Week-07 OWASP A6: Security Misconfiguration + A9 Using Components with Known Vulnerabilities

**Explain briefly about the A6 Security Misconfiguration Risk, including topics from:**

* Is the Application Vulnerable
* How to Prevent
* Example Attack Scenarios

*Explain and demonstrate a number of the new Security Headers used by modern browsers, emphasis on how they can increase security and the problem they protect against.*

*Demonstrate shortly how many properties of a sample (misconfigured) application a hacker can discover.*

*Explain, as many as you can come up with, Security Misconfigurations, made deliberately throughout this semester, to make it possible to demonstrate “real attacks”.*

Fundamentalt set er det når der medfølger en masse standard variabler som en hacker kan finde og misbruge, for eksempel en standard admin user der blev brugt til testing eller lign.

Security headers er headers der bliver sendt med responsen der øger sikkerheden når man besøger webappen. Eksempler på disse headers kunne være:

**Content-Security-Policy** - sørger for kun at whiteliste ønskede assets. <https://scotthelme.co.uk/content-security-policy-an-introduction/> (læs mere)

**X-Frame-Options** - gør det muligt at vælge om du vil have din kan blive framed eller ej.   
<https://scotthelme.co.uk/hardening-your-http-response-headers/#x-frame-options> (læs mere)

Andre headers:

**Referrer-Policy -** [**https://scotthelme.co.uk/a-new-security-header-referrer-policy/**](https://scotthelme.co.uk/a-new-security-header-referrer-policy/)

**Feature-Policy -** [**https://scotthelme.co.uk/a-new-security-header-feature-policy/**](https://scotthelme.co.uk/a-new-security-header-feature-policy/)

**X-Content-Type-Options -** [**https://scotthelme.co.uk/hardening-your-http-response-headers/#x-content-type-options**](https://scotthelme.co.uk/hardening-your-http-response-headers/#x-content-type-options)

**Strict-Transport-Security -** [**https://scotthelme.co.uk/hsts-the-missing-link-in-tls/**](https://scotthelme.co.uk/hsts-the-missing-link-in-tls/)

**Explain briefly about the OWASP A9 threat:**

Provide and demonstrate practical examples for how to ensure that our maven dependencies do not contain Know Vulnerabilities? (Feel free to replace Java/maven above with examples from other technologies such as JavaScript, Python ..)

Demonstrate shortly how many properties of a sample (misconfigured) application a hacker can discover.

Her er Juice shop et perfekt eksempel at bruge, da det er lavet som et sikkerhedsmæssigt dårligt programmeret.

Et fint eksempel som kan vises ved starten af juice shop er når man prøver at logge ind og skriver et ‘ tegn i brugernavns feltet. Inspicerer man det under netværks tabben og går under response kan man se en sql query, hvor man derfra så kan skrive ‘1 OR 1=1 --