

Automatisoitu sähköpostin vastaanotto Linux ympäristössä

SUGIF SAS Technical Club 22.11.2017

Tapio Kalmi, SAS Institute

Esityksen sisältö

Sähköpostin vastaanotto Linux ympäristössä

- Yleistä
- Case, jonka mukainen ratkaisu toimii 24/365
- Suunnittelu: Top Down
- Työvaiheet
- Toteutus: Bottom Up

Yleistä: SAS ja sähköpostin lähettäminen on ollut mahdollista SAS 8.1 versiosta lähtien

- Konfigurointi
 - `OPTIONS EMAILSYS=SMTP ...`
- Asetusten tarkistus
 - `PROC OPTIONS GROUP=EMAIL;`
 - `RUN;`
- Käyttö
 - [FILENAME Statement, EMAIL \(SMTP\) Access Method](#)
- Windows, Unix ja Linux käyttö samanlaista SAS koodin kannalta

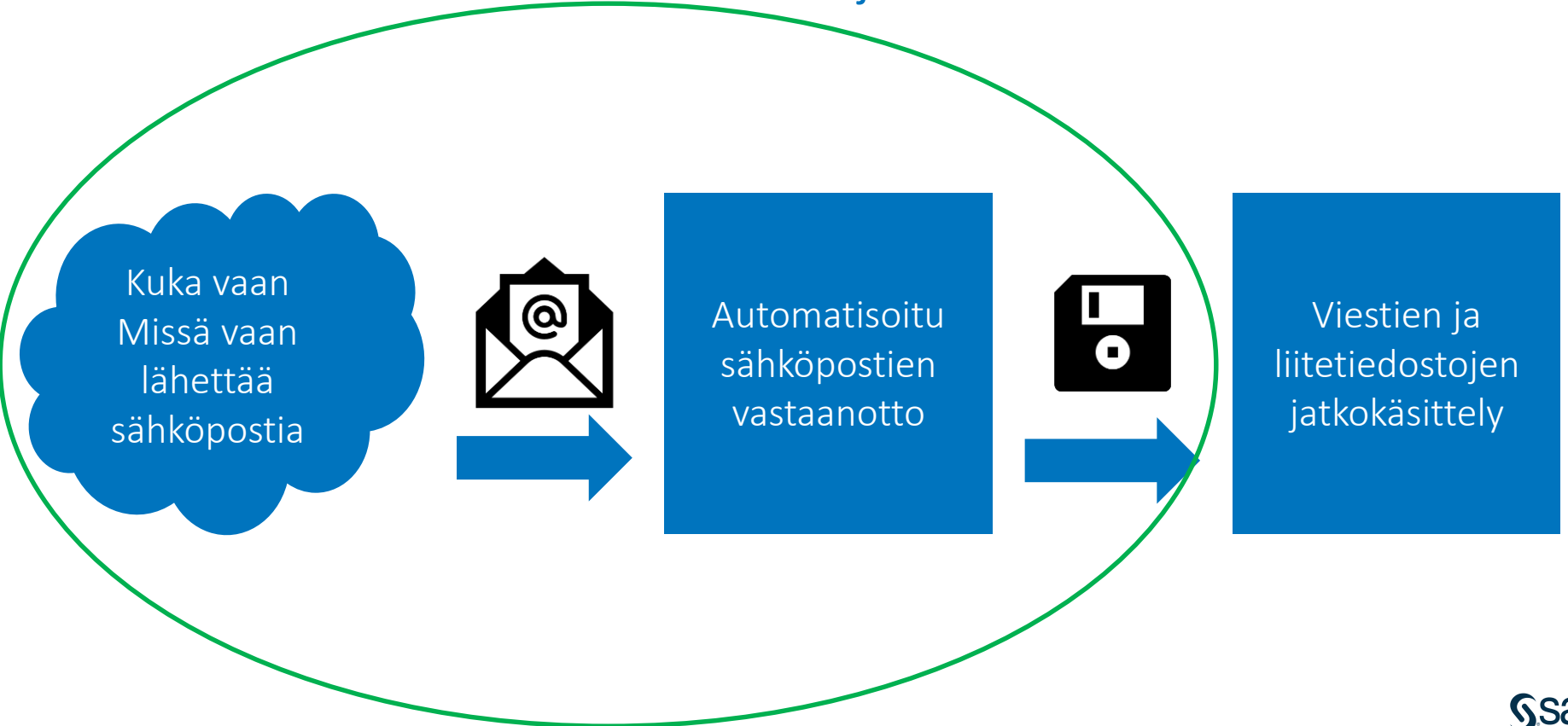
Yleistä: SAS ja sähköpostin vastaanottaminen

???

- SAS ohjelmoinnissa ei ole vakioratkaisua sähköpostien lukuun sähköpostipalvelimilta
- Windows ympäristöissä ratkaisun avaimia ovat sähköpostiohjelmiin tehtävät viestien käsittelysäännöt, joita voidaan laajentaa Visual Basic logiikalla
- Unix ja Linux ympäristöissä tulee tehdä skriptejä, joissa on mail komentoja
- Ympäristöstä riippumatta viestin vastaanoton jälkeen viesti ja liitteet tulee saada talletettua levyille, jonka jälkeen tietoja voidaan käsitellä SAS ohjelmissa normaalisti

Sähköpostin vastaanotto Linux ympäristössä

Case Rajaus



Sähköpostin vastaanotto Linux ympäristössä

Case Arkkitehtuuri



Kysymys 1: Miten saan sähköpostit edustapalvelimelta taustapalvelimelle?

Sähköpostin vastaanotto Linux ympäristössä

Arkkitehtuuri: Edustapalvelin

- yleensä tietoturva vaatimusten vuoksi SAS palvelimille ei sallita ulkopuolista sähköpostiliikennettä vaan tiedonsiirtoa varten on erillinen sähköpostipalvelin, niin sanottu edustapalvelin
- eri tiedonsiirtotarpeita varten luodaan edustapalvelimelle sähköpostitilejä
- sähköpostitilien nimet annetaan viestejä lähettäville tahoille
- lähettäjä tietää vain sähköpostitilin ja palvelimen nimen (=osoite)
- SAS ratkaisua tehtäessä ei voida vaikuttaa mitenkään edustapalvelimien tekniikkaan tai rakenteeseen
- Tieto on saatava siirrettyä edustapalvelimelta sovelluspalvelimelle

FETCHMAIL

Hakee viestit sähköpostipalvelimelta Linux palvelimelle

- Ilmainen apuohjelma, joka noutaa annetun sähköpostitilin sisällön ja välittää sen eteenpäin
 - Huomaa: fetchmail ei todellakaan tee muuta kuin noutaa postin

- Esimerkki ohjauksesta, joilla luetaan kaikki ja poistetaan luetut viestit edustapalvelimelta

```
set syslog set logfile "/opt/thsasapp/lev1/logs/fetchmail/fetchmail.log"
```

```
set daemon 60
```

Millä rytmillä noutoja tehdään (sek)?

```
poll email.palvelin.fi protocol IMAP service 993 user 'postitilin.nimi' password 'sALaSaNA'
```

```
# Perform log rotation (see ~/.logrotate.conf)
```

```
preconnect '/usr/sbin/logrotate -s /home/sasbatch/.logrotate.status /home/sasbatch/.logrotate.conf'
```

```
ssl
```

```
fetchall
```

```
no keep
```

```
no rewrite
```

```
mda '/usr/bin/procmail -d %T'
```

Kutsu procmail ohjelman jatkokäsittelylle

PROCMAIL

Tallettaa haetut viestit Linuxin mail sähköpostilaatikkoon

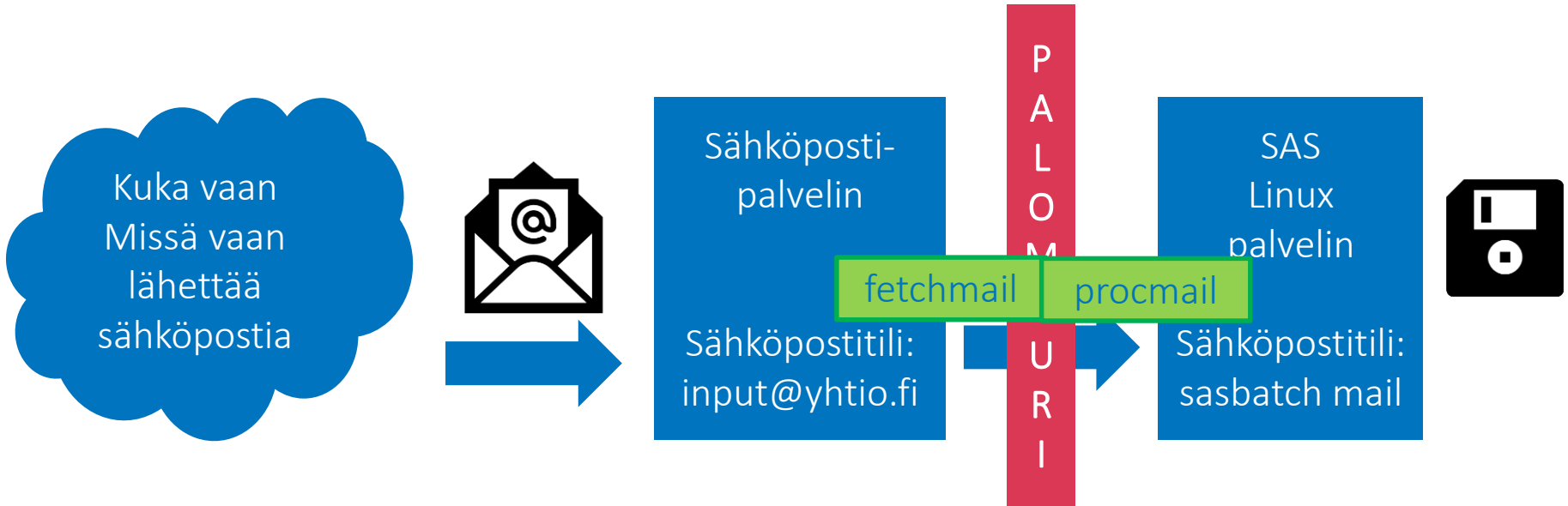
- Ilmainen apuohjelma, käsittelee viestin joka on noudettu jollain muulla ohjelmalla
- Esimerkkinä oletusarvoinen procmail ohjaus

```
:0
*^content-Type:
{
    # backup the complete mail first..
    # you can leave out this part if you don't want a backup of the complete mail
    # :0c:
    # $HOME/mail_backup
}
```

- Tässä ei ole määritelty tehtäväksi mitään, jolloin oletusarvoisesti kaikki viestit kirjoitetaan ohjelmaa ajavan käyttäjän sähköpostilaatikkoon
- Tämä on minimi, mutta procmail ohjelmalla voi tehdä paljon muutakin
- Näin on hyvä toimia, jos kaikki logiikka voidaan toteuttaa myöhemmin yhdellä koodilla

Sähköpostin vastaanotto Linux ympäristössä

Case Arkkitehtuuri



Kysymys 2: Miten saan sähköpostit mail postilaatikosta levyille?

Reunaehdot ja sähköpostien ja liitteiden käsittelyyn

Keep It Simple ;-)

- Sähköpostiohjelmien omien komentojen käyttö on suositeltavaa, koska
 - Viestit kulkevat sähköpostipalvelimien välissä MIME paketteina
 - Viestiin voi kuulua liite tai useita liitteitä
 - Liitteet voivat olla viestejä tai mitä tahansa tiedostoja
 - Liiteviesteissä voi olla liitteitä
- Kannattaa käyttää sähköpostiohjelmien omia komentoja viestin ja liitetiedostojen tallettamiseen levyille erillisinä tiedostoina
 - Näin käytetään sähköpostiohjelmaa siihen mihin se on tehty
 - Näin vältetään se, että pitäisi tehdä ratkaisu MIME pakettien purkuun, joka olisi pyörän keksimistä uudelleen

Postilaatikosta levyille -ratkaisun tekovaiheet

- mieti pseudokoodi
- opettele ja testaa sähköpostikomennot interaktiivisesti
- tee käyttäjärjestelmäkomennot, joilla korvaat interaktiiviset komennot
- liitä käyttäjärjestelmäkomennot SAS ohjelmiin
 - filename + pipe, call system tai X komento
 - XCMD SAS optio tulee olla päällä, että voit ajaa näitä komentoja
- tee eri työvaiheita vastaavat SAS ohjelmat
- tee vastauksista tauluja ja tiedostoja tarpeen mukaan
 - esimerkiksi viestien otsikot on hyvä lukea ensin tauluun, jonka jälkeen voi etsiä haluamaansa viestiä taulua käsittelemällä
 - kun viestin numero on tiedossa, sen ja liitteet voi tallettaa tiedostoina
- tee ohjelmista User Generated transformaatioita (DIS) tai makroja (Base)
- tee jobeja, joissa on tapauskohtaisesti tarvittava logiikka ja toistorakenteet

Esimerkki: Pseudokoodi

- Ennen koodaamista kannattaa miettiä hetki
- Tee logiikasta pseudokoodi
- Esimerkki: Lue ja tarkista kaikki sähköpostiviestit
 - Katso viestien määrä
 - Toista niin monta kertaa kuin viestejä on
 - Lue viestin rakennetiedot
 - Generoi ja aja komento, jolla viesti ja liitteet talletetaan levyille
 - Tutki viestin lähettäjä viestien ryhmittelyä varten
 - Tarkista liitetiedostojen rakenne ja tee virheilmoitukset
 - Siirrä hyväksytyt tiedot sisäänlukualueelle ja virheelliset viestit selvitysalueelle
 - Poista käsitelty viesti sähköpostilaatikosta

Esimerkki: Red Hat Linux mail, otsikot

Kuinka paljon viestejä?

- Ensimmäiseksi tarkistetaan viestien määrä

`mail -H -f otsikot.txt`

- Komento listaa käyttäjän postilaatikon kaikkien viestien otsikkotiedot `otsikot.txt` -nimiseen tiedostoon
- listalla viestin otsikko, numero ja tieto siitä, onko viesti luettu
- Vaatimukset
 - tiedonsiirtoa varten oltava tekninen käyttäjätunnus, jolla on postilaatikko ja jolla ajetaan sisään luku
 - sähköposti tulee olla lähetetty (tai noudettu `fetchmail` ja `procmail` rutiineilla) saman käyttäjätunnuksen postilaatikkoon

Esimerkki: Red Hat Linux mail, viesti

Mitä viesteissä sisällä?

- Edellä luettuja otsikkotietoja tutkimalla tulee löytää haluttu viesti TAI on luettava kaikki viestit yksi kerrallaan. Yhden viestin rakennetiedot haetaan tyyliin

```
echo 'print 9' | mail -f viestinRakenne.txt
```

- Komento listaa viestin numero 9 rakennetiedot viestinRakenne.txt –nimiseen tiedostoon
- mail komento avaa sähköpostiohjelman niin, että ohjelman komentoihin saadut vastaukset kirjoitetaan tiedostoon
- echo komento kiihdyttää mail ohjelmalle print 9 komennon, joka tulostaa viestin numero 9
- viestin lähettäjän tiedot, lähetysaika ja viesti ovat tiedostossa
- lisäksi liitetiedostoista saadaan rakennetiedot, joista käyttökelpoisimpia ovat tiedot tiedostonimistä ja tiedostotyypeistä

Esimerkki: Red Hat Linux mail, liitteet

Talletetaan viesti ja liitetiedostot levyllä

- Kun vestin rakenne on saatu selvitettyä, talletetaan viesti ja liitetiedostot levyllä komennolla

```
printf 'cd polkunimi\nwrite 9 viesti.txt\n\n\n' | mail -f userid
```

- Komento tallettaa viestin ja liitetiedostot polkunimi nimiseen hakemistoon, jonka jälkeen viesti ja kolme liitetiedostoa ovat polkunimi kansiossa luettavissa alkuperäisissä tiedostomuodoissaan
- mail komento avaa sähköpostiohjelman ja käyttää userid nimisen käyttäjän postilaatikkoa tai userid nimistä tiedostoa varsinaisen postilaatikon sijaan
- printf komento kaiuttaa mail ohjelmalle \n (=enter) erotellut komennot
 - cd polkunimi\n vaihtaa kansion, minne tiedostot talletetaan
 - write 9 viesti.txt\n tallettaa viestin viesti.txt tiedostoon
 - \n\n\n ovat kolme enteriä, joilla vastataan mail ohjelman write komennon liitetiedostokohtaisiin kysymyksiin, talletetaanko liite alkuperäisellä nimellä (3 liitettä, 3 enteriä)

Esimerkki: User Generated Transformations

”Duplo palikat”

- Transformaatio per tehtävä
- Otsikot
UX mail: Headers summary for all messages to table
- Viesti
UX mail: Message data to table
- Liitteet
UX mail: Message text and attachments to external files
- Käsitellyn viestin poisto
UX mail: Delete Message

Esimerkki: Pseudokoodista SAS jobi

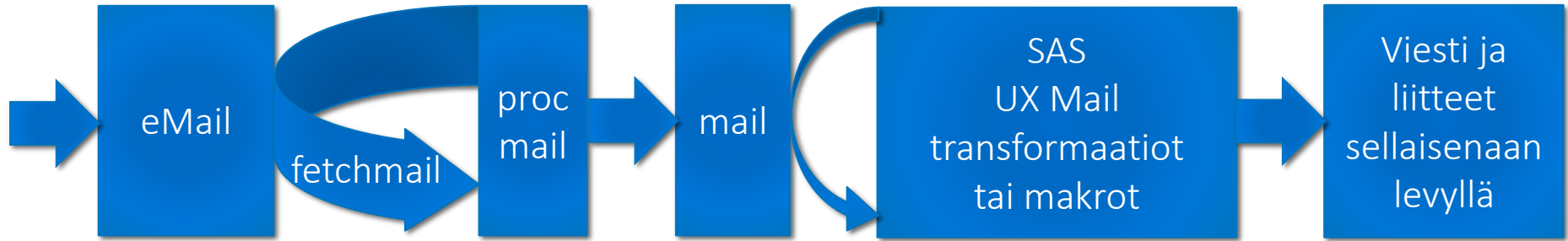
- Jobi: Lue ja tarkista kaikki sähköpostiviestit
- UX mail: Headers summary for all messages to table
- Loop Start
- UX mail: Message data to table
- UX mail: Message text and attachments to external files
- Tähän voi tulla sovelluskohtaista logiikkaa, joka ei liity sähköpostirajapintaan
 - Tutki viestin lähettäjä viestien ryhmittelyä varten
 - Tarkista liitetiedostojen rakenne ja tee virheilmoitukset
 - Siirrä hyväksytyt tiedot sisäänlukualueelle ja virheelliset viestit selvitysalueelle
- UX mail: Delete Message
- Loop End

Vinkki: Tarkkana viestin poiston kanssa

- Kun poistat viestin sähköpostilaatikosta, niin seuraavalla kertaa mail-komentoa suoritettaessa kaikkien poistetun viestin jälkeisten viestien numerot ovat pienentyneet yhdellä.
- Näin ollen numeroita käyttäen on hankala rakentaa selkeää logiikkaa, jos viestejä poistetaan
- Jos viestit poistetaan käsittelyn jälkeen, on helpointa koko ajan käsitellä viestiä numero 1.
- Olennaista on toistojen määrä, jos kaikki viestit halutaan käsitellä

Sähköpostin vastaanotto Linux ympäristössä

Yhteenveto



Vastaanotto
Win/Linux/...

Nouto edustapalvelimelta
Linux palvelimelle

Sisällön purku Linux palvelimella



KYSYTTÄVÄÄ? KIITOS!

sas.com