

Keywords er ideer og inspiration. I kan IKKE nå at gennemgå allesammen i de 10 minutter i præsentationer

## 1.Trusler mod software, oversigt over hvordan sårbarheder i software opstår

---

Green book Design vs implementation plus configuration

OWASP Juice Shop - examples of real vulns

Special cases - use dates like september 1752, leap year etc.

Kan bruge alt hvad vi har talt om på kurset

## 2.Sikkerhed i udviklingsprocesser, Secure Software Development Lifecycle

---

SSDL, SDLC and Risk Ranking

Phases of SSDL

Roles and Responsibilities

Software is insecure

How do we improve quality

Higher quality is more stable, and more secure

A full lifecycle approach is the only way to achieve secure software.

–Chris Wysopal

## 3.Sikkerhed i web applikationer

---

Web Application Security book

OWASP Top 10, guides, OWASP JuiceShop

REST, XML, JSON, APIs,

Using frameworks like Django, Python XML etc.

Tools Nikto, sslscan

## 4.Softwareproblemer med håndtering af hukommelse

---

Buffer overflow - stack based buffer overflow demo.c

C language issues - C is everywhere

Art of Exploitation book

Data types - int, long, short, unsigned, signed

Integer overflow/underflow

Stack protection and related technologies

Advanced ROP - hvis I lyster

## 5.Forbedret sikkerhed med opbygning af software i komponenter

---

Program Building Blocks

Design Patterns

Code Patterns

Libraries, Modules, Frameworks

Common data structures - don't implement them yourself if you can avoid

Development policies, SSDL

## 6.Håndtering af tekststrengene i software, herunder tegnsæt

---

Strings and Metacharacters

C string handling

Bad functions strcpy, good functions strncpy

ASCII - Unicode

memory corruption due to string mishandling

Vulnerabilities due to in-band control data in the form of metacharacters

Vulnerabilities resulting from conversions between character encodings in different languages

Format string vulns

Metacharacters ' ` # % & \$ ; etc.

Shellshock, shell escape etc.

## 7. Netværksangreb mod software

---

Auditing Application Protocols - sniffing og Wireshark

HTTP GET -- request med buffer overflow

ASN.1 - Linux kernel has 5 ASN.1 parsers!

TLV Type Length Value

DNS binary protocol - sammenlign med HTTP

SYN flooding -- denial of service

SNI Exim CVE-2019-15846

ERNW WHITEPAPER 68, SECURITY ASSESSMENT OF CISCO ACI, 2019 - buffer overflow i LLDP

## 8. Audit af software, samt almindelige fejl der skal håndteres

---

Any security policy, mechanism, or procedure is based on assumptions that, if incorrect, destroy the super-structure on which it is built.

Matt Bishop, Computer Security 2019

Access Control

Exceptions

Confidentiality Integrity and Availability

Security is a process, Bruce Schneier

Auditing vs black box

Fuzzing vs code reading

Incorporating audit into development processes SDLC, continuous integration, Git

Common problems, strcpy, memory, strings, ...etc.

Attack trees, reducing attack surface

## 9. Security design og principper for sikkert design

---

Security Principles for software

Principle of least privilege, fail-safe defaults, separation of privilege etc.

Security by Design

Privacy by Design

Benchmarking standards

Security by Design Principles

[https://www.owasp.org/index.php/Security\\_by\\_Design\\_Principles](https://www.owasp.org/index.php/Security_by_Design_Principles)

[https://en.wikipedia.org/wiki/Privacy\\_by\\_design](https://en.wikipedia.org/wiki/Privacy_by_design)

ENISA papers <https://www.enisa.europa.eu/publications/privacy-and-data-protection-by-design>

<https://www.enisa.europa.eu/publications/big-data-protection>

Defense in Depth