**Control questions – session 10**

1. **Name a few examples where encryption (symmetric and asymmetric) can be used in Cloud environment?**

I skymiljøet kan man bruke kryptering til å beskytte dataen (database, sensitiv data) mot datatyveri eller datatap. Kryptering kan også brukes til å beskytte applikasjonene i skyen mot at datainnbrudd. Hvis en hacker eller en angriper får tak i kildekoden til applikasjonen. Det kan brukes til DEVOPS (team som jobber med programvare utvikling /applikasjon) for å øke sikkerheten til applikasjonen.

**Kryptering er viktig for programvareutvikling**

Kryptering kan brukes under filoverføring mellom to parter (Ved mail). Nettverk er kryptert i cloud som bidrar til å gjøre forbindelsen mellom bedriften og kunden/ansatte sikker. Kryptering brukes til å kryptere monitorerings /analyse data som bidrar til å gi bedriften kun autorisasjon til disse dataene og ingen andre. Krypteringsmetodene som brukes i skyen er Symmetrisk og asymmetrisk kryptering metoder. Symmetrisk kryptering er når partene blir enige om en hemmelig nøkkel, som skal benyttes til både kryptering og dekryptering av tekst, filer i en mail (AES). Asymmetriske er når partene benytter seg av to nøkler (en offentlig og en privat) for kryptering og de kryptering av data. RSA er et eksempel på dette. Det finnes mange hashing algoritmer som MD5 og SHA som brukes for å hashe tekst eller data. viktig å benytte seg av en hashing algoritme som gjør det vanskelig for en angriper å kryptere.

1. **What is the difference between microservice and monolithic architecture? Which one is most common in Cloud and why?**

Monolithic arkitektur er bygget som en (mono) solid struktur og er den mest tradisjonelle måten å bygge applikasjoner på. Monolittiske applikasjoner er utviklet for å være selvstendige og strukturen består av en database , en brukergrensesnitt (UI user interface) og en app på serversiden. Disse er koblet sammen og avhengige av hverandre for at applikasjonskoden skal fungere. En applikasjon er ikke i stand til å fungere hvis en av komponentene mangler eller ved feil ikke er inkludert, årsaken er at de er tett koblet med hverandre og deler en felles database som til sammen skal utgjøre en applikasjon. En applikasjon med monothelic arkitektur er bygget opp av en stor felles kodebase.

Microservice arkitekturen er ulik monothelic fordi arkitekturen er bygget opp av mindre uavhengige komponenter som løst og koblet sammen. Dette er en mer moderne måte utvikle en applikasjon på. Komponentene er ikke bygget opp av en felles kodebase som i monothelitic arkitektur. Men komponentene har isteden fått utdelt hver sin egen database for eksempel : brukergrensesnittet har sin egen database , Det er en egen database for varslinger (notifications) osv.

Hva er best for cloud:

Det kommer an på hvilke mål bedriften har selv. Hvis de vil sørge for en stabil vekst velger de en microservice tilnærming, hvis ikke monothelic. Microservice er vanligvis mer kostbar grunnet vedlikehold fordi den har flere komponenter (microservicer) som må vedlikeholdes sammenliknet med monothelic der det vanligvis er få komponenter for vedlikehold. Hvis bedriften ønsker å en raskere applikajonsutvikling går de for monothelic tilnærmingen siden den ikke krever at mange tjenester kjøre samtidig og bidrar til en raskere oppstart av applikasjonen.

Mikrotjenester bedre teknisk, og tilbyr skalering (etter behov) fleksibilitet og ved å benytte seg av denne tilnærmingen gir brukere en bedre måte å administrere tjenestene på som databaser, skylagring. Dette bidrar også til å gjøre automatiseringen raskere og ta i bruk flere løsninger. Ettersom applikasjonen vokser er microservice arkitekturen et bedre valg fordi man har et stort behov for mikrotjenestene ettersom den voksen. Mikrotjenester er dyrere og på mange måter bedre enn monolitt. Dette er på mange måter den beste arkitekturen skyen kan benytte seg av.

**Mikrotjenestene har team jobber med hver spesifikt med en tjeneste en for databaser, et team for brukergrensesnittet osv. mikrotjenester gir tilgang på flere tjenester enn monolitt tjenester**



[**https://yellow.**systems**/blog/microservices-vs-monolithic**](https://yellow.systems/blog/microservices-vs-monolithic)

[**https://xbsoftware.com/blog/microservices**](https://xbsoftware.com/blog/microservices)

1. **Why is it important to have bug-free applications in Cloud? How can this affect the billing / monthly costs?**

[**https://www.vmware.com/topics/glossary/content/application-security.html#:~:text=Application%20security%20is%20important%20because,to%20security%20threats%20and%20breaches**](https://www.vmware.com/topics/glossary/content/application-security.html#:~:text=Application%20security%20is%20important%20because,to%20security%20threats%20and%20breaches)**.**

[**https://www.veracode.com/security/application-security-vulnerability-code-flaws-insecure-code**](https://www.veracode.com/security/application-security-vulnerability-code-flaws-insecure-code)

**Og session 10**

**Stikkord:**

* Det er viktig mot beskyttelse mot angrep og software exploits
* Hvis applikasjonen er usikker kan den være sårbar for sql injections , XSS og liknende.
* Det er viktig mot beskyttelse av sensitiv data , hvis ikke kan dette lede til datatap
* Sikre applikasjonen mot autentisering
* Krypterer innhold på applikasjonen sikrer mot datatyveri
* Sikkerhetsmekanismer gir beskyttelse
* Owasp top 10 gir en guide for applikasjonsutvikling
* Logging eller monitorering kan bidra til overvåkning av systemet eller applikasjonen ved sikkerhetsbrudd , logg gir info om vedkommende og alt av aktivitet hvis det har skjedd et innbrudd.
* Sikre applikasjonen krever mindre vedlikehold og usikre applikasjonen kan medføre til at bedriften mister kunden hvis de utsettes for datainnbrudd og liknende. Usikre applikasjonen har ingen form for sikkerhet og dette kan medføre til at kunden mister tillitt til appen og kan medføre til økonomisk tap hvis ingen bruker tjenesten og videre konkurs hvis bedriften ikke klarer å få inn noen kunder.

**4. What is the difference between DevOps and DevSecOps? Can’t we just add a security level to DevOps afterwards?**

**Skrev om dette se eksamensnotatene**