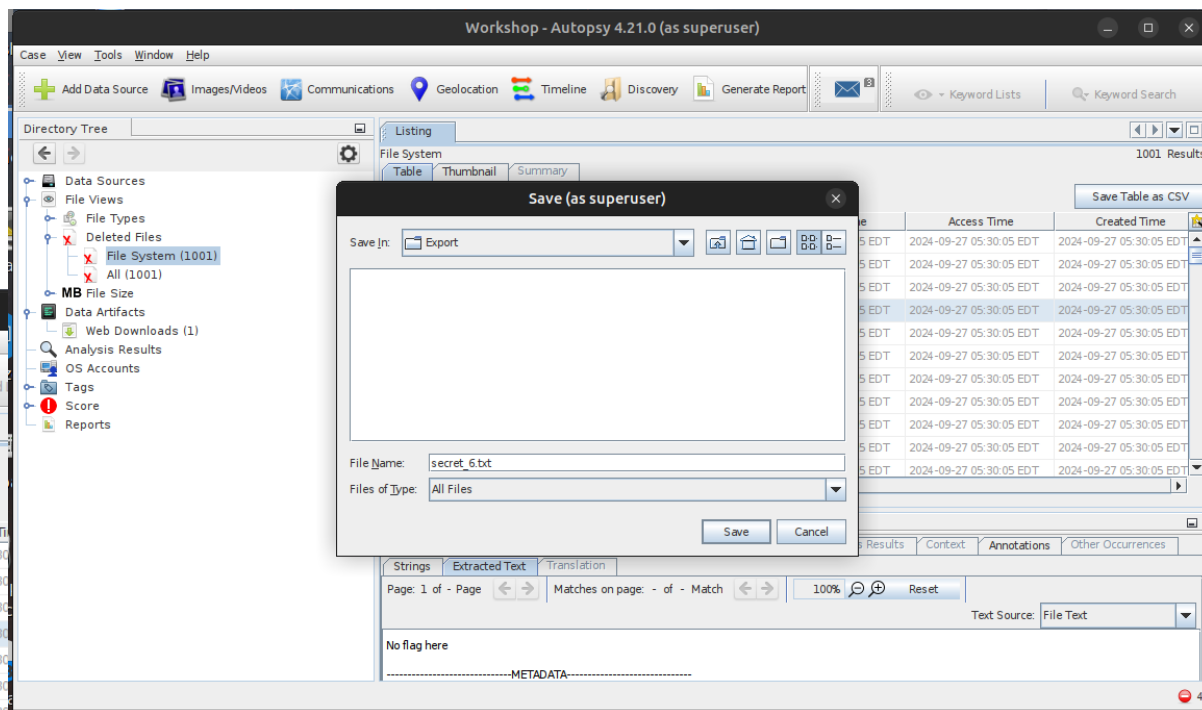
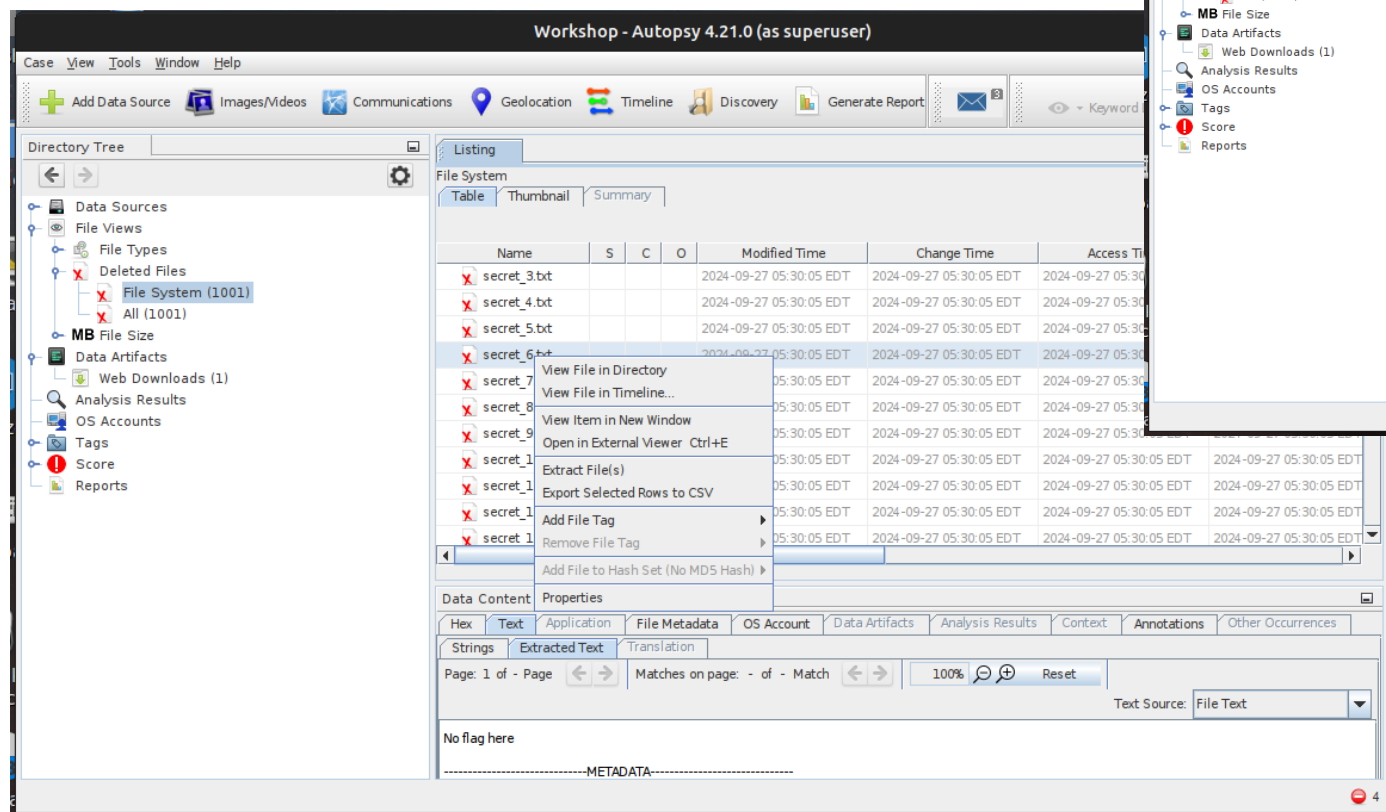
    - ☒ Recent Activity
    - ☐ Hash Lookup
    - ☐ File Type Identification
    - ☐ Extension Mismatch Detector
    - ☐ Embedded File Extractor
    - ☐ Picture Analyzer
    - ☐ Keyword Search
    - ☐ Email Parser
    - ☐ Encryption Detection
    - ☐ Interesting Files Identifier
    - ☐ Central Repository
    - ☐ PhotoRec Carver
    - ☐ Virtual Machine Extractor
    - ☐ Data Source Integrity

The selected module has no per-run settings.

Extracts recent user activity, such as Web browsing, recently ...
- Screenshot 5: Add Data Source (as superuser) - Confirmation**
  - Steps:**
    1. Select Host
    2. Select Data Source Type
    3. Select Data Source
    4. Configure Ingest
    5. Add Data Source
  - Add Data Source**

Data source has been added to the local database. Files are being ingested.

# Autopsy - Analyse af filer



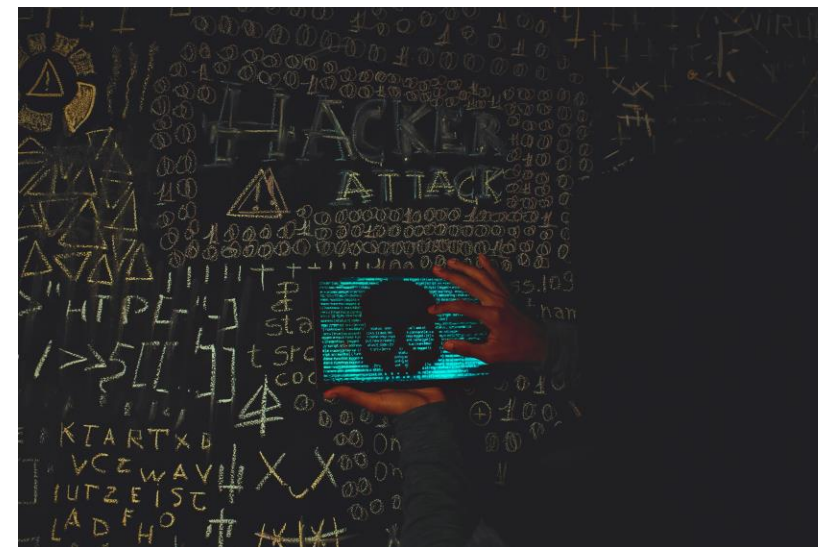
# Analyse af script og binære filer

- Mål ved analyse af malware
  - Finde flere IoC'er til forensics
  - Finde C2 infrastruktur
  - Finde ud af aktør (attribuering)
- Malware kan være lavet i script sprog (python, powershell, ...)
- Malware kan være lavet i binære sprog (C, C++, Golang, Rust, ...)



# Hvordan kigger man på malware?

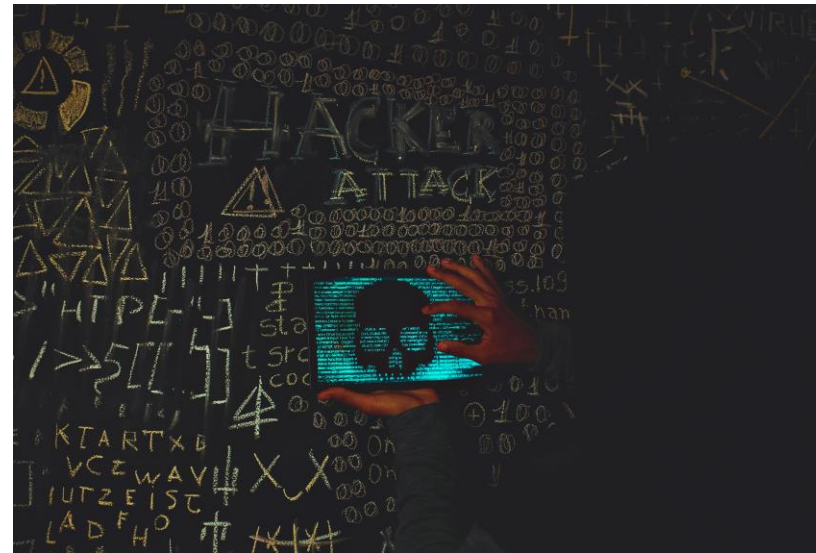
- I en VM
- På en maskine uden internet
- Sandbox – execution platforms
- Generelt: Meget forsigtigt!



- PS: I denne workshop er der ingen aktiv malware I skal kigge på!

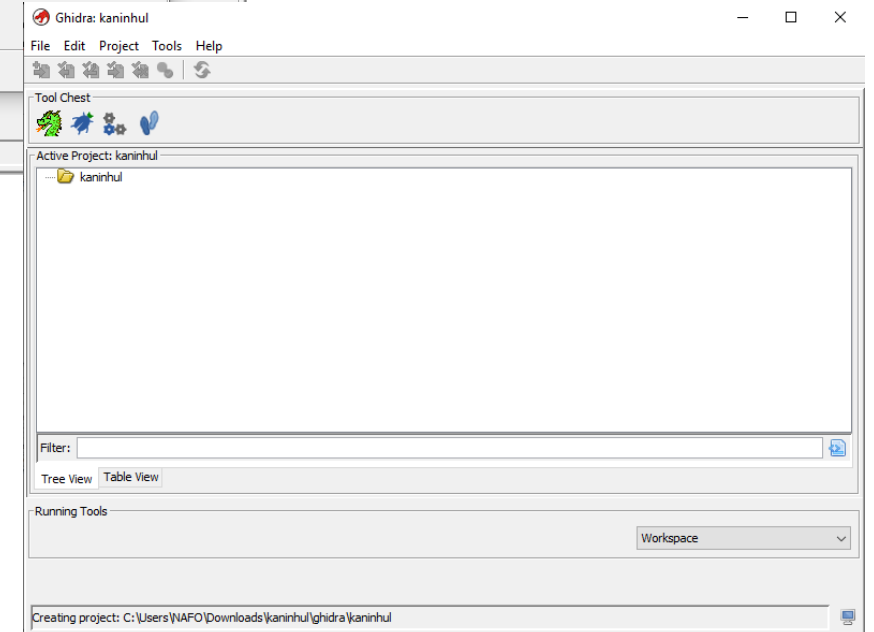
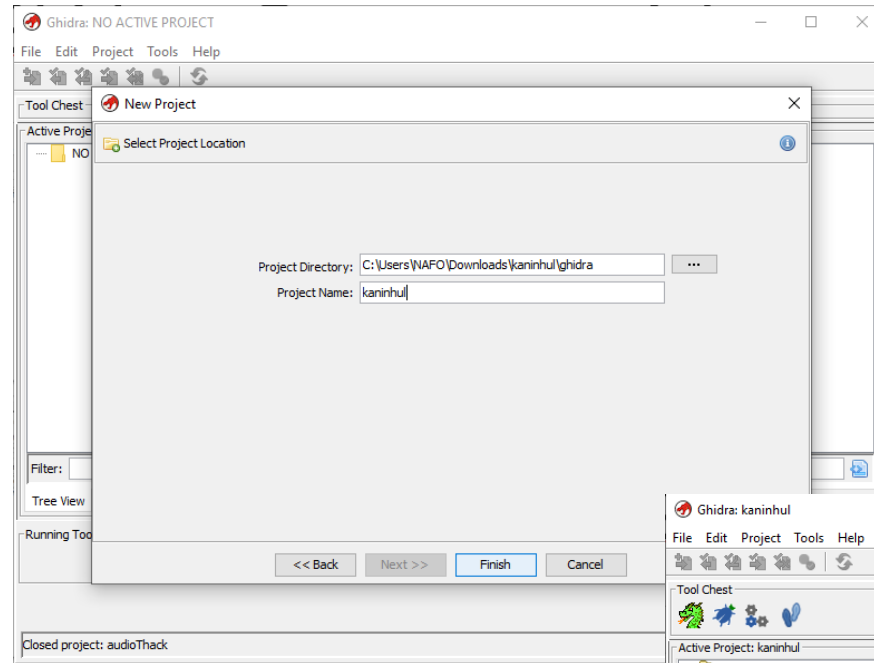
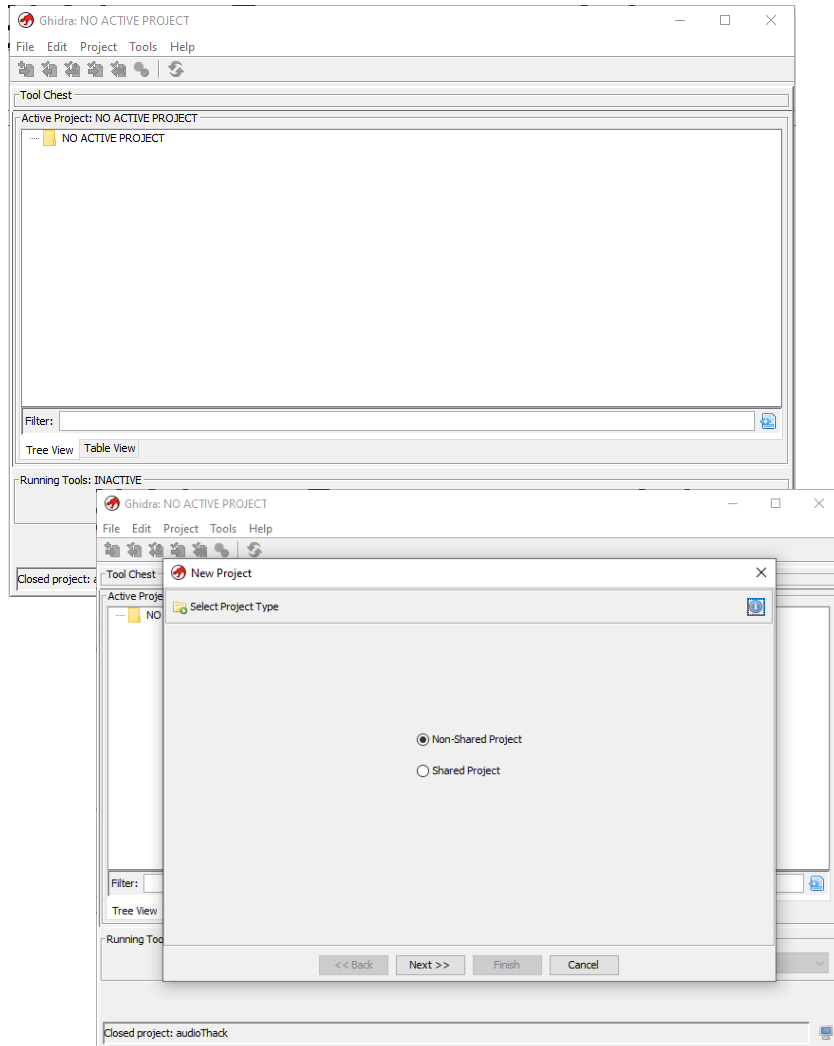
# Reversing binære filer

- Scripts – kan kigges på i notepad eller anden tekst editor
- Opret nyt projekt
- Tilføjelse af fil
- Analyse af fil

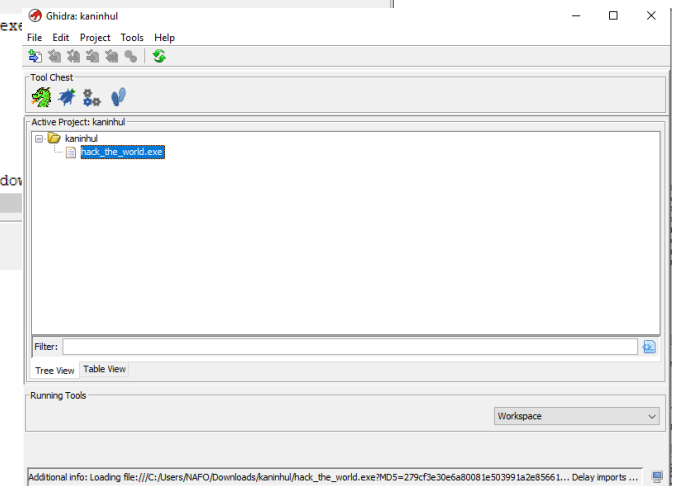
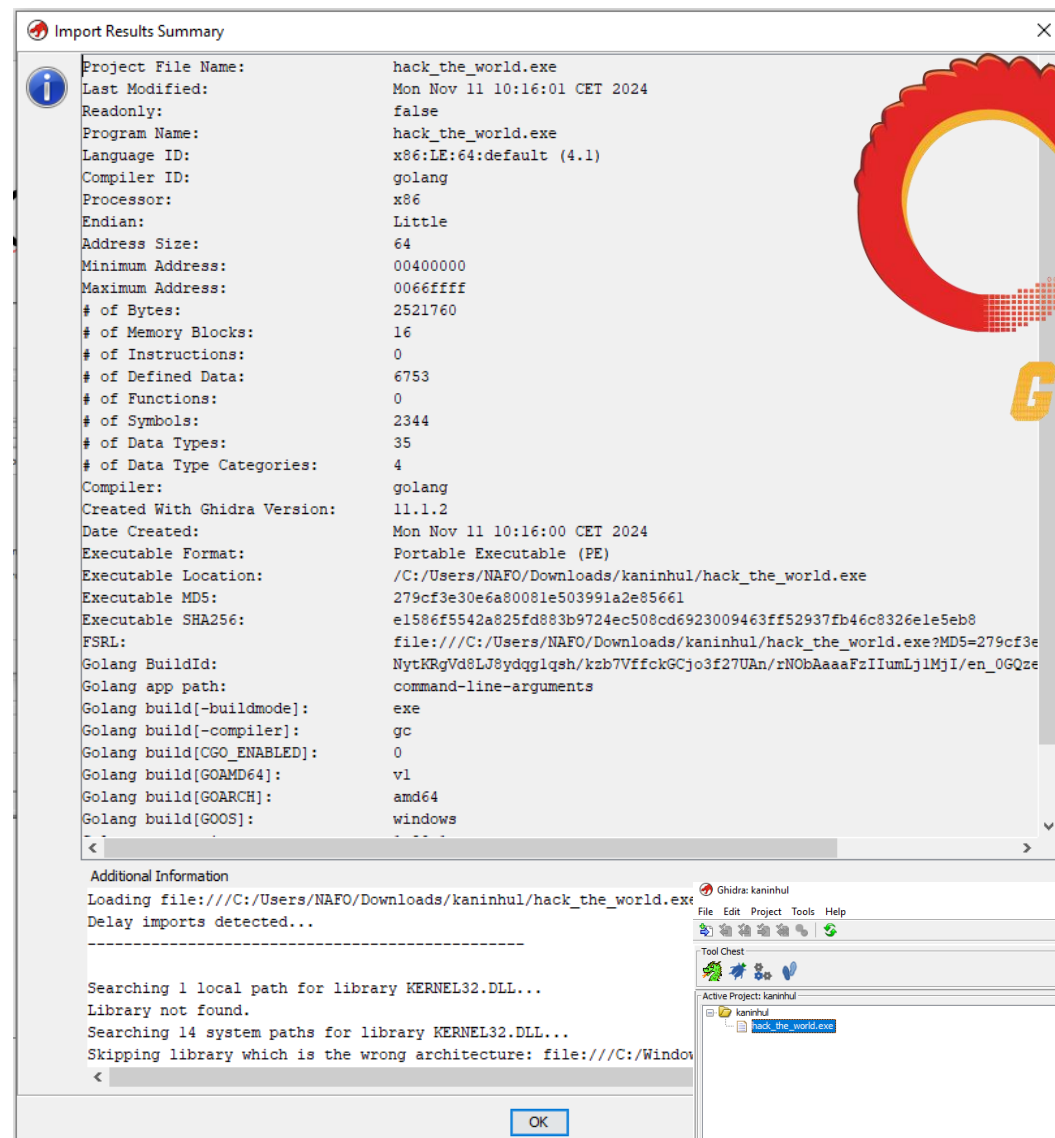
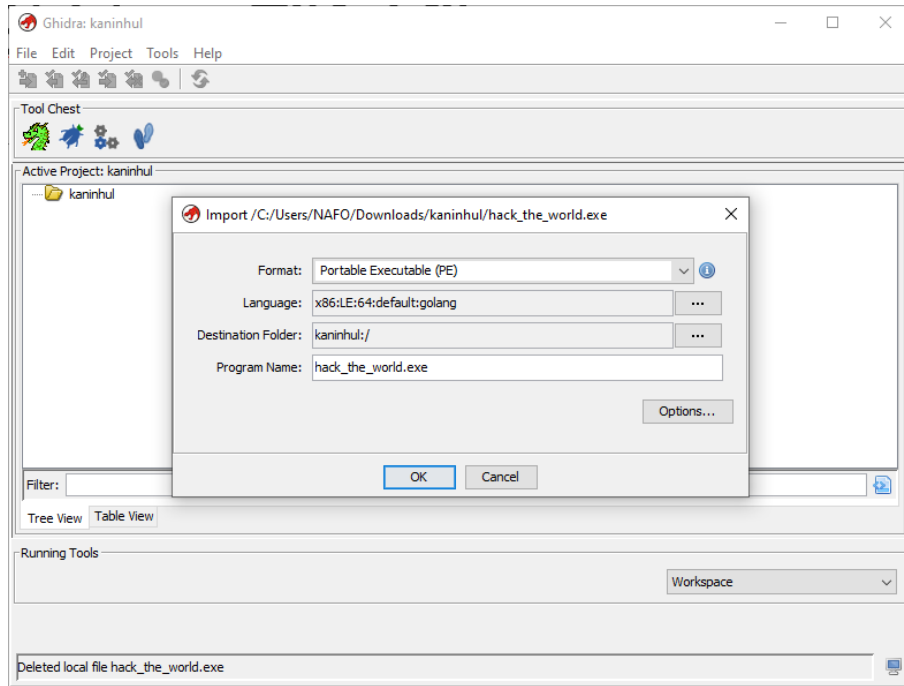




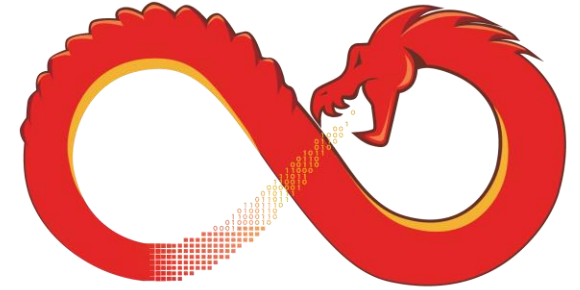
# Ghidra – Opret nyt projekt



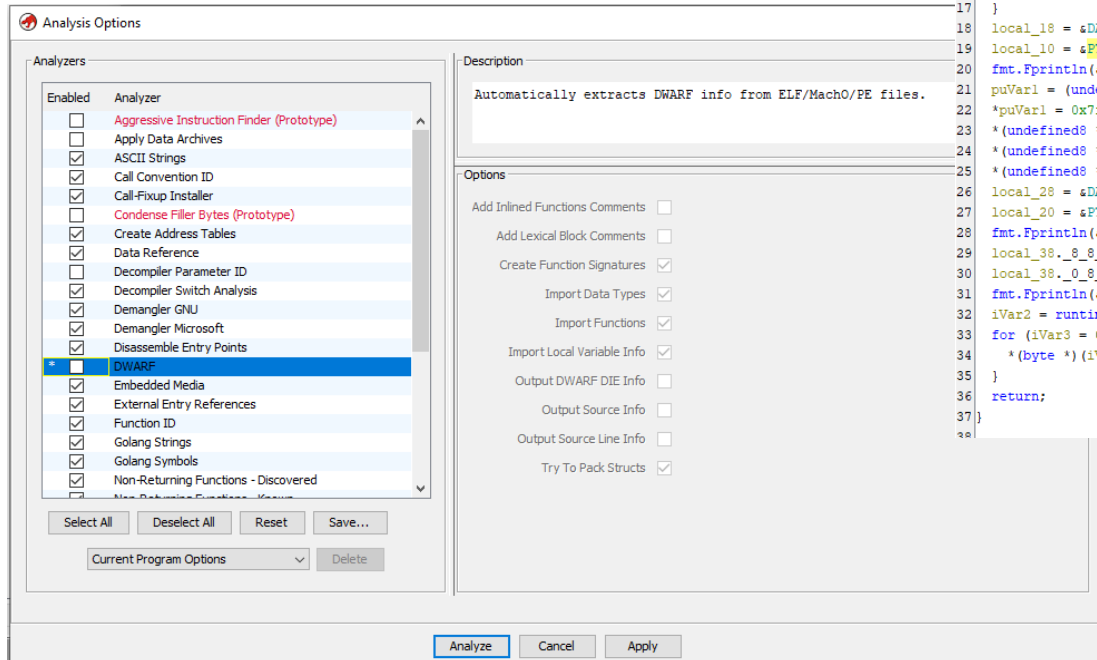
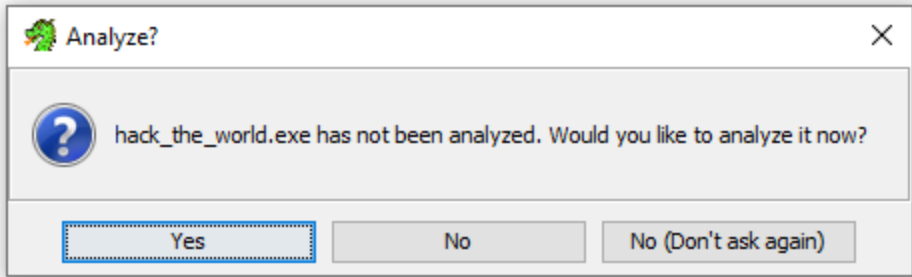
# Ghidra – Tilføj fil



# Ghidra – Analyse af fil



**GHIDRA**



```
Decompile: main.main - (hack_the_world.exe)
1
2 void main.main(void)
3
4 {
5     undefined *puVar1;
6     int iVar2;
7     int iVar3;
8     int unaff_R14;
9     undefined local_38 [16];
10    undefined *local_28;
11    undefined **local_20;
12    undefined *local_18;
13    undefined **local_10;
14
15    while (stack0x00000000 <= *(undefined **) (unaff_R14
16        runtime.morestack_noctxt.abi0());
17    }
18    local_18 = &DAT_004a2180;
19    local_10 = &PTR_s_Congratz_You_found_the_flag_KE_004e05c0;
20    fmt.Fprintln(sgo:itab.*os.File,io.Writer,os.Stdout,&local_18,1,1);
21    puVar1 = (undefined *)runtime.newobject(&DAT_004a31e0);
22    *puVar1 = 0x7f;
23    *(undefined8 *) (puVar1 + 1) = 0x436b46044c4f7571;
24    *(undefined8 *) (puVar1 + 9) = 0x1415e6b015f4604;
25    *(undefined8 *) (puVar1 + 0x11) = 0x4908480404586b03;
26    local_28 = &DAT_004a2180;
27    local_20 = &PTR_s_You_will_also_get_an_encrypted_f_004e05d0;
28    fmt.Fprintln(sgo:itab.*os.File,io.Writer,os.Stdout,&local_28,1,1);
29    local_38._0_0_ = runtime.convTslice(puVar1,0x19,0x19);
30    local_38._0_0_ = &DAT_004a13c0;
31    fmt.Fprintln(sgo:itab.*os.File,io.Writer,os.Stdout,local_38,1,1);
32    iVar2 = runtime.makeslice(&DAT_004a2200,0x19,0x19);
33    for (iVar3 = 0; iVar3 < 0x19; iVar3 = iVar3 + 1) {
34        *(byte *) (iVar2 + iVar3) = puVar1[iVar3] ^ 0x34;
35    }
36    return;
37 }
```

```
PTR_s_Congratz_You_found_the_flag_KE_004e05c0 XREF[2]:    main.main:004981be(*),
                                                         main.main:004981c5(*)
004e05c0 3d 37 4c      addr      s_Congratz_You_found_the_flag_KE_004c1320+9245 = "Congratz. You found the flag:...
               00 00 00
               00 00
004e05c8 40             ??      40h      0
= "Congratz. You found the flag: KEA{g0,
argument was allocated into an arenac
```

# Offensive hacker teknikker

- Portskanning
- Sårbarhedsskaning
- SQL-injection angreb

# Portskanning

- Portskanning giver et indledende overblik over hvilke services der kører
- Typiske nmap kald:
  - `nmap -F -oG output.txt --open <ip>`
- Dette kan give et overblik over hvilke porte der er åbne blandt de 1000 mest almindelig porte. Kan bruges som første spadestik ind i en ny penetrationstest
- Der er ingen grund til at portskanne i denne workshop



# Sårbarhedsskanning

- En sårbarhedsskanning kan målrette en penetration test
- Der findes værktøjer der direkte kan identificere sårbarheder
  - Eksempler på værktøjer: nuclei og nmap (Der findes mange andre)
- En anden slags sårbarhed er configurations fejl, såsom åbne webfoldere
  - Brug dirb, dirbuster, nmap eller lignende værktøjer til at finde disse
- Nogle gange kan der også identificeres sårbarheder med shodan.io

# SQL Injection

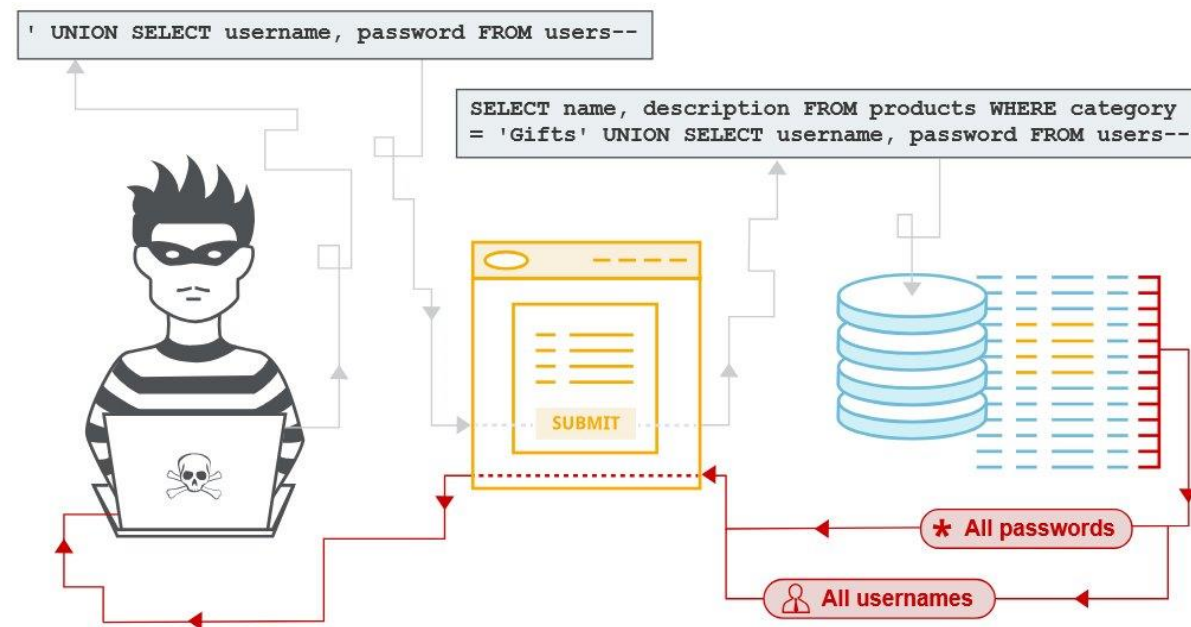
---

- TYPISKE STEDER AT KIGGE EFTER SQLi
- TRICKS TIL AT IDENTIFICERE SQLi SÅRBARHEDER
- EKSEMPLER PÅ SQLi
- KØRE SQLi PÅ JUICEBOX



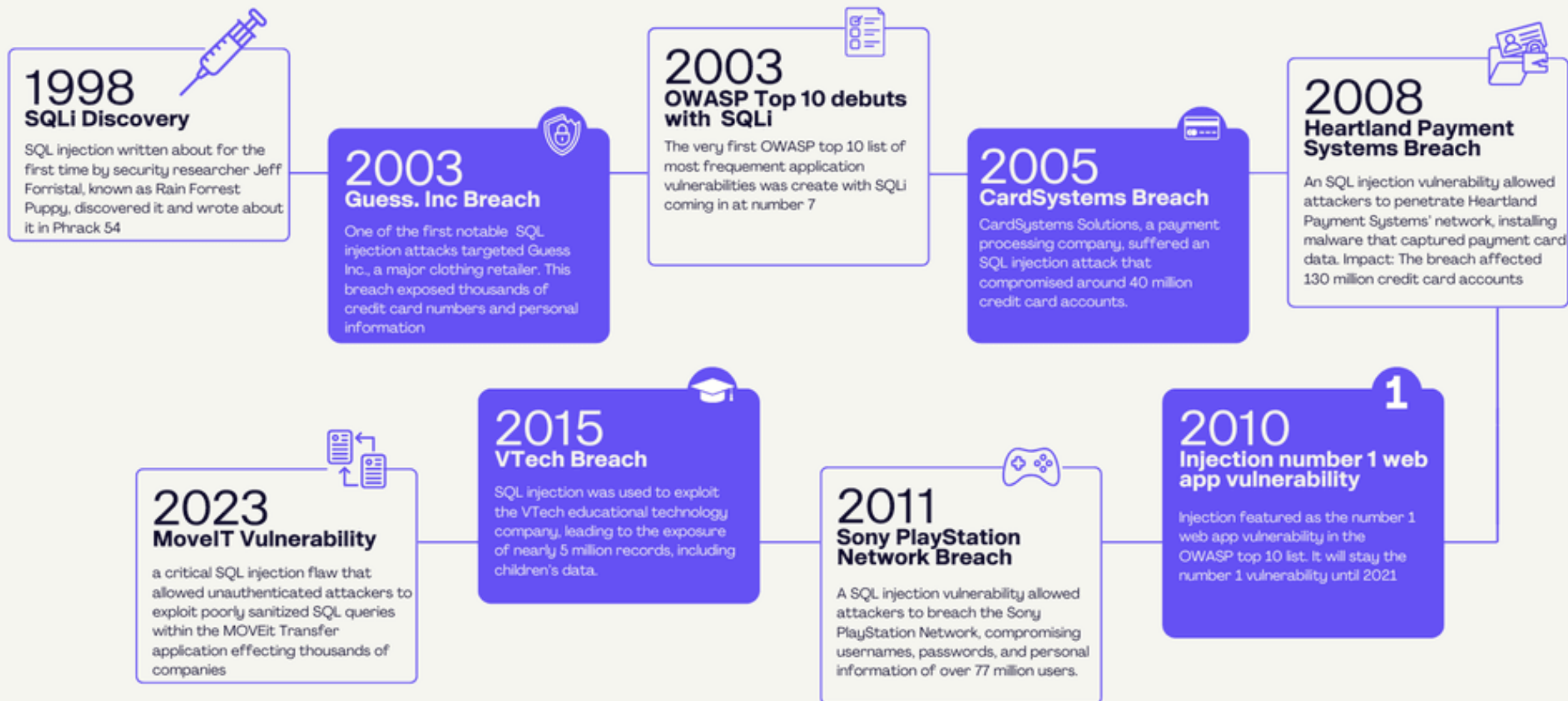
# Hvad er en SQL injection? (SQLi)

- SQL injection (SQLi) er en web-sikkerhedssårbarhed, der tillader en angriber at forstyrre de forespørgsler, som en applikation laver til sin database.
- Dette kan tillade en angriber at se data, som de normalt ikke kan hente.
- Det kan inkludere data, der tilhører andre brugere, eller andre data, som applikationen har adgang til. I mange tilfælde kan en angriber ændre eller slette disse data, hvilket forårsager vedvarende ændringer i applikationens indhold eller adfærd.
- I nogle situationer kan en angriber eskalere et SQL injection-angreb for at kompromittere den underliggende server eller anden back-end infrastruktur. Det kan også give dem mulighed for at udføre denial-of-service-angreb.



<https://portswigger.net/web-security/sql-injection>

# SQL INJECTION A HISTORY





# Historiske hacks med SQLi

---

## ResumeLooters

Active:  
beginning  
of 2023 – present

Activity:  
SQL injection,  
XSS attacks



### Tools:

- sqlmap
- Acunetix
- Beef Framework
- X-Ray
- Metasploit
- ARL
- Dirsearch

Known victims:  
65 companies

Geography:  
APAC-focused



<https://www.group-ib.com/media-center/press-releases/resumelooters/>



# Hvordan opdager man SQLi sårbarheder?

Du kan manuelt opdage SQL injection ved at bruge et systematisk sæt af tests mod hvert indgangspunkt i applikationen. For at gøre dette, vil du typisk indsende:

- Et enkelt anførselstegn ' og kigge efter fejl eller andre anomalier.
- SQL-specifik syntaks, der evaluerer til den oprindelige værdi af indgangspunktet, og til en anden værdi, og kigge efter systematiske forskelle i applikationens svar.
- Booleske betingelser som **OR 1=1** og **OR 1=2**, og kigge efter forskelle i applikationens svar.
- Payloads designet til at udløse tidsforsinkelser, når de udføres inden for en SQL-forespørgsel, og kigge efter forskelle i den tid, det tager at svare.

# SQL forspørgsler:

<b>user_id</b>	<b>username</b>	<b>password</b>
<b>1</b>	'john_doe',	'password123'
<b>2</b>	'jane_smith'	'securepass'
<b>3</b>	'alice_jones'	'mypassword'

```
SELECT user_id, username  
FROM users  
WHERE username = 'john_doe' AND password = 'password123';
```

<b>user_id</b>	<b>username</b>	<b>password</b>
<b>1</b>	'john_doe',	'password123'
<b>2</b>	'jane_smith'	'securepass'
<b>3</b>	'alice_jones'	'mypassword'

# Typiske steder at kigge efter SQLi

Alle steder hvor brugere kan indsætte tekst kan potentielt være en andgrebsflade for SQLi. Et klassisk eksempel er login formularer.

## Login

Username:

Password:

Login

## Dynamic SQL Statement:

```
SELECT * FROM users WHERE username =  
'john_doe' AND password = 'password123'
```

# SQL injection I login input felt:

Ved at indsætte tegnet ' kan man unslippe det SQL kode som køres mod serveren og indskyde sin egen SQL og bygge videre på forspørglsen. Derved navnet SQL-Injection.

## Login

Username:

Password:

Login

Dynamic SQL Statement:

```
SELECT * FROM users WHERE username = '' or 1 = 1;' AND password = '1234'
```

Warning: Potential SQL Injection detected.

I eksemplet indsættes '**OR 1 = 1**' og derved vælges den første bruger I databasen.

# SQL injection I login input felt:

Ved at indsætte to bindestreger (uden mellemrum mellem) - - kan man ignorere efterfølgende SQL I en statement.

## Login

Username:

Password:

Login

## Dynamic SQL Statement:

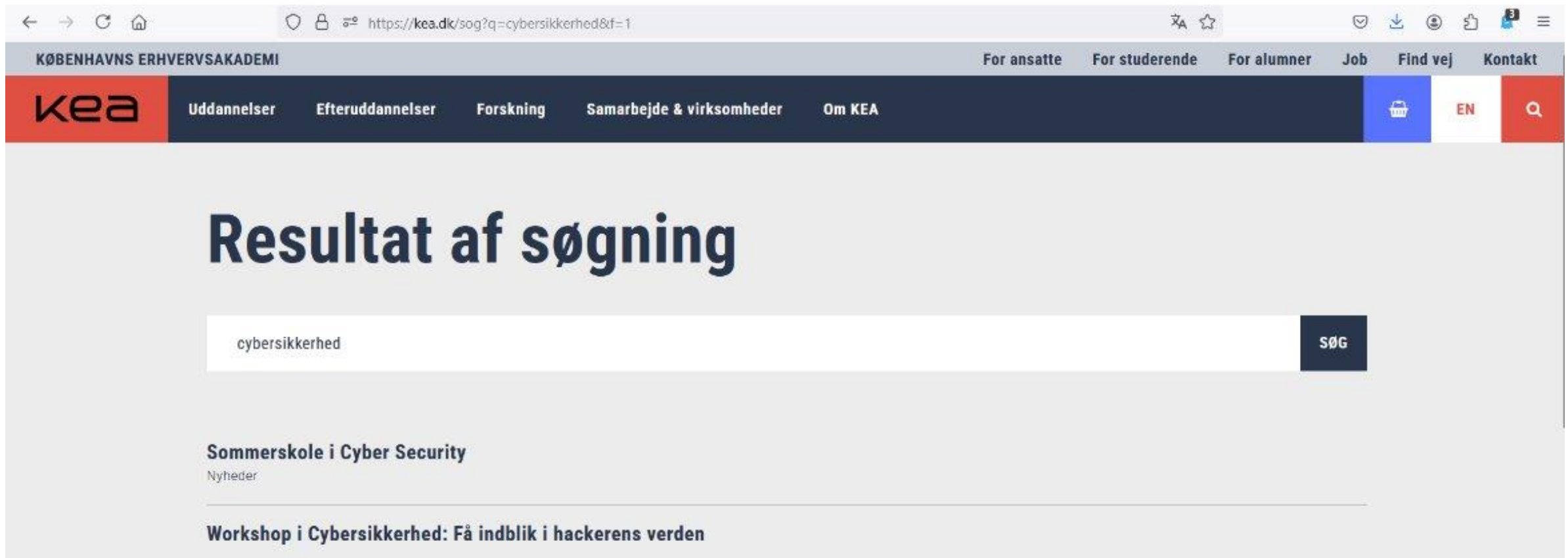
```
SELECT * FROM users WHERE username = 'someuser@website.com'--' AND  
password = '1234'
```

Warning: Potential SQL  
Injection detected.

I eksemplet vil man blive logget ind med brugernavnet og adgangskoden vil ignoreres.



# Typiske steder at kigge efter SQLi



The screenshot shows a web browser window with the URL `https://kea.dk/sog?q=cybersikkerhed&f=1`. The page is the KEA website, featuring a dark blue header with the KEA logo and navigation links. The main content area is light gray and displays the search results for 'cybersikkerhed'. The results include a section for 'Sommerskole i Cyber Security' with a 'Nyheder' link, and a section for 'Workshop i Cybersikkerhed: Få indblik i hackerens verden'.

KØBENHAVNS ERHVERVSAKADEMI

For ansatte For studerende For alumner Job Find vej Kontakt

kea Uddannelser Efteruddannelser Forskning Samarbejde & virksomheder Om KEA

## Resultat af søgning

cybersikkerhed **SØG**

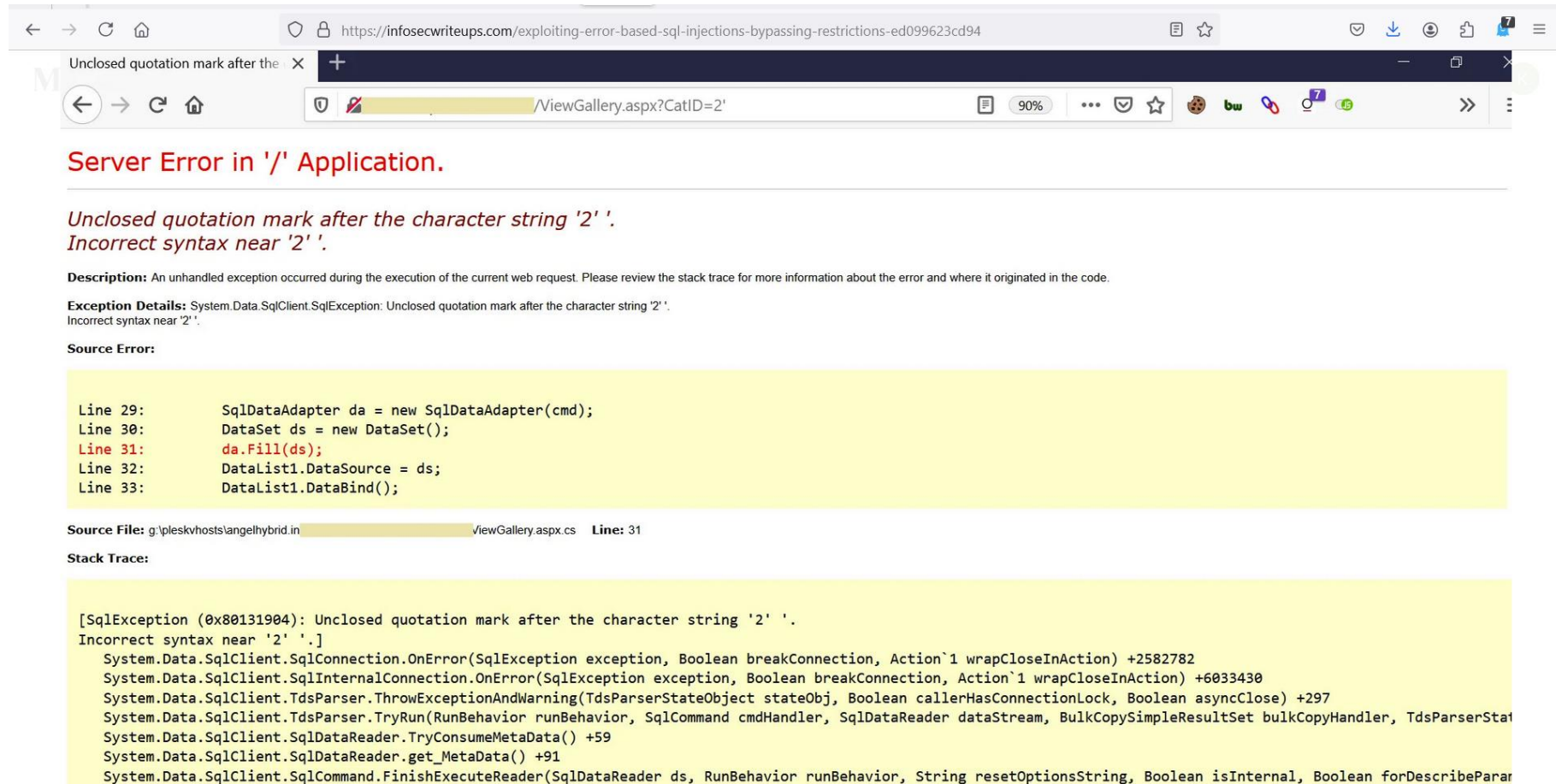
**Sommerskole i Cyber Security**  
Nyheder

**Workshop i Cybersikkerhed: Få indblik i hackerens verden**

# Typiske steder at kigge efter SQLi

The screenshot shows a web browser window with the address bar displaying `https://kea-fronter.itslearning.com`. The page content is for the KEA login interface. At the top, the KEA logo is displayed, followed by the text "Københavns Erhvervsakademi / Copenhagen School of Design and Technology". Below this, there is a link: [Ikke fra Københavns Erhvervsakademi / Copenhagen School of Design and Technology?](#). The login section is divided into two columns. The left column is for logging in with "itslearning" credentials, featuring a "Brugernavn" (Username) field with the value "some\_kea\_user@kea.dk", an "Adgangskode" (Password) field with masked characters and a toggle icon, and a "Log på" (Log in) button. The right column contains a message: "Der vises et nyt vindue, hvis du ikke allerede er logget på." and a button labeled "Log på med KEA Login". A vertical line with the word "ELLER" (OR) separates the two login options. At the bottom of the left column, there is a link: [Har du glemt adgangskoden?](#).

# SQL fejlmeddelelser kan give værdifuld information: Error Based SQLi



Unclosed quotation mark after the character string '2' '.

Server Error in '/' Application.

*Unclosed quotation mark after the character string '2' '.*  
*Incorrect syntax near '2' '.*

**Description:** An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

**Exception Details:** System.Data.SqlClient.SqlException: Unclosed quotation mark after the character string '2' '.

**Source Error:**

```
Line 29:      SqlDataAdapter da = new SqlDataAdapter(cmd);
Line 30:      DataSet ds = new DataSet();
Line 31:      da.Fill(ds);
Line 32:      DataList1.DataSource = ds;
Line 33:      DataList1.DataBind();
```

**Source File:** g:\pleskvhosts\angelhybrid.in\viewGallery.aspx.cs **Line:** 31

**Stack Trace:**

```
[SqlException (0x80131904): Unclosed quotation mark after the character string '2' '.
```

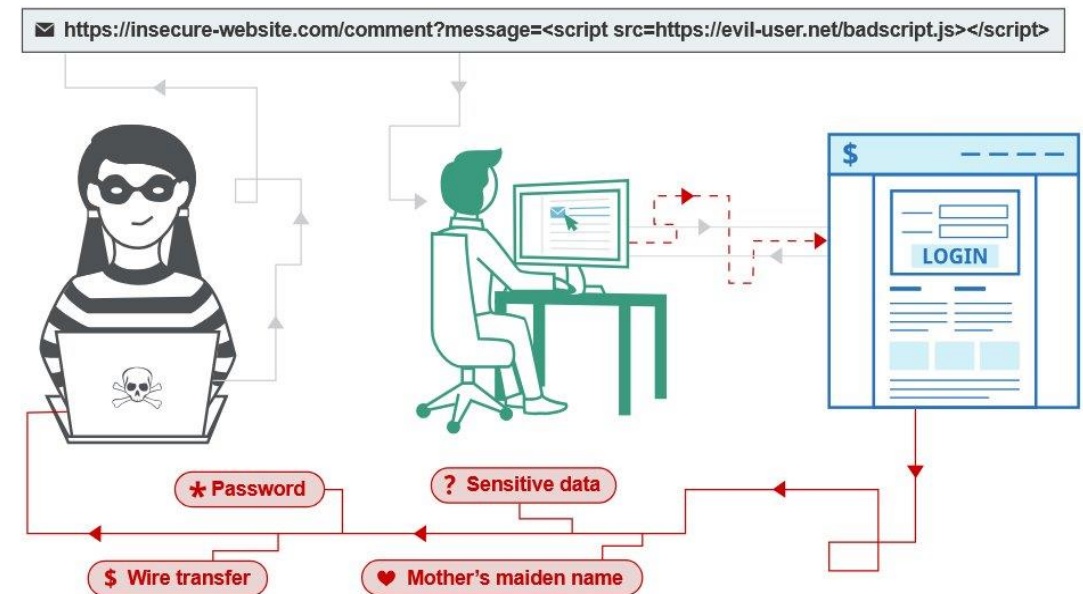
# cross-site scripting (XSS)

Cross-site scripting (også kendt som XSS) er en web-sikkerhedssårbarhed, der tillader en angriber at kompromittere de interaktioner, som brugere har med en sårbar applikation.

Det tillader en angriber at omgå same origin policy, som er designet til at adskille forskellige websites fra hinanden.

Cross-site scripting-sårbarheder tillader normalt en angriber at udgive sig for at være en offerbruger, udføre alle de handlinger, som brugeren er i stand til at udføre, og få adgang til alle brugerens data.

Hvis offerbrugeren har privilegeret adgang inden for applikationen, kan angriberen muligvis få fuld kontrol over al applikationens funktionalitet og data.

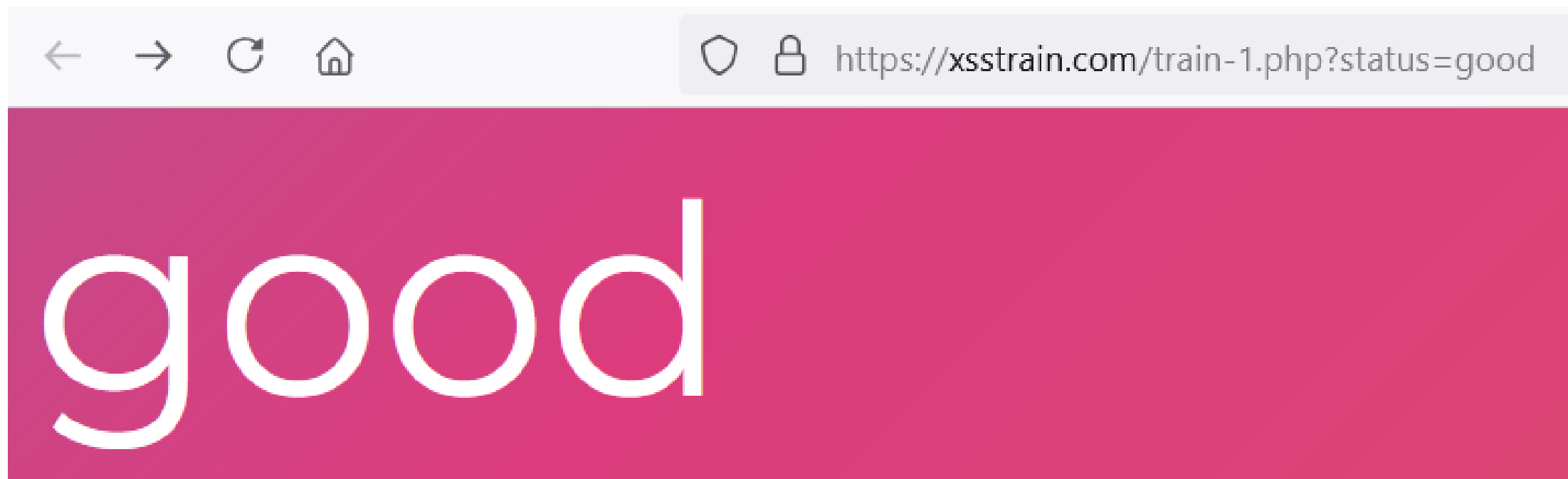


<https://portswigger.net/web-security/cross-site-scripting>

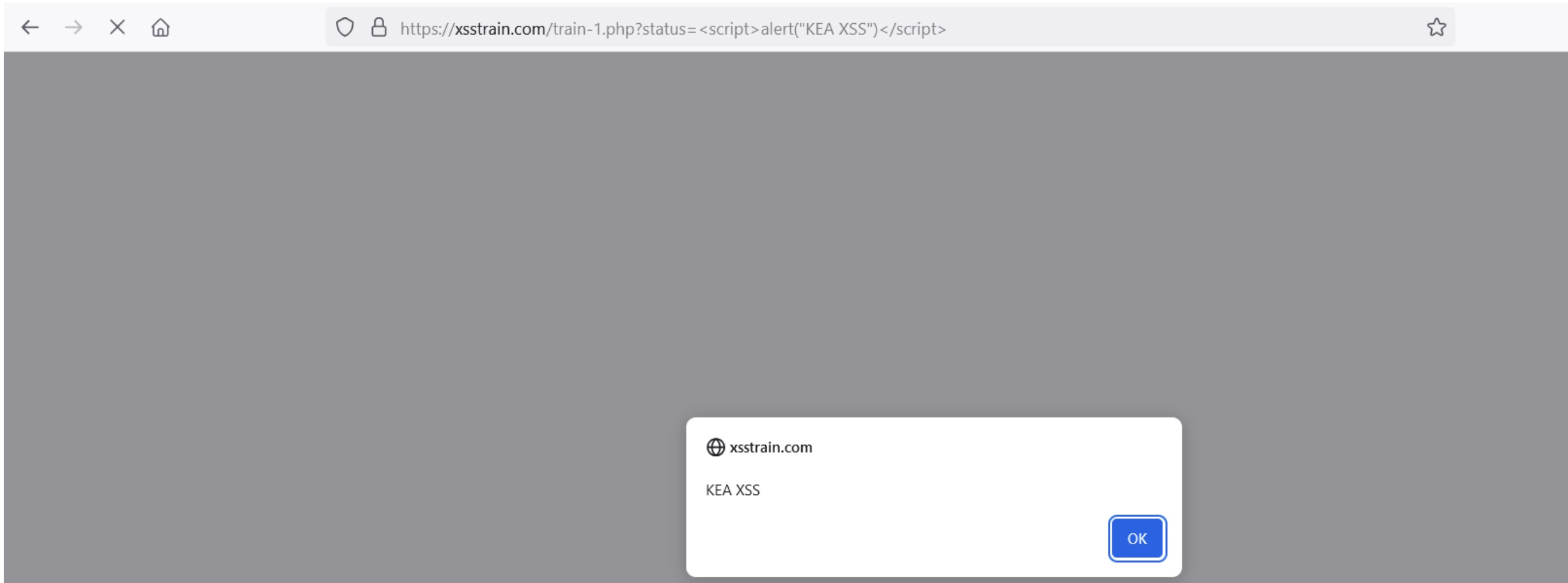
# Hvordan opdager man XSS sårbarheder?

**Cross-site scripting** fungerer ved at manipulere et sårbart websted, så det returnerer ondsindet JavaScript til brugerne.

Når den ondsindede kode udføres i en offers browser, kan angriberen fuldt ud kompromittere deres interaktion med applikationen.

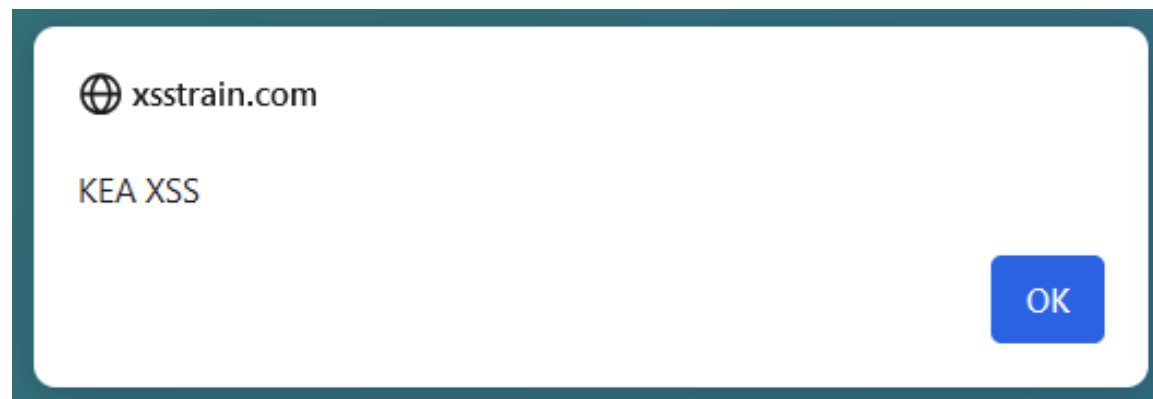


# Hvordan opdager man XSS sårbarheder?



# Hvordan opdager man XSS sårbarheder?

  [https://xsstrain.com/train-1.php?status=<script>alert\("KEA XSS"\)</script>](https://xsstrain.com/train-1.php?status=<script>alert('KEA XSS')</script>)



Du kan bekræfte de fleste typer af XSS-sårbarheder ved at injicere en payload, der får din egen browser til at udføre noget vilkårligt JavaScript.

Det har længe været almindelig praksis at bruge **alert()**-funktionen til dette formål, fordi den er kort, harmløs og ret svær at overse, når den kaldes med succes.

Faktisk løses mange XSS-labs ved at påkalde **alert()** i en simuleret offers browser.

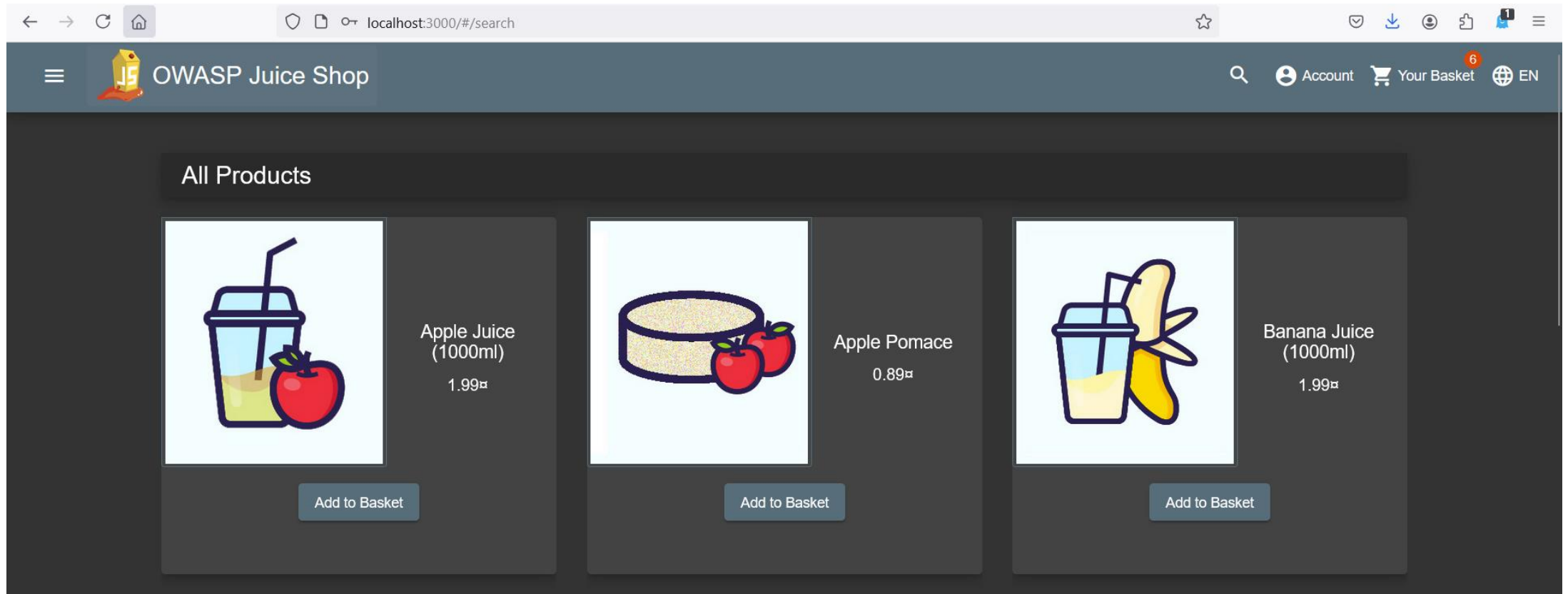




# Typer af XSS

- **Reflected XSS**, hvor det ondsindede script kommer fra den aktuelle HTTP-anmodning.
- **Stored XSS (persistent)**, hvor det ondsindede script kommer fra webstedets database.
- **DOM-based XSS**, hvor sårbarheden findes i klient-side kode i stedet for server-side kode

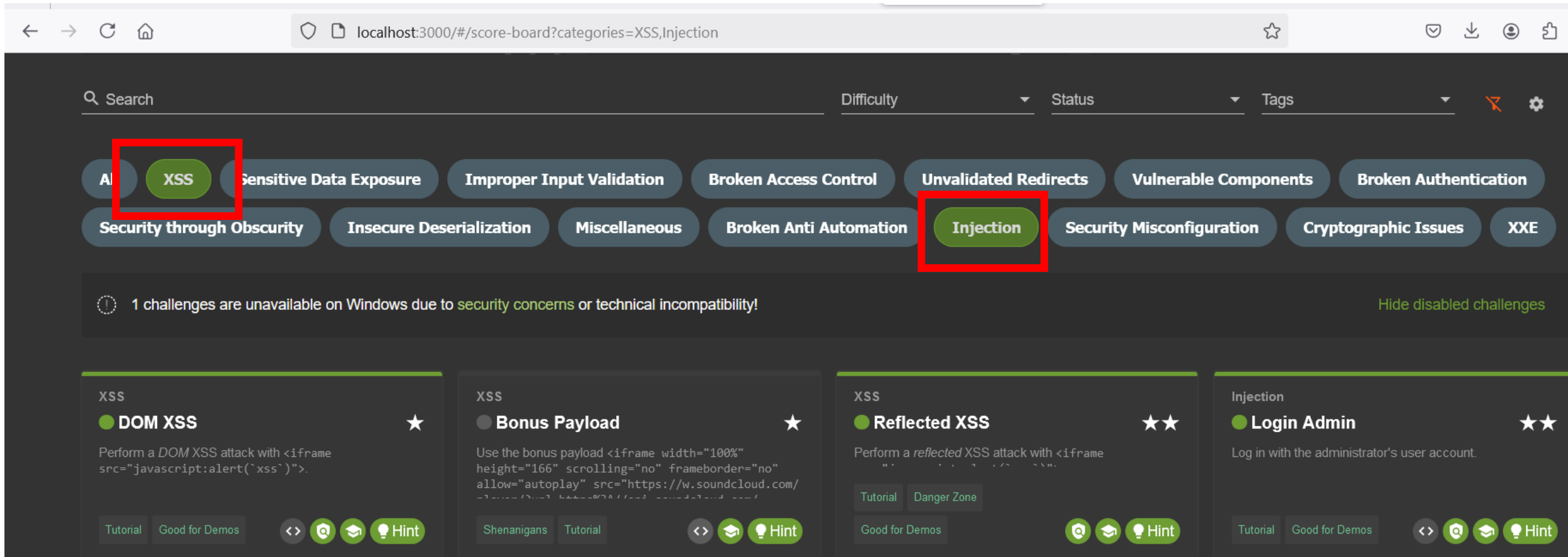
# OWASP - Juice shop




Den måske, mest avancerede sårbare hjemmeside:

<https://owasp.org/www-project-juice-shop/>

# Se forskellige challenges under /score-board



A large orange circle is positioned on the left side of the slide, partially overlapping the text.

# Nødvendige værktøjer til juice shop hacking


Mozilla Firefox browser

<https://www.mozilla.org/en-US/firefox/new/>

Postman (Kan anvendes til at sende HTTP requests)

<https://www.postman.com/>

Andre værktøjer er I selvfølgelig også velkomne til at bruge, men det er ikke et krav for at løse opgaverne.

A blue dashed line is located in the bottom right corner of the slide, consisting of several short, curved segments.

# Mozilla Firefox developer tools

For at åbne developer tools i Mozilla firefox browser kan man anvende følgende:

Toggle Developer Tools

F12

Ctrl

+

Shift

+

I

# Mozilla Firefox developer tools



The screenshot shows a Mozilla Firefox browser window displaying the KEA website. The address bar shows the URL `https://kea.dk/efteruddannelser`. The website header includes the KEA logo and navigation links: Uddannelser, Efteruddannelser, Forskning, Samarbejde & virksomheder, and Om KEA. The main content area features the heading "Efteruddannelser" and a paragraph about investing in future skills. A right-click context menu is open, highlighting the "Inspect (Q)" option, which is used to access the developer tools.

KØBENHAVNS ERHVERVSAKADEMI

For ansatte For studerende For alumner Job Find vej Kontakt

kea Uddannelser Efteruddannelser Forskning Samarbejde & virksomheder Om KEA

Forside / Efteruddannelser

## Efteruddannelser

Vil du investere i din fremtid og opgradere dine kompetencer? Hvis du er på udkig efter en hel efteruddannelse, så kan du på KEA vælge mellem 16 akademi- og diplomuddannelser. Vi har også 16 særlige uddannelser - sat sammen og målrettet til forskellige fagområder. Du har også mulighed for blot at tage et enkelt kursus, som passer til dine arbejdsopgaver.

Alle akademiuddannelser

- Save Page As...
- Save Page to Pocket
- Select All
- Take Screenshot
- View Page Source
- Inspect Accessibility Properties
- Inspect (Q)
- Block element...

Man kan også højreklikke på en side og åbne developer tools i Mozilla firefox browser, og vælge inspect.

# Mozilla Firefox developer tools

The screenshot displays the Mozilla Firefox Developer Tools interface for the OWASP Juice Shop application. The browser window shows the URL `localhost:3000/#/search?q=KEA`. The developer tools toolbar includes icons for Inspector, Console, Debugger, Style Editor, Performance, Memory, Storage, Network, Accessibility, and Application.

The **Inspector** panel is active, showing the **HTML** tab. The selected element is the `body` tag with classes `mat-app-background` and `bluegrey-lightgreen-theme`. The **Styles** panel shows the default user agent styles for the `body` element, including `background-color: #303030` and `color: #fff`. The **Layout** panel shows a box model diagram with dimensions: `margin: 8px`, `border: 0px`, `padding: 0px`, and `width: 1520px`.

The **HTML** panel shows the following structure:

```
<!-- Copyright (c) 2014-2024 Bjoern Kimminich & the OWASP Juice Shop contributors. ~ SPDX-License-Identifier: MIT-->
<!DOCTYPE html>
<html class="fontawesome-i2svg-active fontawesome-i2svg-complete" lang="en">
  <head>
    <body class="mat-app-background bluegrey-lightgreen-theme">
      <div class="cc-window cc-floating cc-type-info cc-theme-classic cc-bottom cc-right cc-color-override--1225450786 cc-invisible" role="dialog" aria-live="polite" aria-label="cookieconsent" aria-describedby="cookieconsent:desc">
        <app-root _ngghost-woy-c132="" ng-version="15.2.10">
          <script src="runtime.js" type="module"></script>
          <script src="polyfills.js" type="module"></script>
          <script src="vendor.js" type="module"></script>
          <script src="main.js" type="module"></script>
          <div class="cdk-live-announcer-element cdk-visually-hidden" aria-atomic="true" aria-live="polite"></div>
          <div class="cdk-overlay-container bluegrey-lightgreen-theme">
            <div class="cdk-describedby-message-container cdk-visually-hidden" style="visibility: hidden;">
          </div>
        </app-root>
      </div>
    </body>
  </html>
```

The **Styles** panel shows the following styles:

```
element {
  background-color: #303030;
  color: #fff;
}
.bluegrey-lightgreen-theme.mat-app-background, .bluegrey-lightgreen-theme.mat-app-background {
  background-color: #303030;
  color: #fff;
}
.bluegrey-lightgreen-theme.mat-app-background {
  background-color: #303030;
  color: #fff;
}
.bluegrey-lightgreen-theme {
  --theme-primary: #546e7a;
  --theme-primary-lighter: #607e8c;
  --theme-primary-light: #698998;
  --theme-primary-darker: #485e68;
  --theme-primary-dark: #3f535c;
  --theme-primary-fade-10: rgba(84, 110, 122, .9);
  --theme-primary-fade-20: rgba(84, 110, 122, .9);
}
```

The **Layout** panel shows a box model diagram with dimensions: `margin: 8px`, `border: 0px`, `padding: 0px`, and `width: 1520px`.



# Mozilla Firefox developer tools

The screenshot displays the Mozilla Firefox Developer Tools interface. The browser window at the top shows the URL `localhost:3000/#/search` and the OWASP Juice Shop header. The Developer Tools panel is open, with the **Network** tab selected. It shows a list of network requests, with the first request, `saveLoginIp`, highlighted. The right-hand pane displays the **Headers** section for this request, showing a `304 Not Modified` status.

**Network Tab Requests:**

Status	Method	Domain	File	Initiator	Type	Transferred	Size
304	GET	localhost:3000	saveLoginIp	polyfills.js:1 (xhr)	json	cached	344 B
304	GET	localhost:3000	/api/Quantities/	polyfills.js:1 (xhr)	json	cached	6.26 kB
304	GET	localhost:3000	search?q=	polyfills.js:1 (xhr)	json	cached	14.79 kB
200	GET	localhost:3000	apple_juice.jpg	img	jpeg	cached	16.51 kB
200	GET	localhost:3000	apple_pressings.jpg	img	jpeg	cached	32.90 kB
200	GET	localhost:3000	banana_juice.jpg	img	jpeg	cached	20.61 kB
200	GET	localhost:3000	artwork2.jpg	img	jpeg	cached	36.99 kB
200	GET	localhost:3000	carrot_juice.jpeg	img	jpeg	cached	20.61 kB
200	GET	localhost:3000	user_day_ticket.png	img	png	cached	413.82 kB

**Headers Section (GET http://localhost:3000/rest/saveLoginIp):**

- Status: 304 Not Modified (?)
- Version: HTTP/1.1
- Transferred: 649 B (344 B size)
- Referrer Policy: strict-origin-when-cross-origin
- Request Priority: Highest
- DNS Resolution: System

**Response Headers (305 B):** Raw

# Øvelser - Forensics (4n6)

- Download fil I skal arbejde ud fra her:
  - <https://github.com/KEA-IT-TEKNOLOG/workshop/>
- Der er også nogle usb nøgler med øvelse og værktøjer



# Øvelser - Offensiv hacking

For at tilgå denne øvelse skal man forbinde til vores router først.

SSID: **IT-TEKNOLOG-2**

Password: **KeaTeknolog6!**

- På det lille papir står der hvilken IP og port I har fået tildelt.
  - Brug kun den IP og port.
- IP1: **192.168.0.170**
- IP2: **192.168.0.39**
- Port er fra **3010-3050** (begge inklusiv)

Et eksempel kunne se således ud: **192.168.0.39:3010**

- I må ikke DDoS'e eller på anden måde aktiv gå efter at ødelægge host maskinen. Du er ikke alene på hostmaskinen

# Kilder

- Billeder fra pexels.com
- Logoer fra produkternes hjemmesider
- Screenshots fra programmerne