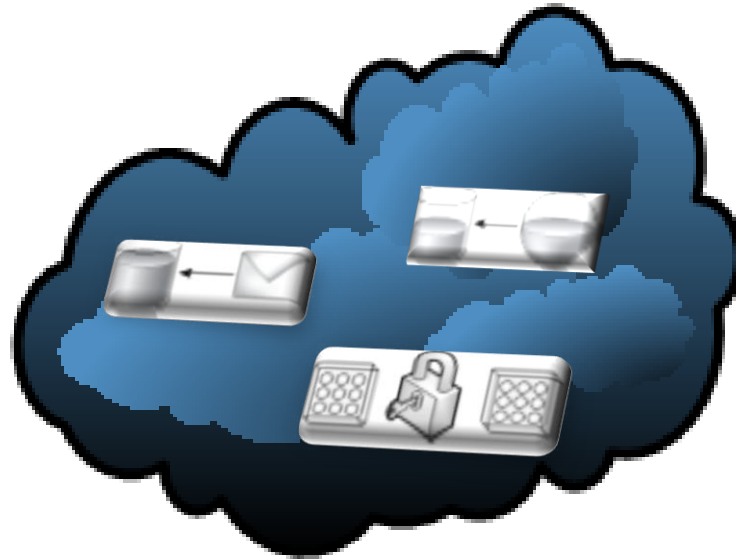


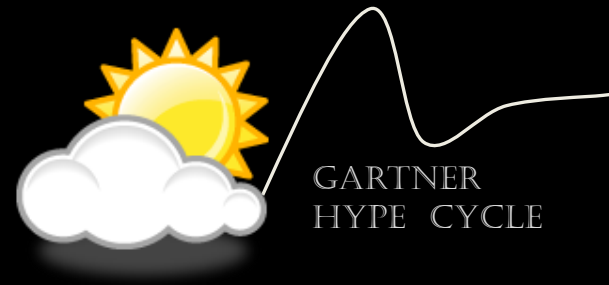
# SOA Designmönster i Molnet



SOA Summit 2010

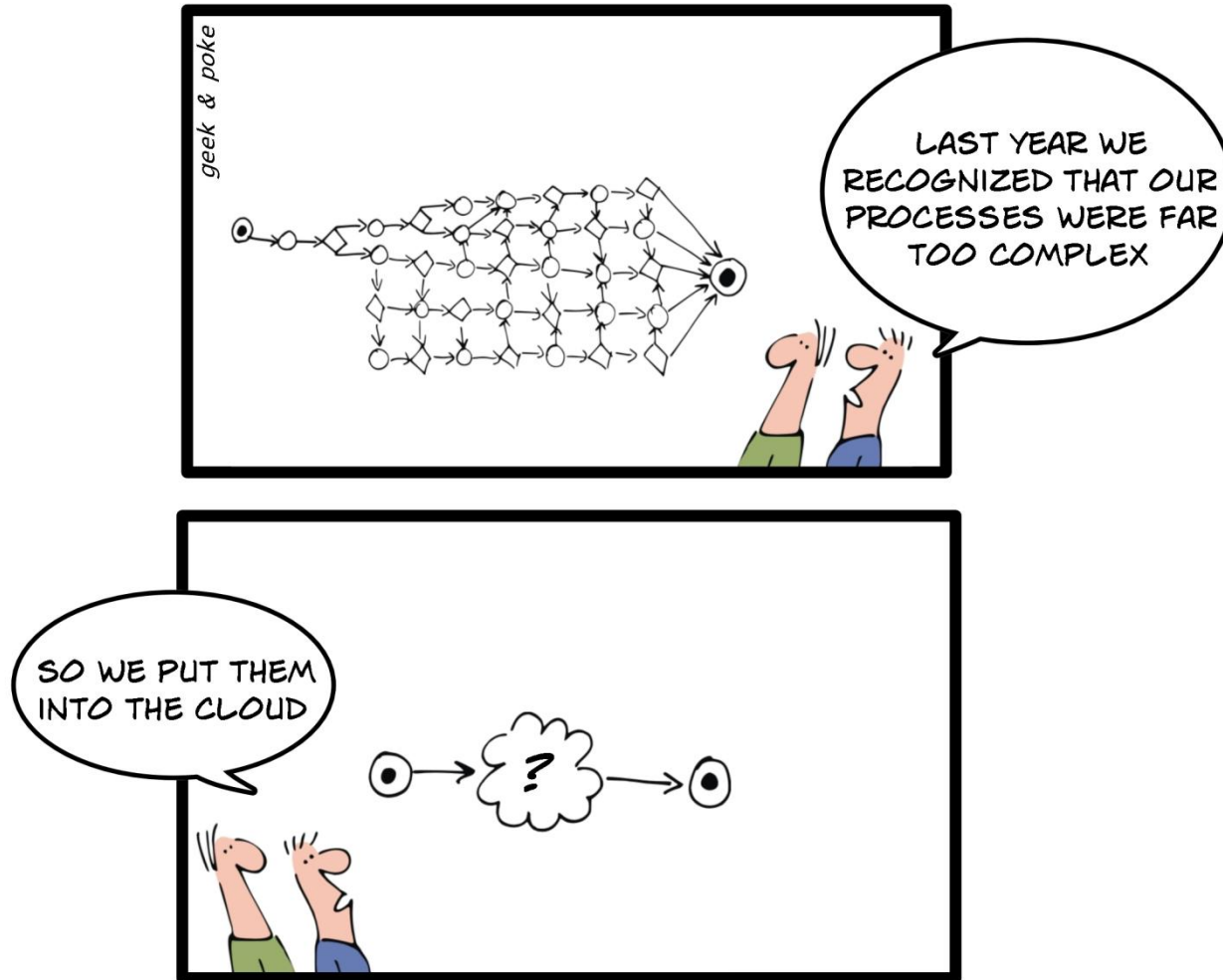
2010-04-20

# CLOUD IS THE NEW BLACK



GARTNER  
HYPE CYCLE

# Molnet inger stora förhoppningar



LET THE CLOUDS MAKE YOUR LIFE EASIER

<http://geekandpoke.typepad.com/.a/6a00d8341d3df553ef01156f3f1664970b-pi>

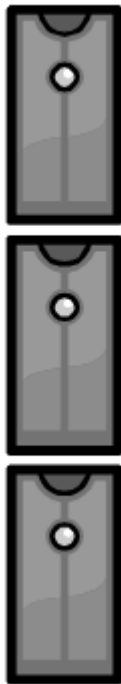
# Det bästa och det värsta med molnet



# Elasticitet



...många servrar

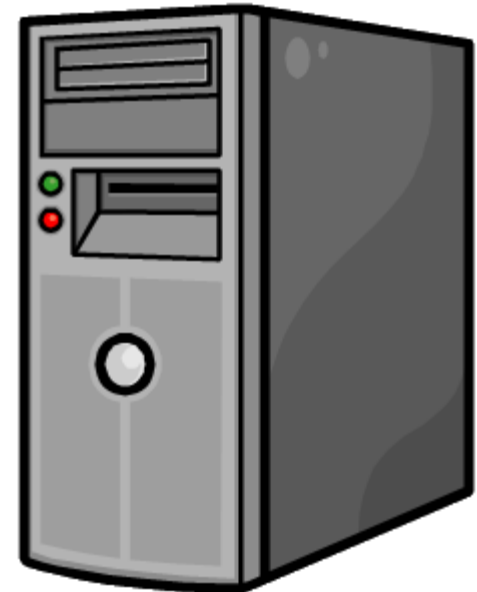


Skala ut

Från en till



...en kraftigare server

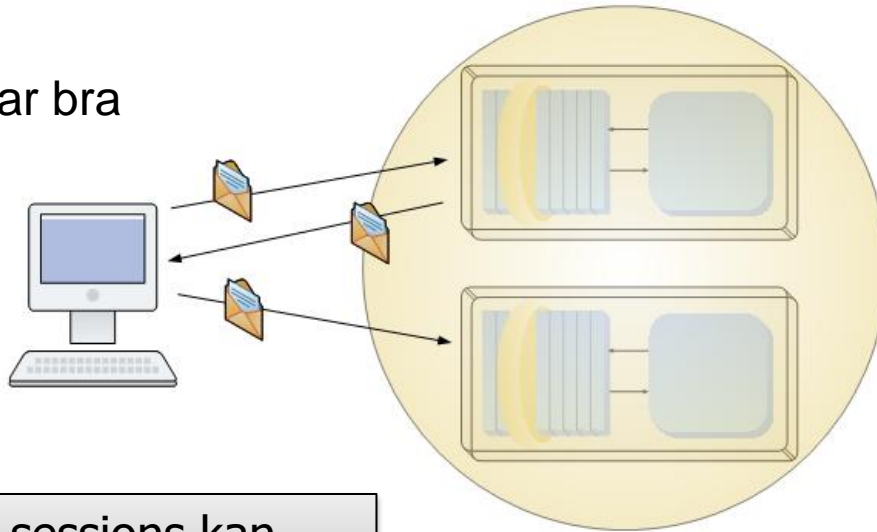


Skala upp

Viktigt:  
Elasticiteten i molnet  
kräver många instanser

# Tillstånd (state) och många instanser

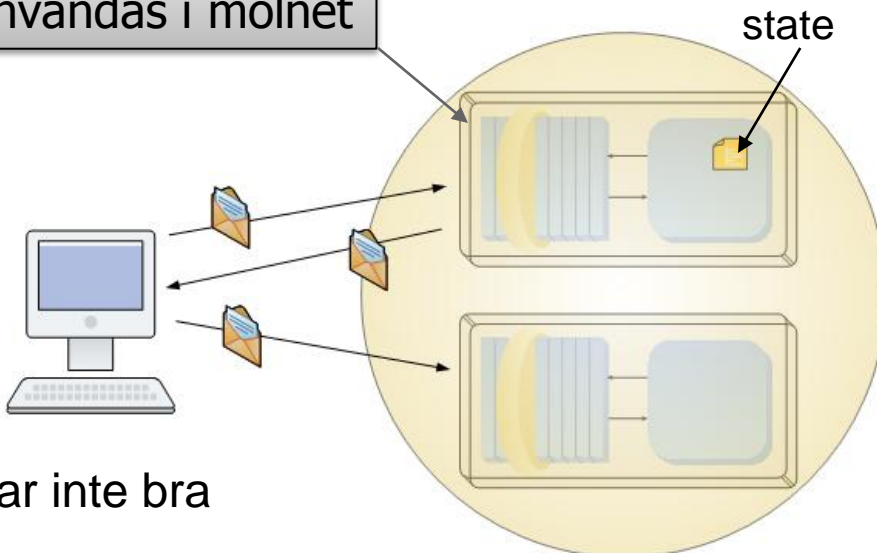
Skalar bra



Stateless:

Nya anrop behöver inte känna till gamla anrop

Sticky sessions kan inte användas i molnet

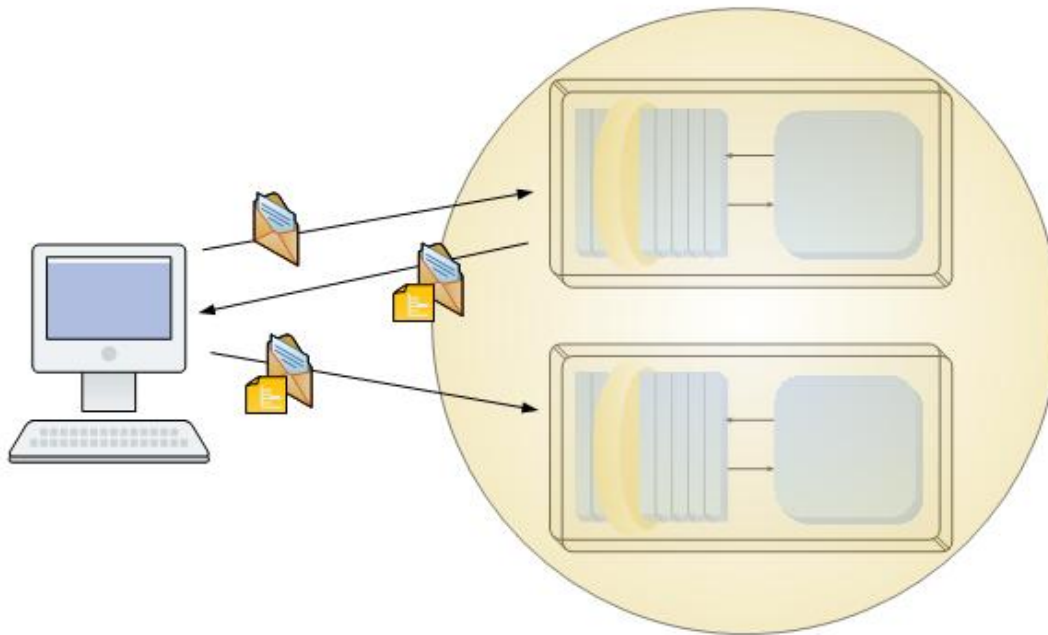


Stateful:

Tjänsten måste känna till en hel konversation

Skalar inte bra

# State Messaging (1 av 2)



- Lägg tillståndet i meddelandet
- Skalar lika bra som stateless
- Problem:
  - mycket tillstånd ger stora meddelanden = påverkar prestandan
  - kan inte dela tillstånd mellan användare
  - kan inte dela tillstånd mellan olika tjänster
  - jobbigt att skydda känslig information i meddelanden



# State Messaging (2 av 2)

Varför är det jobbigt att skydda tillstånd med känslig information i meddelanden?

Från boken "SOA with .Net & Windows Azure"

No security	Signed headers	Signed headers & body	Signed headers & signed + encrypted body
928 characters	3563 characters	3830 characters	6860 characters

☀ Större meddelanden

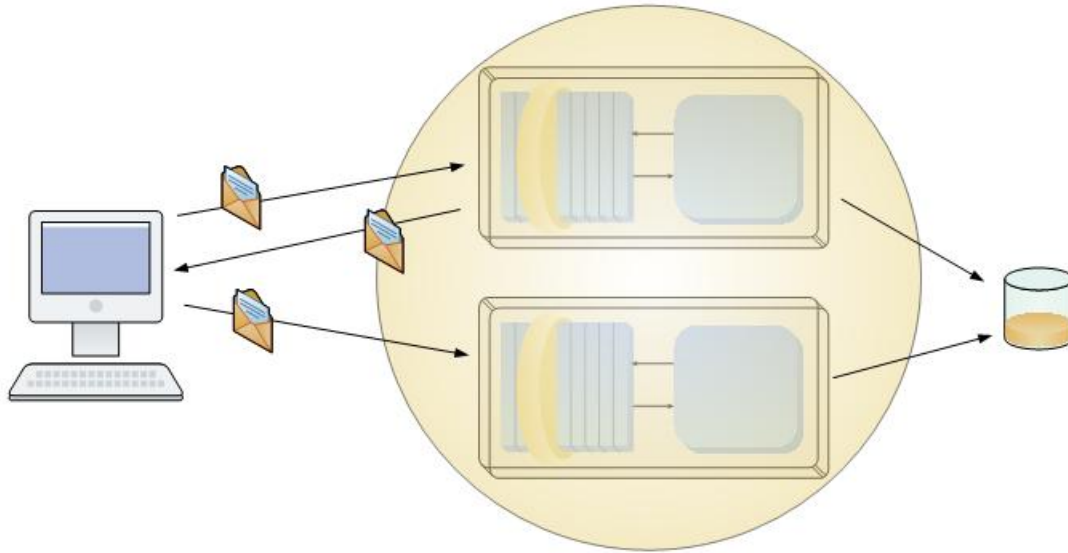
☀ Processorkraft för kryptering / dekryptering

☀ Certifikathantering

OBS: SSL skyddar inte hela vägen

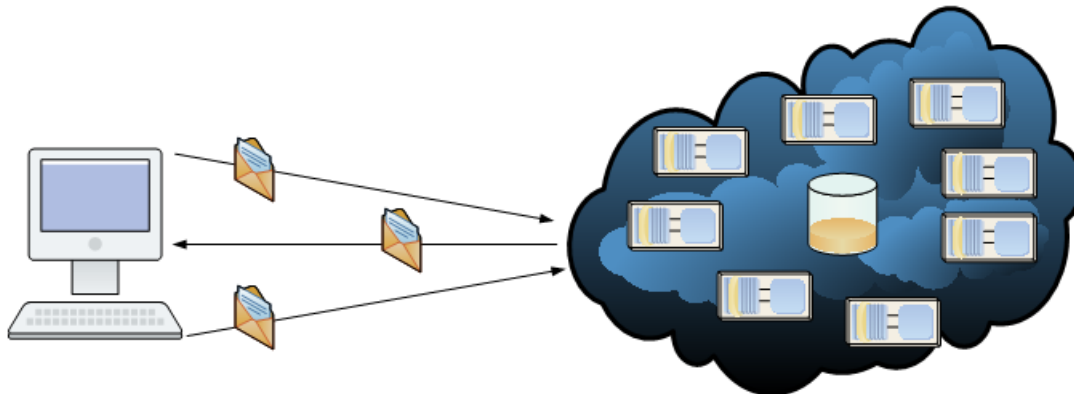


# State Repository (1 av 2)



Fungerar bra för

- delat tillstånd
- stora mängder tillstånd
- tillstånd med känslig information



Skalbarhetsproblem i molnet

- single point-of-failure
- STOR flaskhals



# State Repository (2 av 2)

Flera datakällor för tillstånd:

- ingen single-point-of-failure
- datakällorna måste synkroniseras



CAP (Consistency-Availability-Partition tolerance teoremet) ställer till det för oss!  
Vi måste välja mellan konsistens och tillgänglighet, exempel:

Majoriteten bestämmer:

- skriv till 2 av 3
- läs från 2 av 3
- läs från den 3:e vid behov

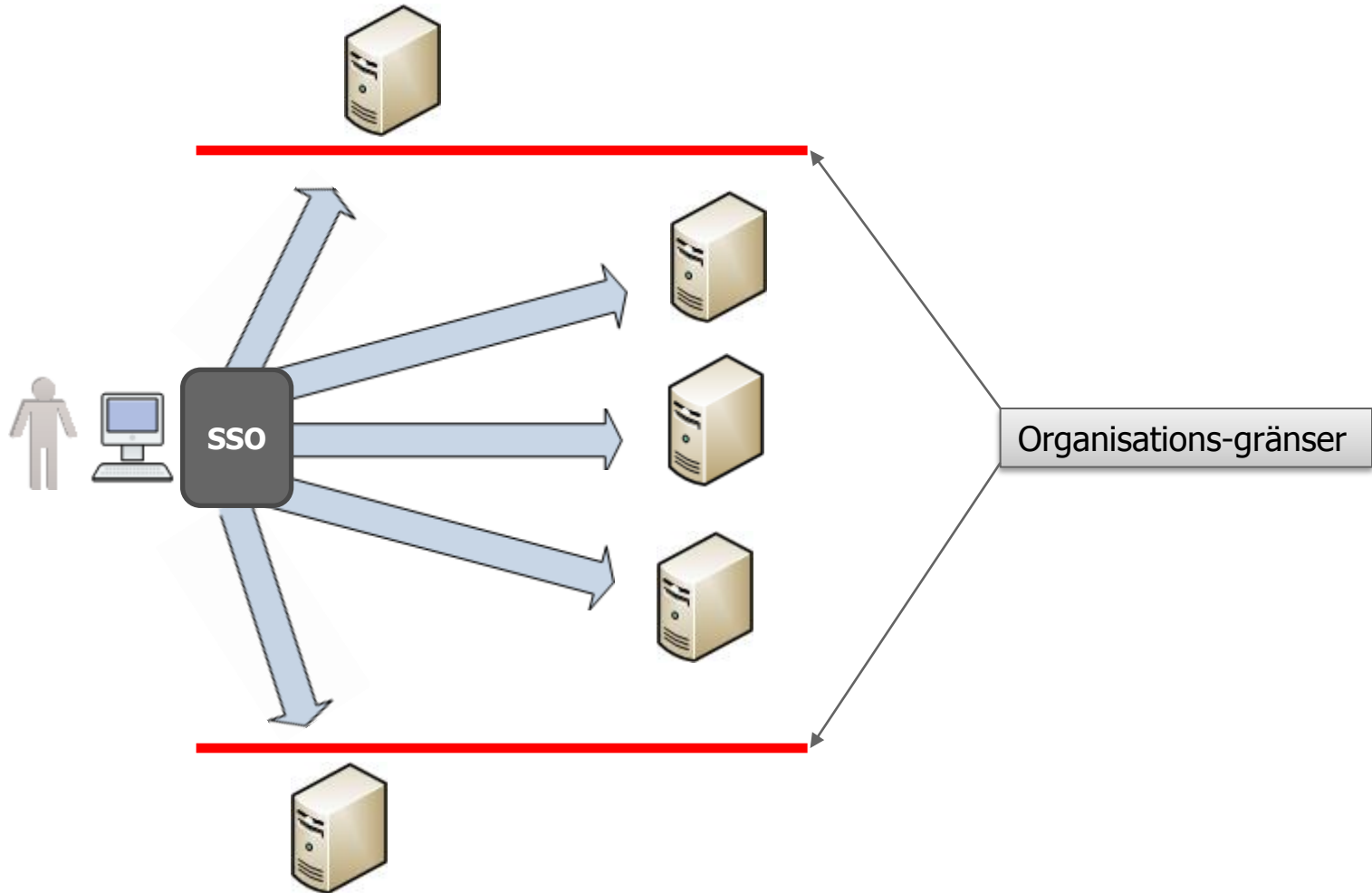
CP

Oberoende datakällor:

- skriv till 1 av 3
- läs från 1 av 3
- events / bäst före + sammanfoga

AP

En viktig säkerhetsaspekt:  
**Åtkomstkontroll**



SSO =  
Single Sign On

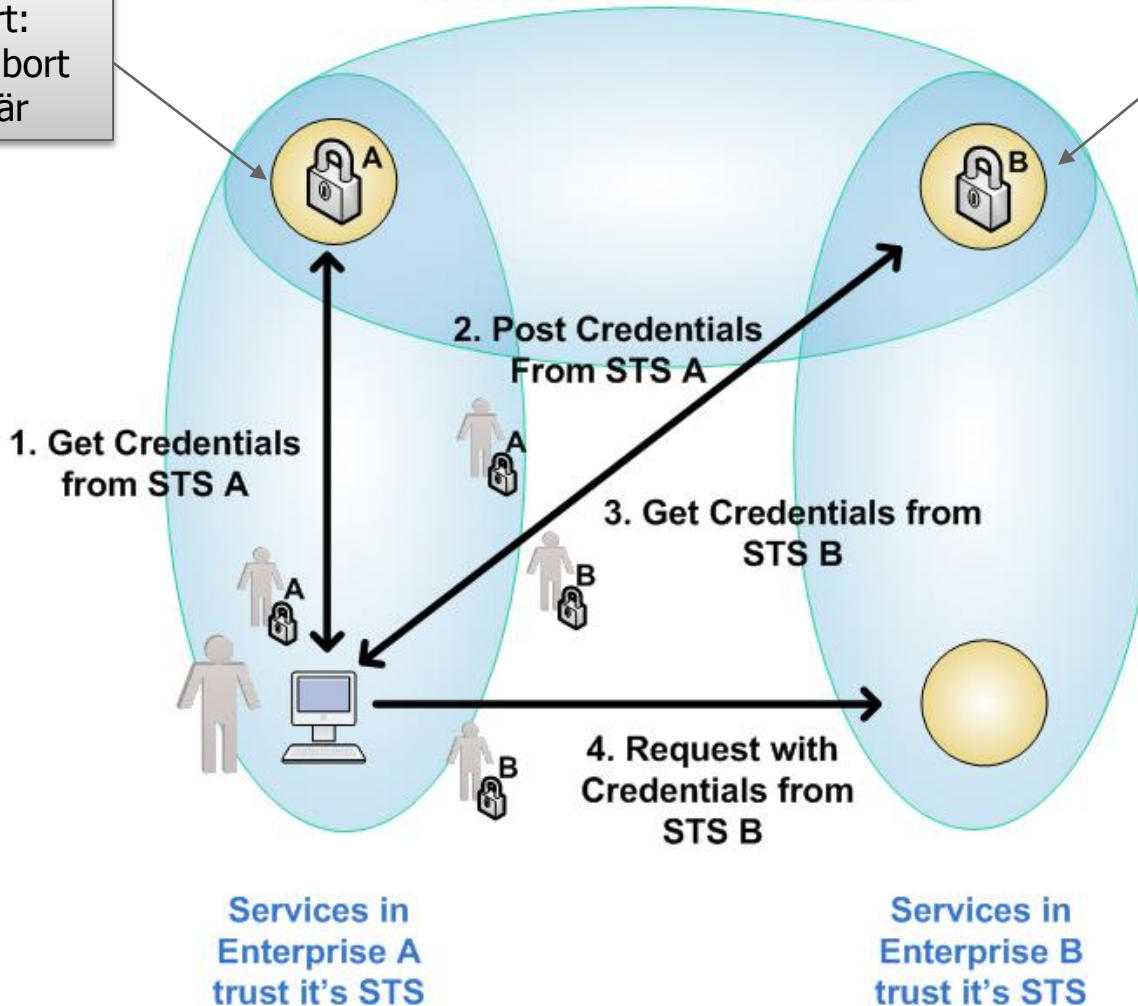


# Federated Identity (1 of 3)

The STS of Enterprise B trusts the STS of Enterprise A

Mycket säkert:  
Lägg till / ta bort  
rättigheter här

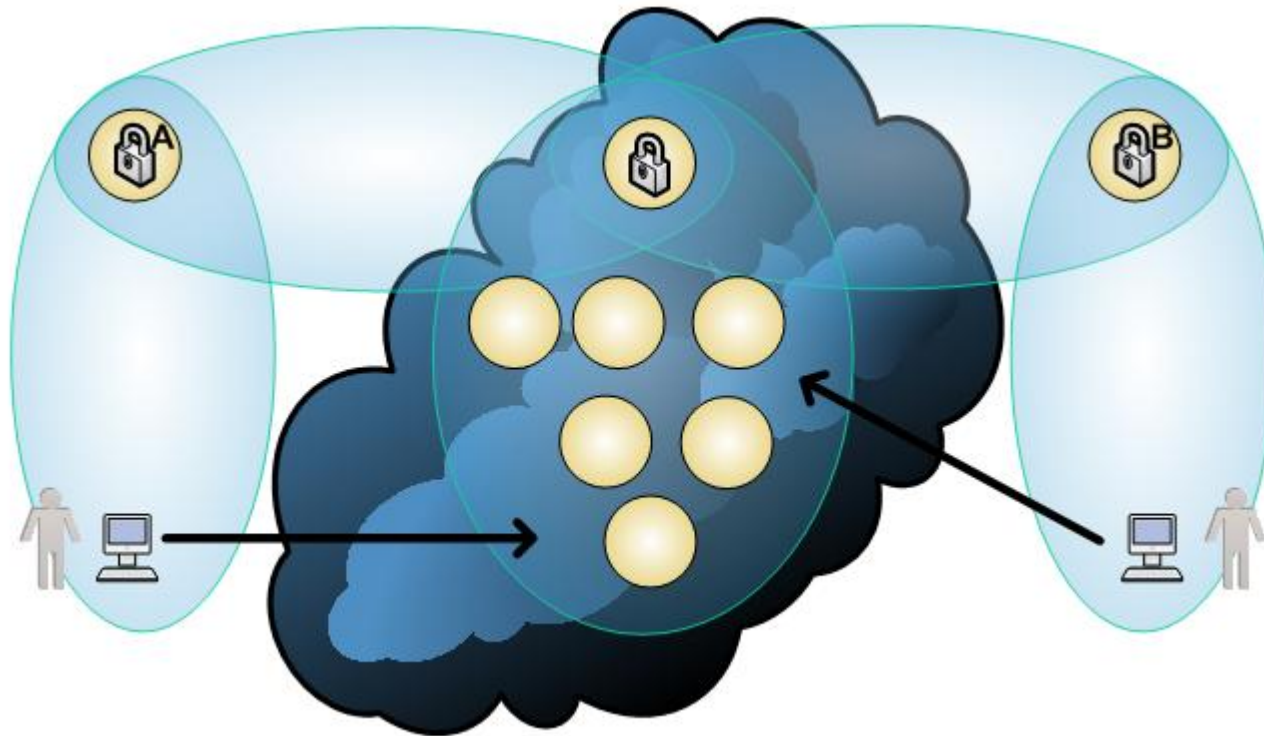
Mycket säkert:  
Behöver inte  
lägga till / ta bort  
rättigheter här



STS =  
Security Token Service



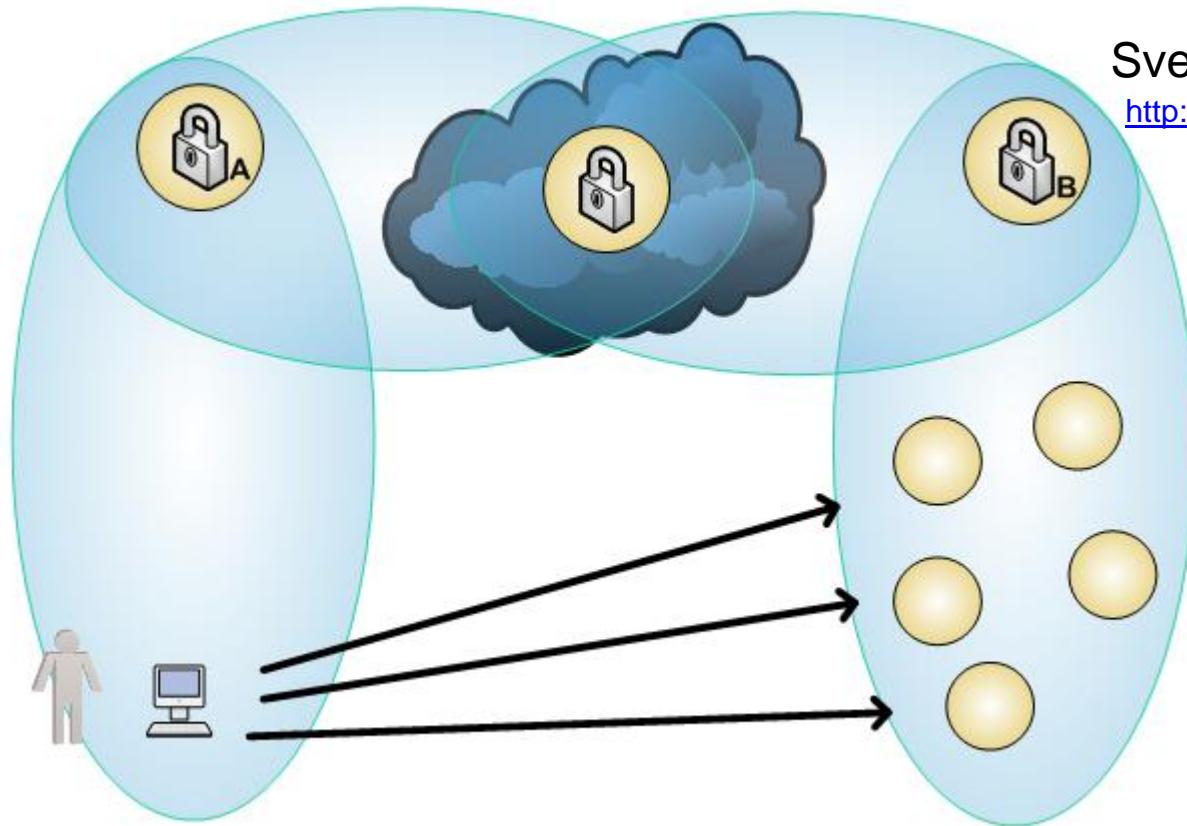
# Federated Identity (2 of 3)



Federated Identity passar mycket bra i molnet!



# Federated Identity (3 of 3)



Svensk sjukvård BIF och NPÖ  
<http://itivarden.idg.se/2.2898/1.211880>

Standarder:

- SAML
- XACML

En tredjeparts-STS i molnet kan knyta ihop bolag

Säkerhet:  
**Skillnaden mellan hosting och molnet**



Traditionell hosting

Vilka lagar gäller?



Hosting i molnet (stora leverantörer)

<http://www.idg.se/2.1085/1.308868/universitet-jo-anmals-for-google-avtal>

# Frågor?



**forefront**  
consulting group

Mer information:

<http://soapatterns.com>

<http://soapatterns.org>

<http://soaschool.se>

<http://cloudsweden.wordpress.com/>



Herbjörn Wilhelmsen

[herbjorn@wilhelmsen.se](mailto:herbjorn@wilhelmsen.se)

[http://twitter.com/herbjorn\\_w](http://twitter.com/herbjorn_w)

<http://se.linkedin.com/in/herbjorn>

# UTLOTTNING!

