Försvar mot nätverksintrång

Kim Hammar

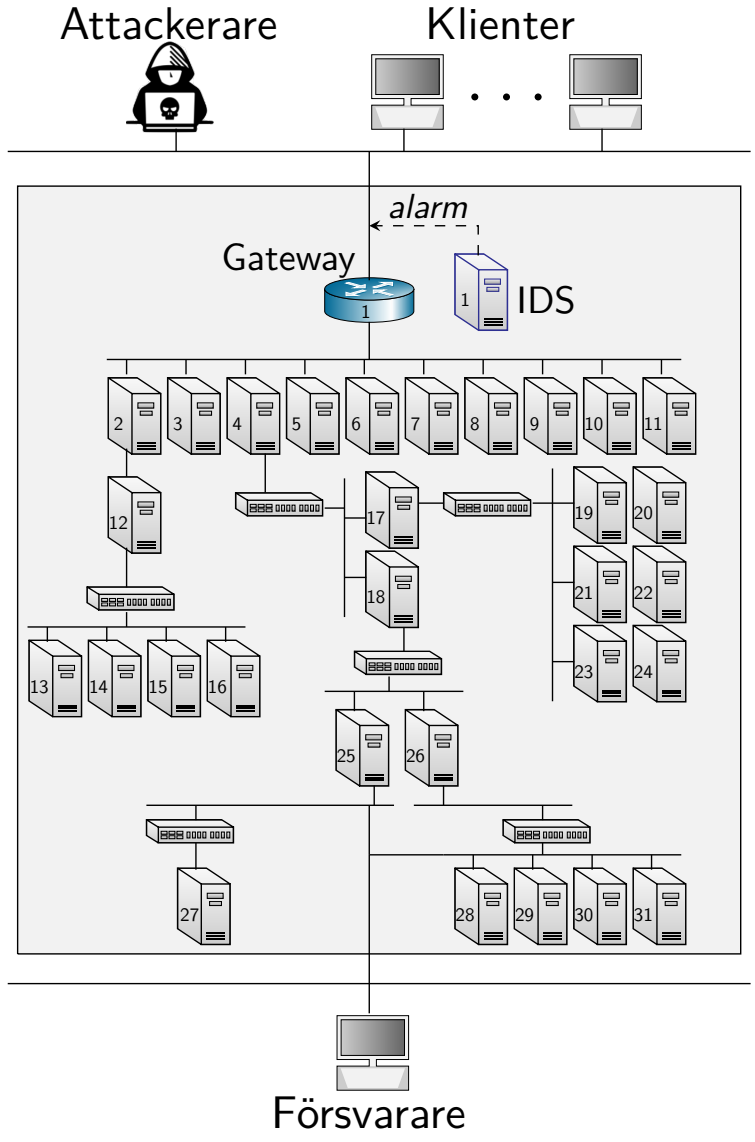
kimham@kth.se

CDIS, Centrum för cyberförsvar och informationssäkerhet
NSE, Avdelningen för nätverk och systemteknik
KTH Kungliga Tekniska Högskolan

16 Feb, 2022

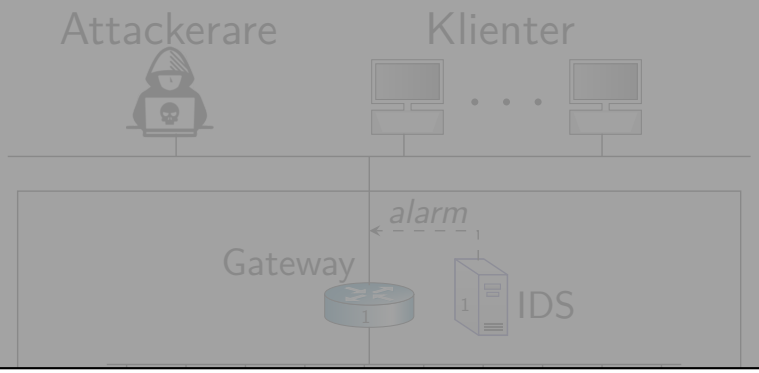
Ett nätverksintrång scenario

- En **försvarare** administrerar en IT-infrastruktur.
  - IT-infrastrukturen består av komponenter i ett datornät.
  - Komponenterna erbjuder nätverkstjänster.
  - Försvararen skyddar infrastrukturen genom nätverksövervakning och aktivt försvar
- En **attackerare** har som mål att göra ett intrång på infrastrukturen.
  - Har partiell information om infrastrukturen.
  - Vill hacka specifika komponenter.
  - Attackerar genom rekognisering och exploatering.



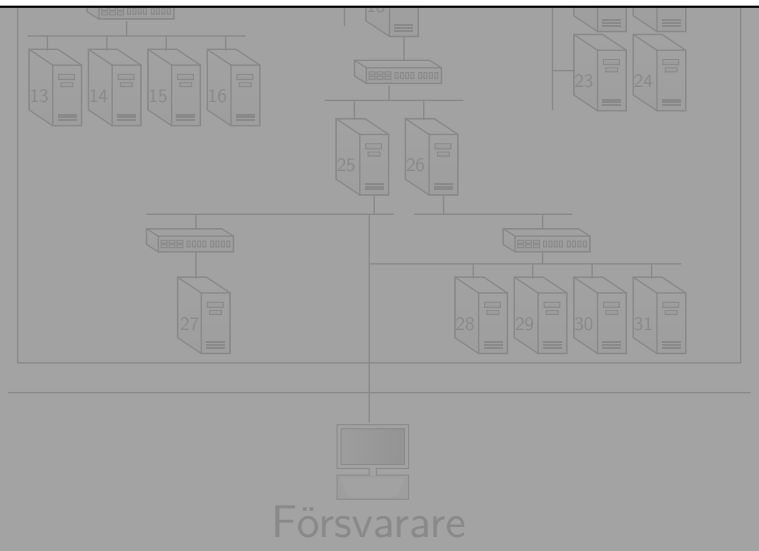
Ett nätverksintrång scenario

- En **försvarare** administrerar en IT-infrastruktur.
  - IT-infrastrukturen består av komponenter i ett datornät.
  - Komponenterna erbjuder nätverkstjänster.

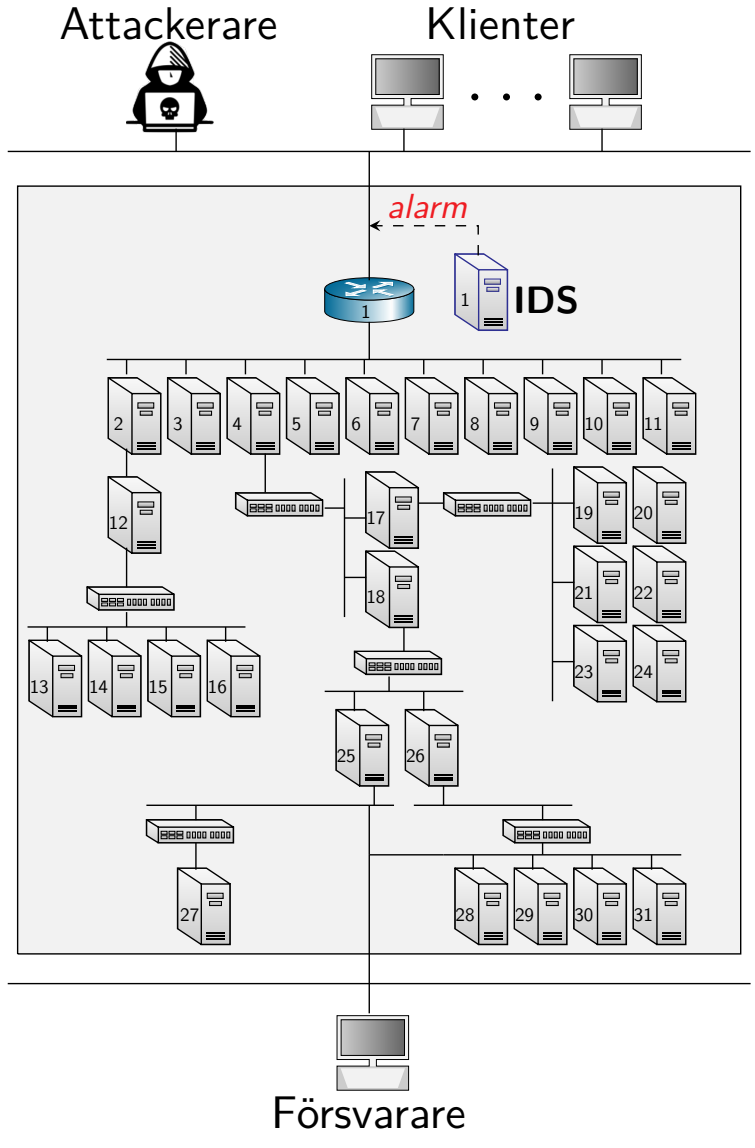


Hur upptäcker vi attacker? Hur åtgärdar vi dem?

- En **attackerare** har som mål att göra ett intrång på infrastrukturen.
  - Har partiell information om infrastrukturen.
  - Vill hacka specifika komponenter.
  - Attackerar genom rekognisering och exploatering.

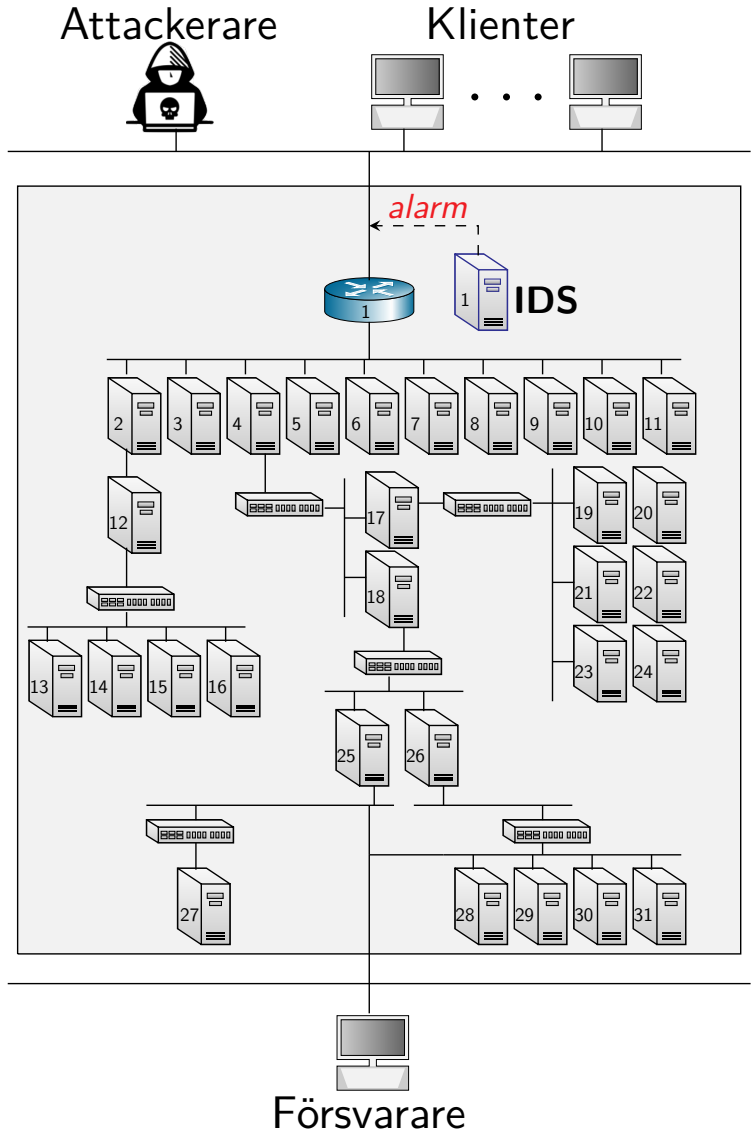


Intrångsdetektering: hur upptäcker vi en attackerare?



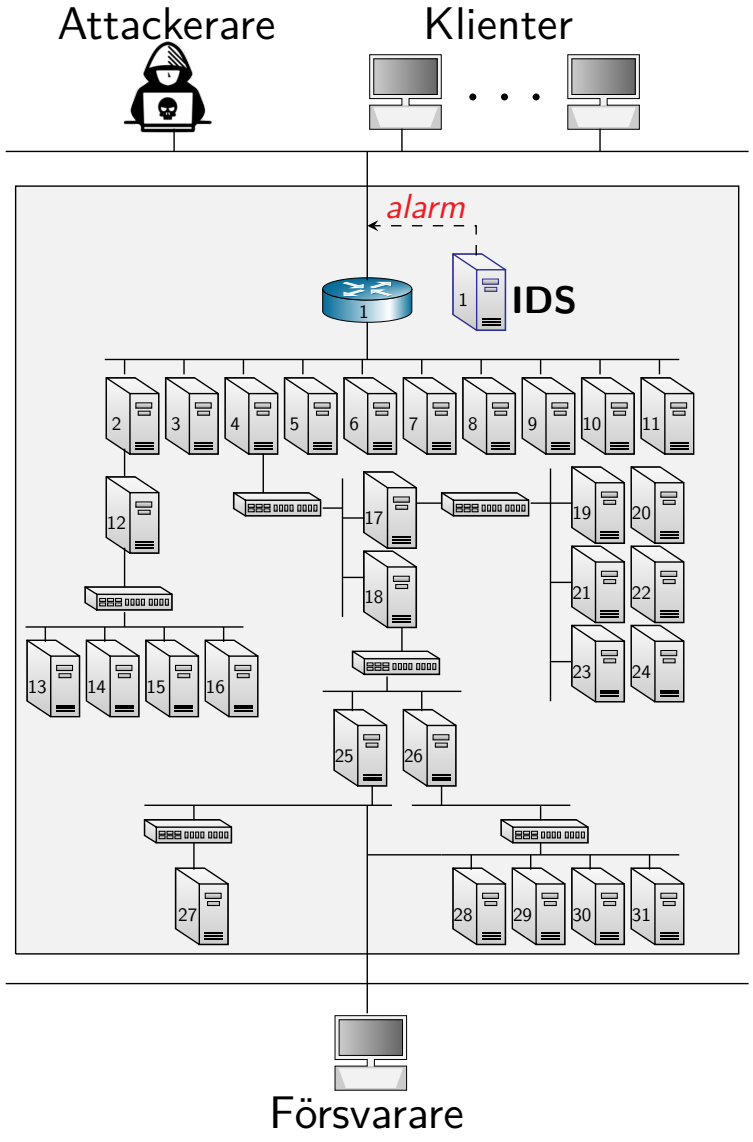
Intrångsdetektering: hur upptäcker vi en attackerare?

- **En typ av detektering:** detektera attacksignaturer



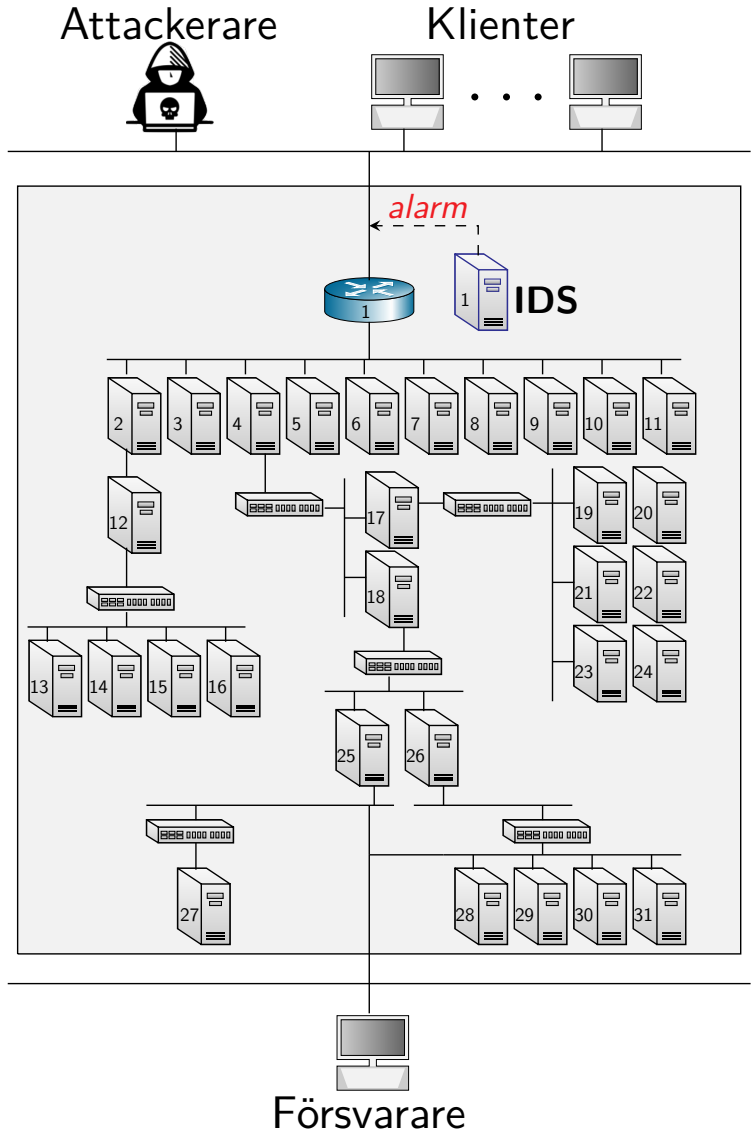
Intrångsdetektering: hur upptäcker vi en attackerare?

- **En typ av detektering:** detektera attacksignaturer
- **Exempel på signaturdetektering:**
  - *Om det är utgående nätverkstrafik på TCP-port 2589 så är det en signature på ett virus vid namn "dagger".*



Intrångsdetektering: hur upptäcker vi en attackerare?

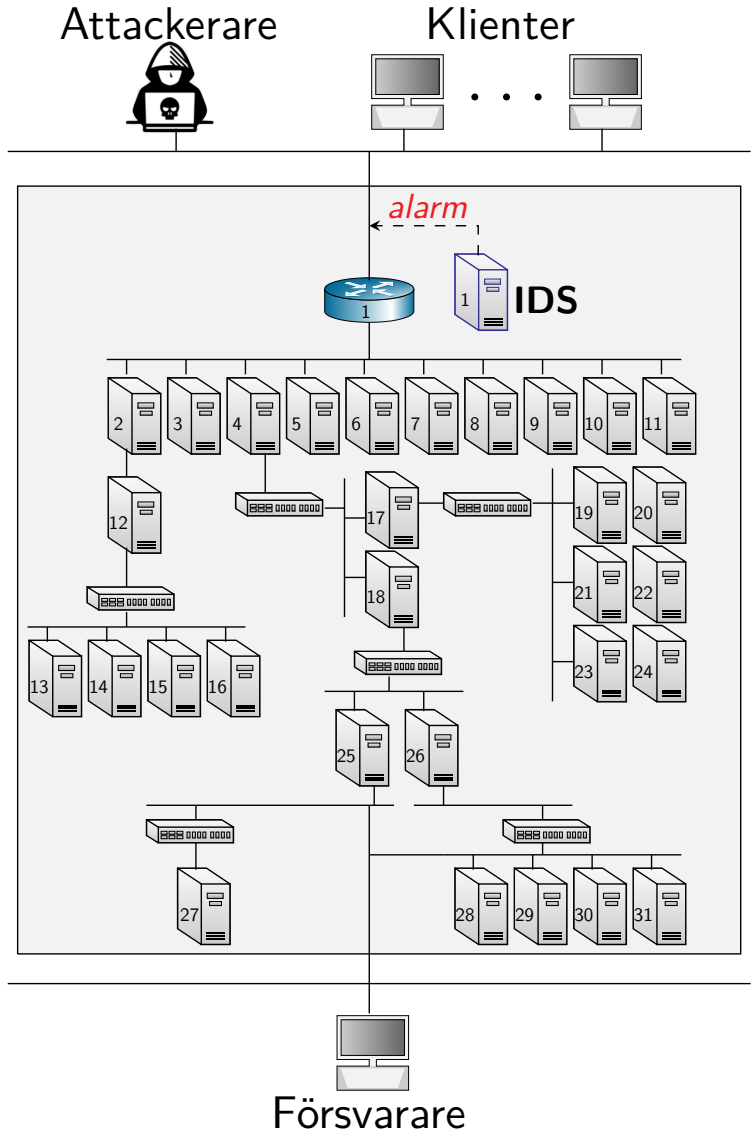
- **En typ av detektering:** detektera attacksignaturer
- **Exempel på signaturdetektering:**
  - *Om det är utgående nätverkstrafik på TCP-port 2589 så är det en signature på ett virus vid namn "dagger".*
- **Alternativ typ av detektering:** detektera anomalier



Intrångsdetektering: hur upptäcker vi en attackerare?

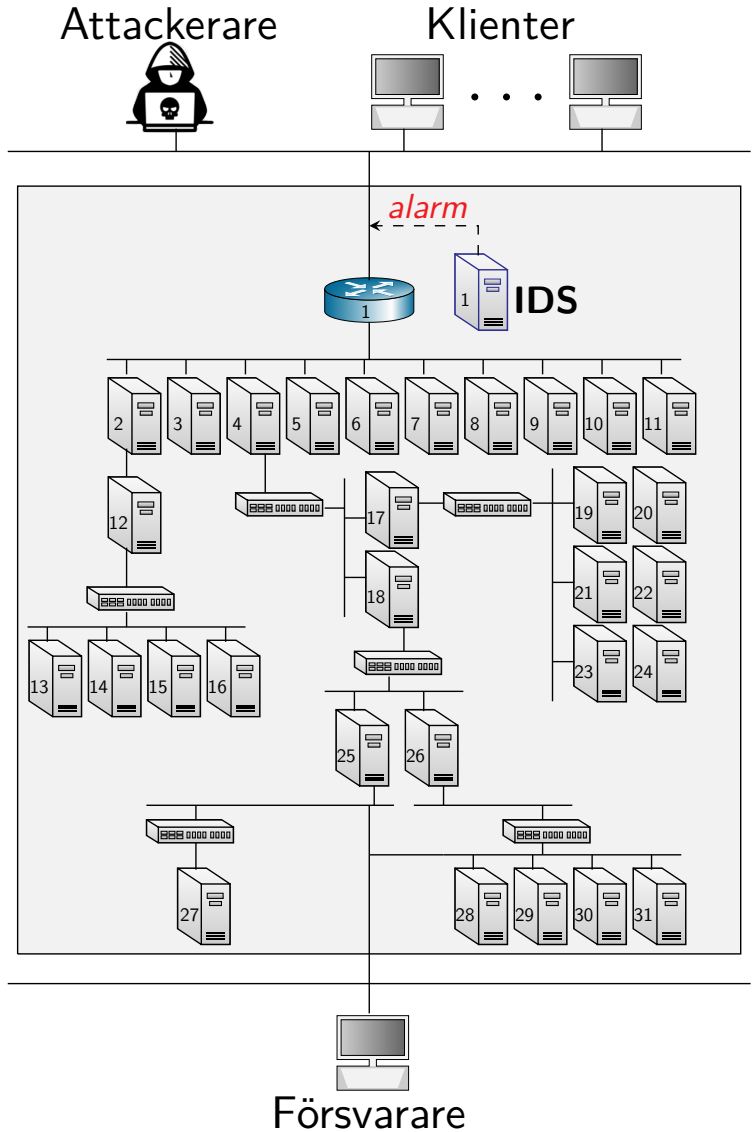
- **En typ av detektering:** detektera attacksignaturer
- **Exempel på signaturdetektering:**
  - *Om det är utgående nätverkstrafik på TCP-port 2589 så är det en signature på ett virus vid namn "dagger".*

- **Alternativ typ av detektering:** detektera anomalier
- **Exempel på anomalidetektering:**
  - *Om p-värdet för att trafiken genererades av en klient är mindre än 0.05*



Intrångsdetektering: hur upptäcker vi en attackerare?

- **En typ av detektering:** detektera attacksignaturer
- **Exempel på signaturdetektering:**
  - *Om det är utgående nätverkstrafik på TCP-port 2589 så är det en signature på ett virus vid namn "dagger".*
- **Alternativ typ av detektering:** detektera anomalier
- **Exempel på anomalidetektering:**
  - *Om p-värdet för att trafiken genererades av en klient är mindre än 0.05*
- **Utmaning:**
  - Vill inte riskera att missa attacker, vilket leder till ett stort antal alarm varav många är falska alarm.



Problemet med Informationsöverflöd

Dashboard Tools
Add New Dashboard
Manage My Dashboards
Deploy Dashboards
My Dashboards
Home Page
10000 ft view
Database Server Group
Demo Dash
Guages
Hostgroups
Localhost Health
London
Map & Latest Alerts
Minnesota
Networking Dashboard
Notifications
XI System Health
nagios.com
Add Dashlets
Available Dashlets
Manage Dashlets

Minemap

Hostgroup 'switches' Status Grid

Hosts	Services
192.168.5.41	
192.168.5.43xx	

Last Updated: 2017-10-05 16:48:41

Network Outages

Severity	Host	State	Duration	Hosts Affected	Services Affected
There are no blocking outages at this time.					

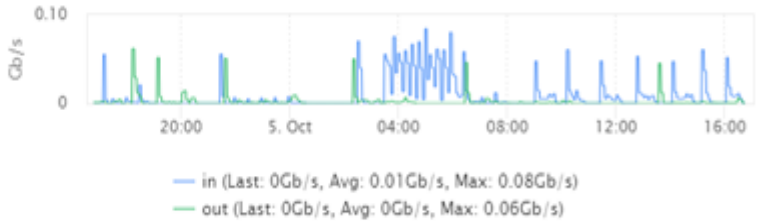
Last Updated: 2017-10-05 16:48:41

Switches (switches)

Host	Status	Services
192.168.5.41	Up	46 Ok 15 Critical
192.168.5.42	Down	No services found
192.168.5.43xx	Up	43 Ok 8 Critical
192.168.5.90	Up	No services found

Last Updated: 2017-10-05 16:48:41

192.168.5.41 : Port 1 Bandwidth



192.168.5.41 : Port 1 Bandwidth - Last 3 days

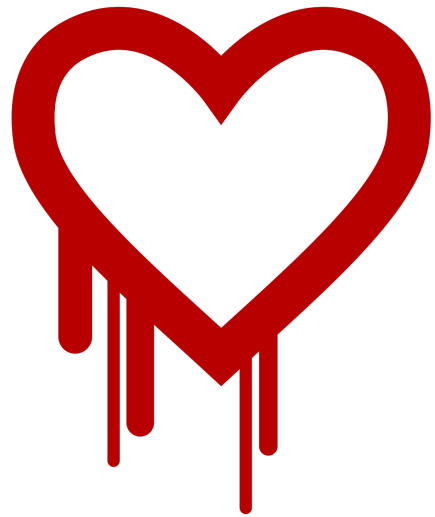
192.168.5.41 : Port 11 Bandwidth

192.168.5.41 : Port 3 Bandwidth

DEMO: Intrångsdetekteringsystem

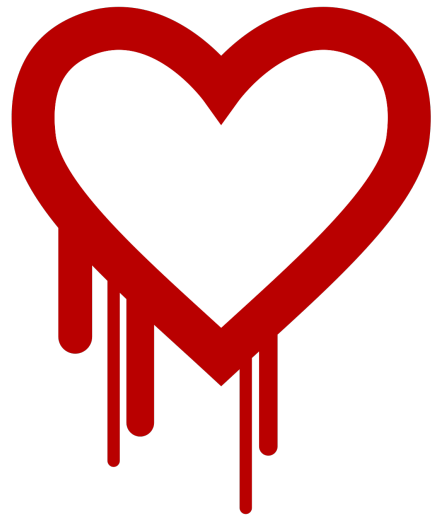
Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren



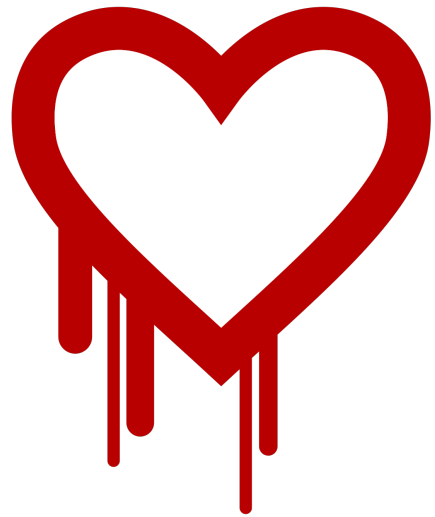
Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren



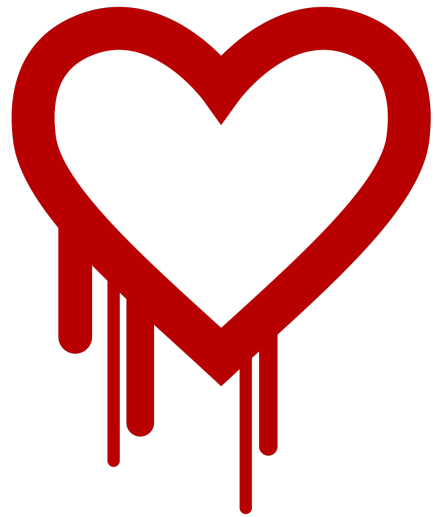
Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren



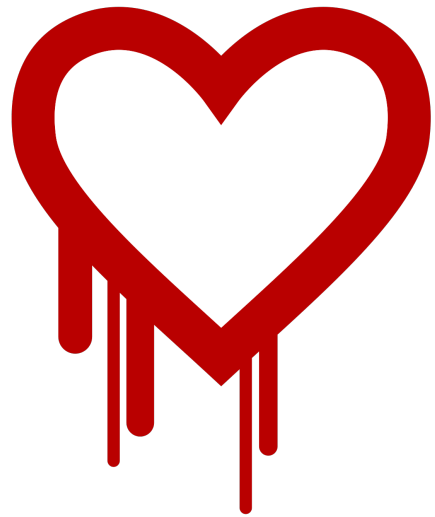
Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren



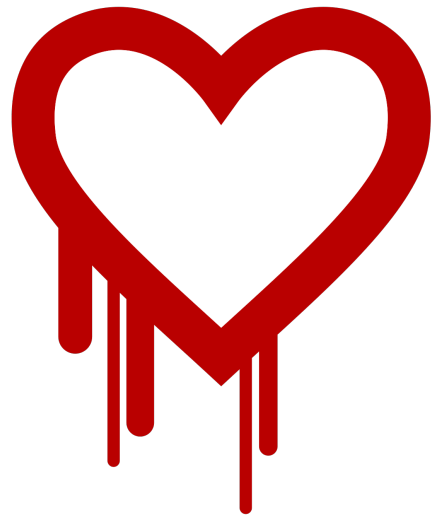
Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren



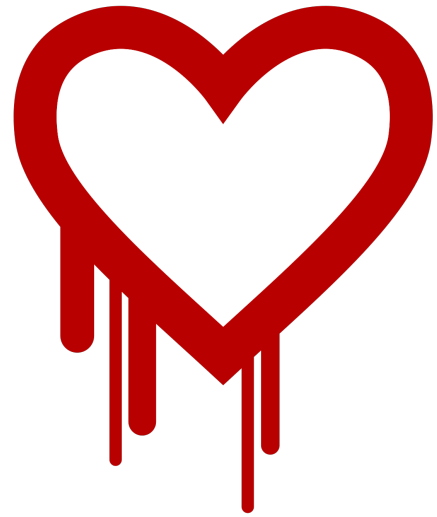
Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren

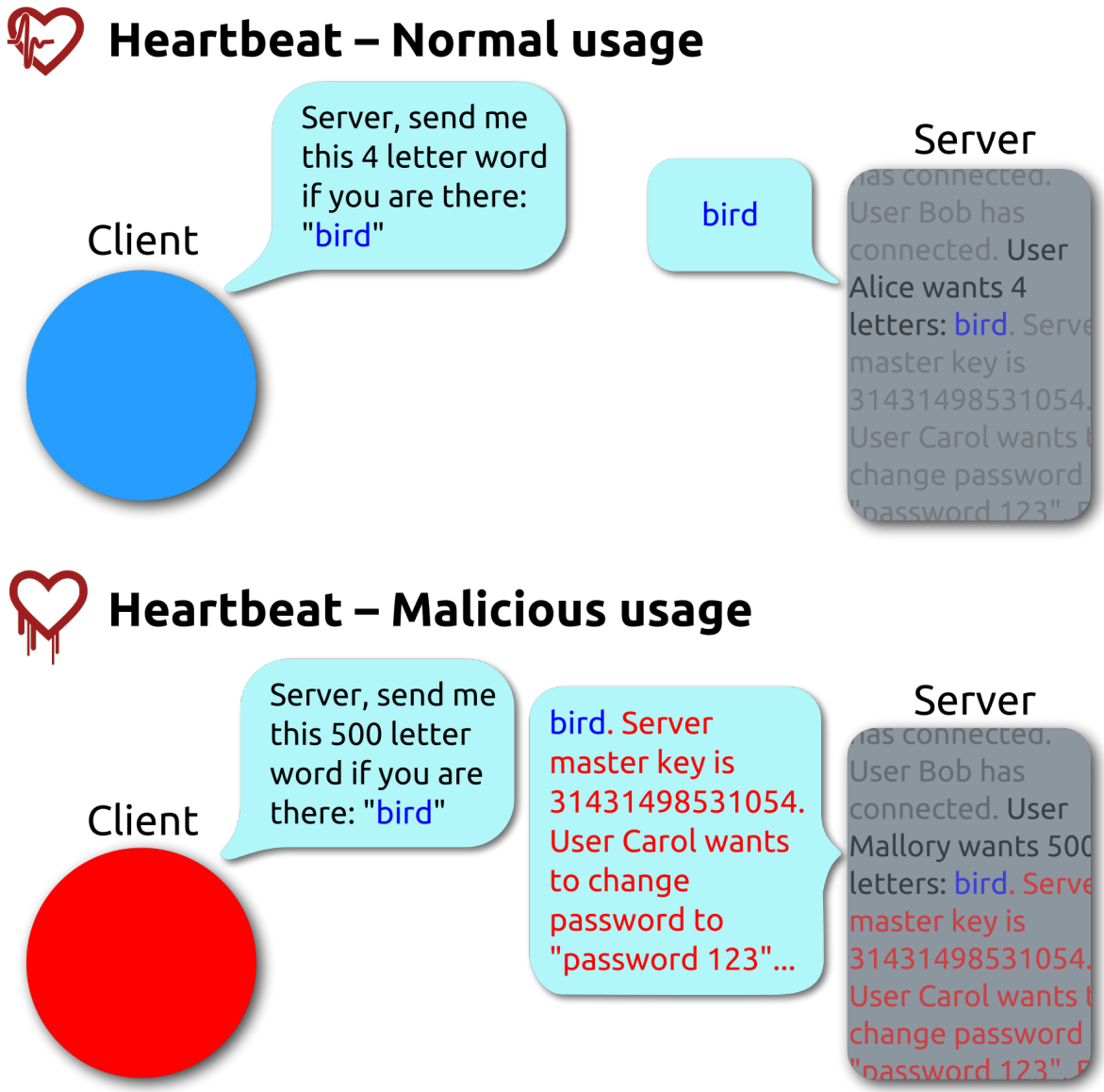


Exempelattack: Buffer-overread (Heartbleed)

- **En säkerhetsbugg i OpenSSL-biblioteket**
  - Buggen släpptes 2012
  - Upptäcktes 2014 (!)
- **Påverkad mjukvara:** de flesta implementationerna av TLS
- **Hur attacken fungerar:**
  - En sändare i OpenSSL kan skicka ett "heartbeat"-meddelande med data+längd
  - Mottagaren allokerar en minnesbuffer enligt den rapporterade längden utan att verifiera längden
  - Mottagaren skriver datan till buffern
  - Mottagaren skickar tillbaka innehållet i buffern till avsändaren
  - Eftersom bufferstorleken kan vara större en datan (det verifieras inte) så är det möjligt att mottagaren skickar tillbaka mer data än datan som skickades av sändaren - möjligtvis känslig data.



Exempelattack: Buffer-overread (Heartbleed)



Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

- Vi kan använda så kallad djuppaketinspektering “deep packet inspection”:
  1. Fånga nätverkstrafiken
  2. Filtrera paket som skickats med TLS-protokollet
  3. Filtrera Heartbeat meddelanden som skickats med TLS-protokollet
  4. Verifiera att längd=storlek på data i varje heartbeat meddelande
  5. Om heartbeat meddelanden skickats med längd $\neq$ datastorlek så är det ett försök till en attack!

Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

- Vi kan använda så kallad djuppaketinspektering “deep packet inspection”:
  1. Fånga nätverkstrafiken
  2. Filtrera paket som skickats med TLS-protokollet
  3. Filtrera Heartbeat meddelanden som skickats med TLS-protokollet
  4. Verifiera att längd=storlek på data i varje heartbeat meddelande
  5. Om heartbeat meddelanden skickats med längd $\neq$ datastorlek så är det ett försök till en attack!

Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

- Vi kan använda så kallad djuppaketinspektering “deep packet inspection”:
  1. Fånga nätverkstrafiken
  2. Filtrera paket som skickats med TLS-protokollet
  3. Filtrera Heartbeat meddelanden som skickats med TLS-protokollet
  4. Verifiera att längd=storlek på data i varje heartbeat meddelande
  5. Om heartbeat meddelanden skickats med längd $\neq$ datastorlek så är det ett försök till en attack!

Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

- Vi kan använda så kallad djuppaketinspektering “deep packet inspection”:
  1. Fånga nätverkstrafiken
  2. Filtrera paket som skickats med TLS-protokollet
  3. Filtrera Heartbeat meddelanden som skickats med TLS-protokollet
  4. Verifiera att längd=storlek på data i varje heartbeat meddelande
  5. Om heartbeat meddelanden skickats med längd $\neq$ datastorlek så är det ett försök till en attack!

Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

- Vi kan använda så kallad djuppaketinspektering “deep packet inspection”:
  1. Fånga nätverkstrafiken
  2. Filtrera paket som skickats med TLS-protokollet
  3. Filtrera Heartbeat meddelanden som skickats med TLS-protokollet
  4. Verifiera att längd=storlek på data i varje heartbeat meddelande
  5. Om heartbeat meddelanden skickats med längd $\neq$ datastorlek så är det ett försök till en attack!

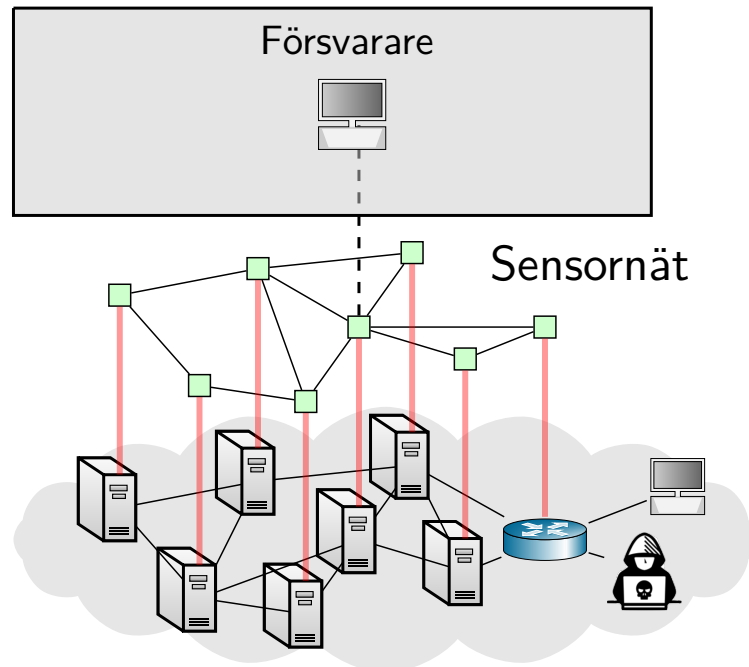
Hur kan vi upptäcka Heartbleedattacker via nätverksövervakning?

- Vi kan använda så kallad djuppaketinspektering “deep packet inspection”:
  1. Fånga nätverkstrafiken
  2. Filtrera paket som skickats med TLS-protokollet
  3. Filtrera Heartbeat meddelanden som skickats med TLS-protokollet
  4. Verifiera att längd=storlek på data i varje heartbeat meddelande
  5. Om heartbeat meddelanden skickats med längd $\neq$ datastorlek så är det ett försök till en attack!

DEMO: Detektering av en heartbleed attack

Försvar

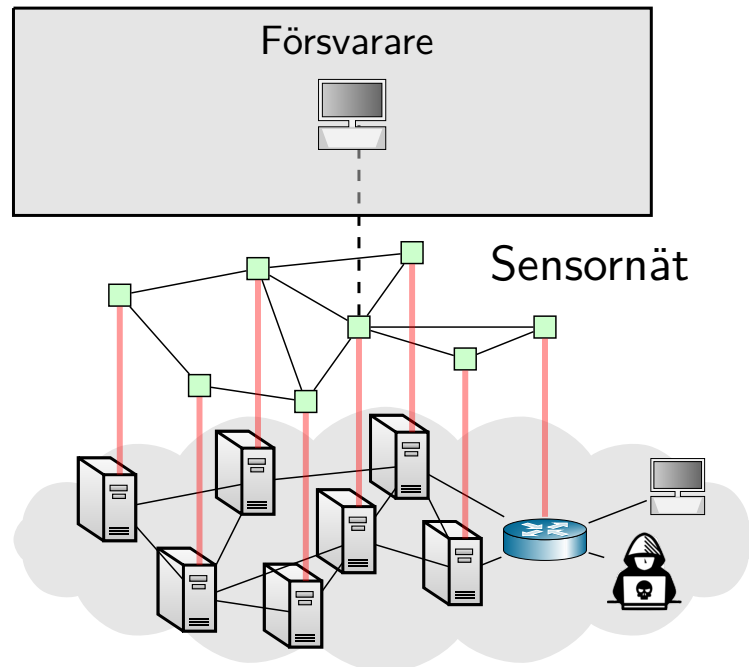
- **Upptäcka attacker:**
  - Metriska data: CPU-användning, bandbredd, processer, osv.
  - Loggfiler: IDS-logg, inloggningsförsök.
  - Paketinspektion (exempelvis med Wireshark)
- **Stoppa/förhindra attacker**:
  - Uppdatera brandväggen.
  - Återkalla certifikat/lösernord.
  - Uppdatera användarrättigheter.
  - Stäng av tjänster/komponenter.



Försvar

- **Upptäcka attacker:**
  - Metriska data: CPU-användning, bandbredd, processer, osv.
  - Loggfiler: IDS-logg, inloggningsförsök.
  - Paketinspektion (exempelvis med Wireshark)

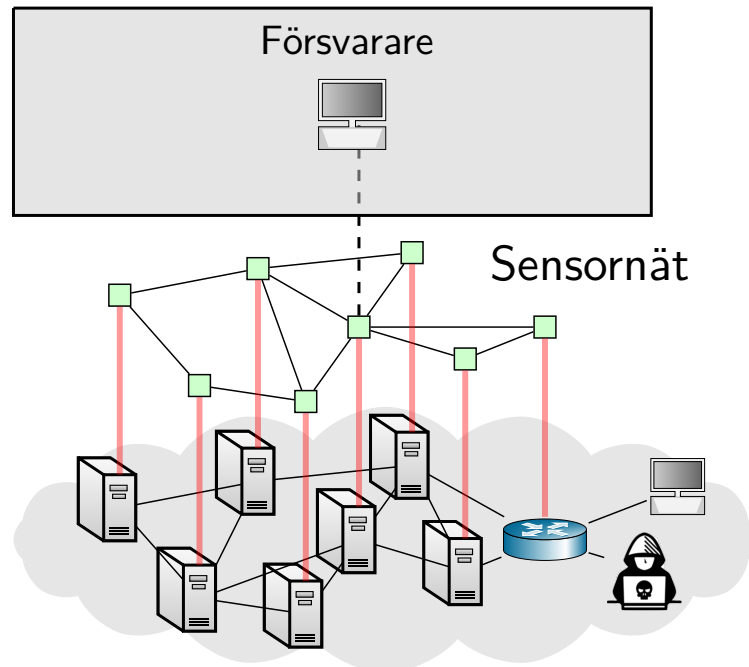
- **Stoppa/förhindra attacker**:
  - Uppdatera brandväggen.
  - Återkalla certifikat/lösernord.
  - Uppdatera användarrättigheter.
  - Stäng av tjänster/komponenter.



Försvar

- **Upptäcka attacker:**
  - Metriska data: CPU-användning, bandbredd, processer, osv.
  - Loggfiler: IDS-logg, inloggningsförsök.
  - Paketinspektion (exempelvis med Wireshark)

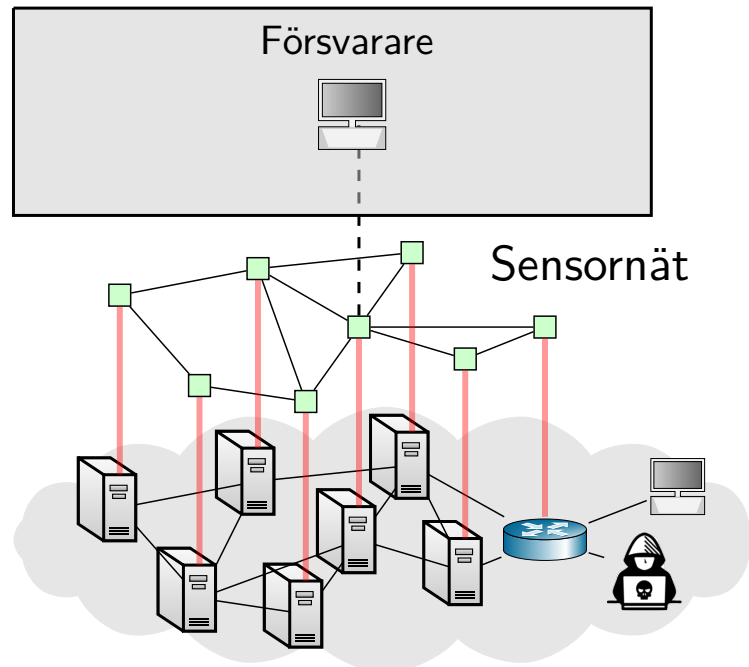
- **Stoppa/förhindra attacker**:
  - Uppdatera brandväggen.
  - Återkalla certifikat/lösernord.
  - Uppdatera användarrättigheter.
  - Stäng av tjänster/komponenter.



Försvar

- **Upptäcka attacker:**
  - Metriska data: CPU-användning, bandbredd, processer, osv.
  - Loggfiler: IDS-logg, inloggningsförsök.
  - Paketinspektion (exempelvis med Wireshark)

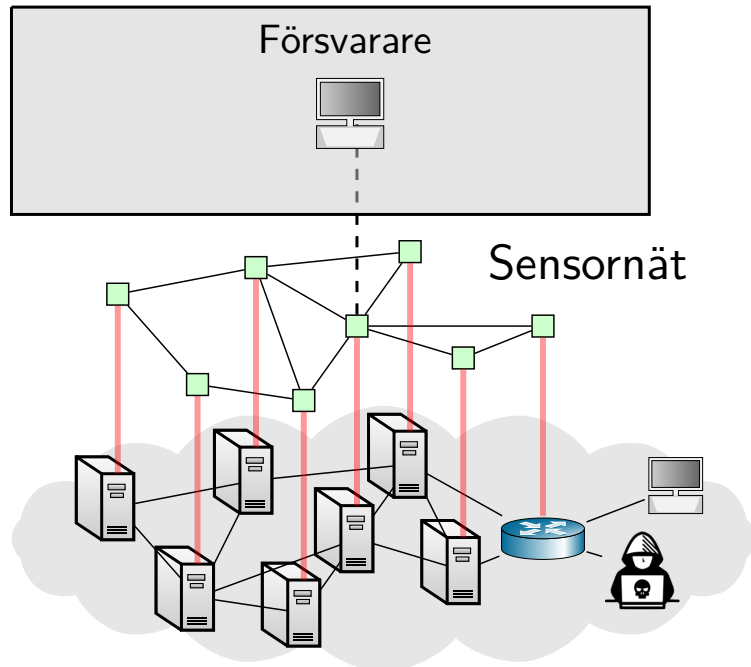
- **Stoppa/förhindra attacker**:
  - Uppdatera brandväggen.
  - Återkalla certifikat/lösernord.
  - Uppdatera användarrättigheter.
  - Stäng av tjänster/komponenter.

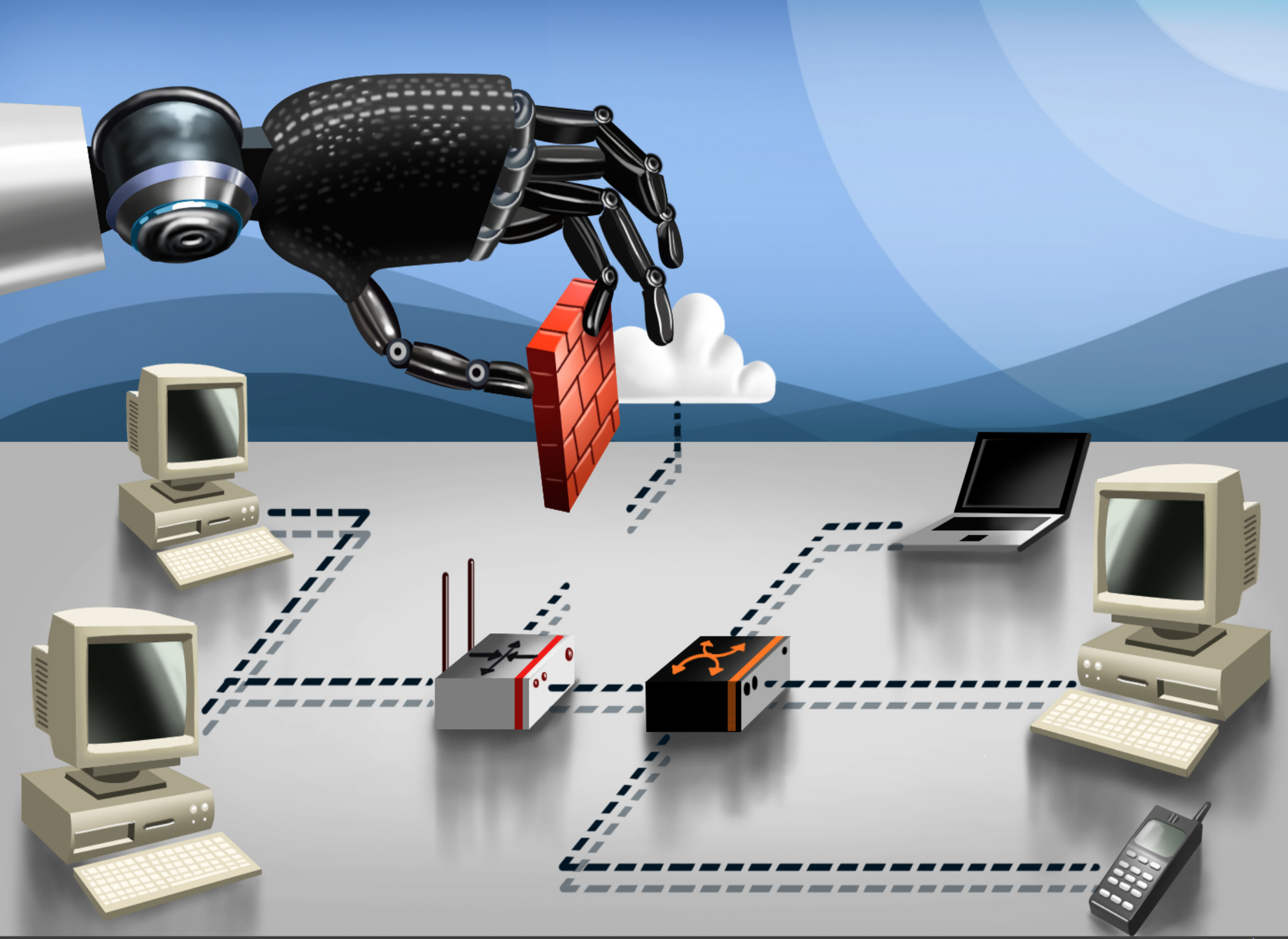


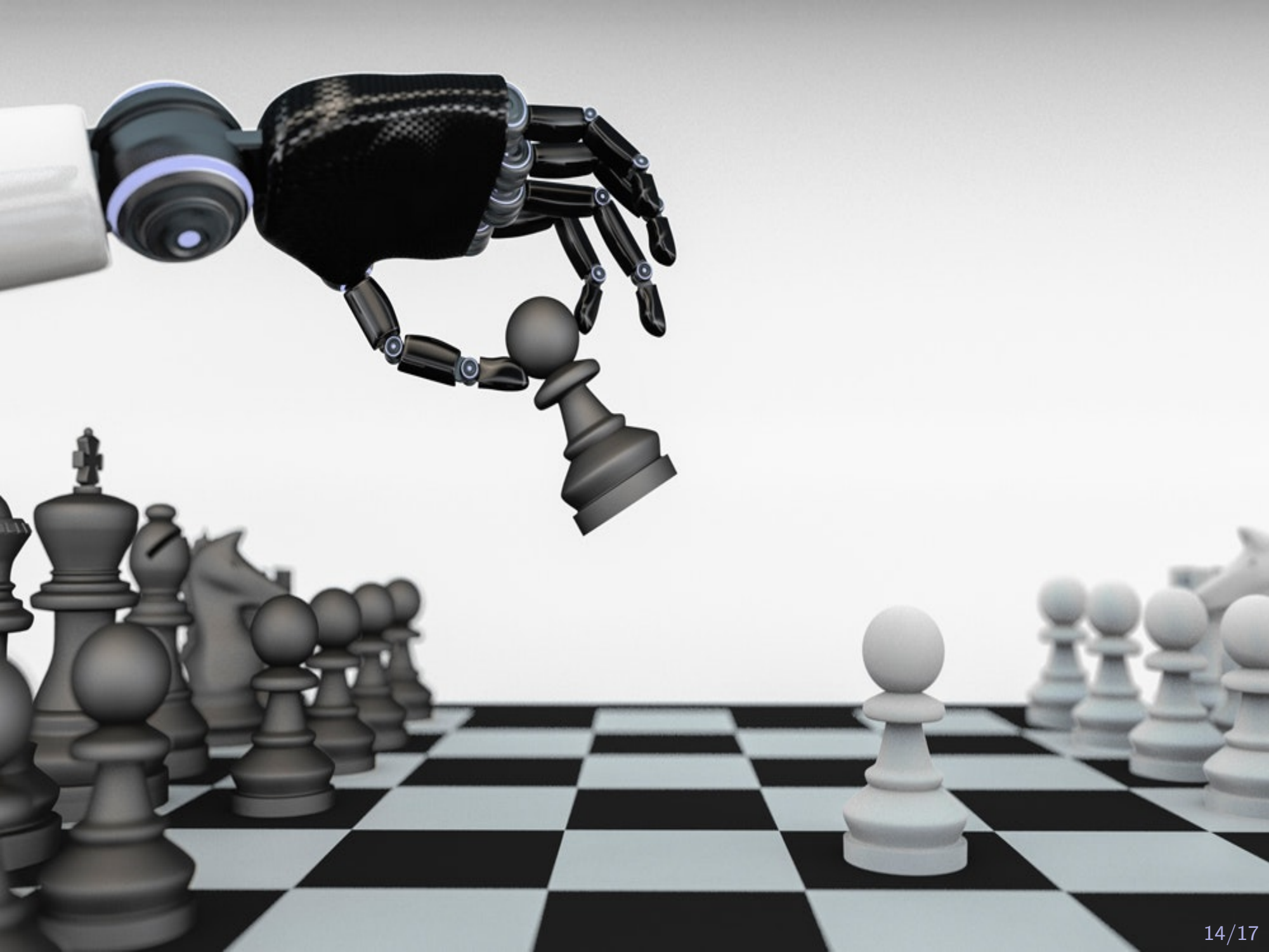
Försvar

- **Upptäcka attacker:**
  - Metriska data: CPU-användning, bandbredd, processer, osv.
  - Loggfiler: IDS-logg, inloggningsförsök.
  - Paketinspektion (exempelvis med Wireshark)

- **Stoppa/förhindra attacker**:
  - Uppdatera brandväggen.
  - Återkalla certifikat/lösernord.
  - Uppdatera användarrättigheter.
  - Stäng av tjänster/komponenter.

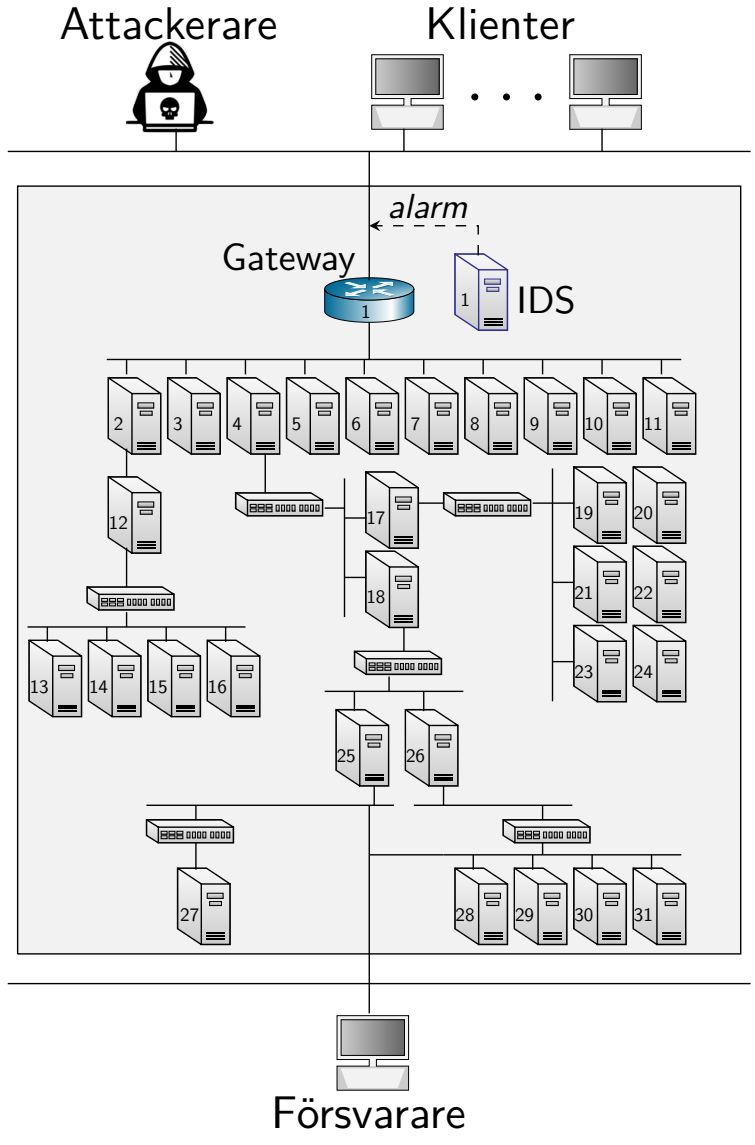






Utmaning: Automatiserade och föränderliga attackmetoder

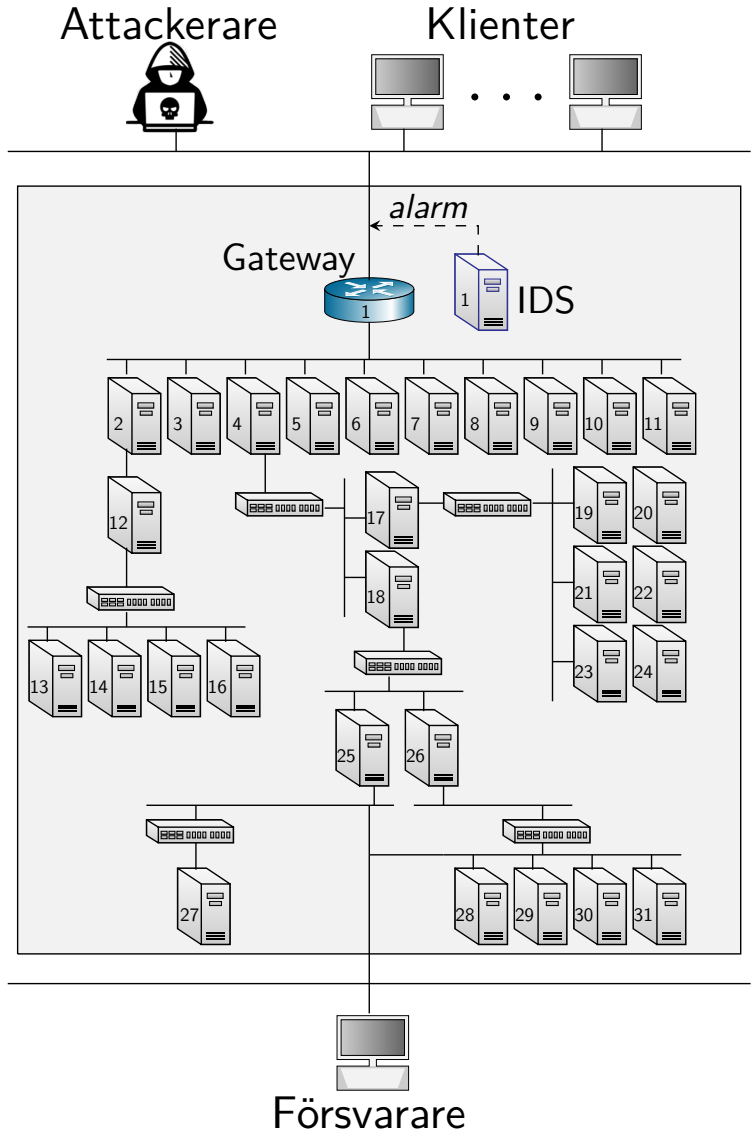
- **Utmaningar**:
  - Attackmetoder är i en konstant förändring och utveckling
  - Komplicerade IT-infrastrukturer



Forskningsmål: Automatiserad säkerhet och inlärning

- **Utmaningar**:
  - Attackmetoder är i en konstant förändring och utveckling
  - Komplicerade IT-infrastrukturer

- **Forskningsmål**:
  - Automatisera säkerhetsfunktioner
  - Anpassa system till föränderliga attackmetoder



DEMO: Automatiserade och självlärande säkerhetsstrategier

Sammanfattning

- **En introduktion till intrångsdetekering och försvar**
  - En viktig del i försvar är att förstå hur en attack går till
  - Försvar: nätverksövervakning och aktivt försvar
  - Intrångsdetekteringssystem
  - Detektering av attack signaturer
  - Detektering av anomalier
  - Åtgärda alarm
  - Strategi