SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

Session - 41 Agenda:
1. SMTP Relay Mail Server Configuration Using Ansible

Sendmail:
Sendmail is a general-purpose internetwork email routing facility that supports many kinds of mail-transfer and delivery methods, including the Simple Mail Transfer Protocol used for email transport over the Internet.

Sending mails in a linux system to other users:
First, we have to install any Mail Transport Agent tool like sendmail.
 # dnf install -y sendmail

Now we have to start the service and also set it at boot time so that it can start automatically at its own when we reboot the machine.
 # systemctl enable --now sendmail.service

Check the status of service.
 # systemctl status sendmail.service

Create a file for sending the mail like below, having all details like mail subject and body.
 # vim /tmp/email.txt
Subject: System Reboot Time At 10 PM IST.
Hi Vikas,
This system will reboot at 10 PM IST, please save your work before time.
Thanks
Admin

Now, send the mail to vikasnehra user using sendmail command.
 # sendmail vikasnehra@localhost < /tmp/email.txt

Now, verify that the user vikasnehra has received this mail.
$ cd /var/spool/mail
$ ls -lh
$ sudo cat vikasnehra

You can also check the mail logs for more details.
 # tail -20 /var/log/maillog

Some details you can also capture from /var/log/messages file as well.
 # tail -20 /var/log/messages

You can also send mail using echo command with sendmail like this.
 # echo "leaving now" | sendmail vikasnehra

Verify the same.
 # cat /var/spool/mail/vikasnehra

You can also use mail command to send emails. But it requires mailx or s-nail package install in the system.
 # mail -s "Test Subject" vikasnehra@localhost < /dev/null
 # dnf install -y mailx #(for rhel 7 & 8 only, not rhel 9)
 # dnf install -y s-nail # (for rhel 9 only)
 # mail -s "Test Subject" vikasnehra@localhost < /dev/null

SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

Verify the same, whether the user has received it or not.
$ sudo cat /var/spool/mail/vikasnehra

OR you can use mail command to see the details of all emails you have received.
$ mail

To send an attachment included within the email, type the below-mentioned line.
 # cal > /tmp/cal.txt
 # vim /tmp/mail.txt
Hi, please find the attached file for the calendar.
 # mail -a /tmp/cal.txt -s "December Calendar" vikasnehra@localhost < /tmp/mail.txt

Verify the same for vikasnehra user.
$ mail

You can also use echo command with filter to use with mail command to send emails.
 # echo "Nehra Classes Are Awesome" | mail -s "Nehra Classes" vikasnehra@localhost

Verify the same from vikasnehra user account.
$ mail

You can also see the details of what is going on at the background using -v option after mail command for the verbosity.
 # echo "Nehra Classes Are Awesome" | mail -v -s "Nehra Classes Are Awesome" vikasnehra@localhost

Verify the same from vikasnehra user account.
$ mail

You can also use journalctl command to see the mail activity for sendmail service.
 # journalctl -u sendmail.service -x

To send emails to other domains or systems you can use relay mail server.

Please follow the SMTP Relay Mail Configuration tutorial of Nehra Classes for the same. Link Below 😊
https://youtube.com/live/BgGRS-i7x5A

SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

Configuration of SMTP Relay Mail Server Using Ansible:

Let's create an ansible playbook to setup the SMTP relay mail server at the managed node(s).

```
$ vim mail-server.yml
---
- name: Relay Mail Server Configuration Using SMTP Playbook
  hosts: node1
  become: true
  tasks:
   - name: Setting up the static hostname in the machine.
     hostname:
       name: mail-server.nehraclasses.local
       use: systemd

   - name: Installing sendmail packages in the machine.
     dnf:
       name:
         - sendmail*
         - procmail
         - s-nail
         - make
       state: latest

   - name: Creating authinfo file in the /etc/mail directory having thentication information.
     template:
       src: authinfo.j2
       dest: /etc/mail/authinfo

   - name: Copying sendmail configuration file in the /etc/mail directory.
     template:
       src: sendmail.mc.j2
       dest: /etc/mail/sendmail.mc
       force: true

   - name: Making changes in the sendmail.cf file using m4 processor with the help of sendmail.mc file.
     shell: m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf

   - name: Creating encypted database file (authinfo.db) using authinfo file.
     shell: makemap hash /etc/mail/authinfo < /etc/mail/authinfo

   - name: Starting & enabling the sendmail service.
     service:
       name: sendmail
       state: started
       enabled: yes
...
```

Create ansible jinja template files which are used in the playbook.

```
$ vim authinfo.j2
AuthInfo:smtp.gmail.com "U:nehraclassestester@gmail.com" "P:iliftbmofciciayq" "M:PLAIN"

$ vim sendmail.mc.j2
divert(-1)dnl
dnl #
dnl # This is the sendmail macro config file for m4. If you make changes to
```

SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
dnl # /etc/mail/sendmail.mc, you will need to regenerate the
dnl # /etc/mail/sendmail.cf file by confirming that the sendmail-cf package is
dnl # installed and then performing a
dnl #
dnl #     /etc/mail/make
dnl #
include(`/usr/share/sendmail-cf/m4/cf.m4')dnl
VERSIONID(`setup for linux')dnl
OSTYPE(`linux')dnl
dnl #
dnl # Do not advertize sendmail version.
dnl #
dnl define(`confSMTP_LOGIN_MSG', `$j Sendmail; $b')dnl
dnl #
dnl # default logging level is 9, you might want to set it higher to
dnl # debug the configuration
dnl #
dnl define(`confLOG_LEVEL', `9')dnl
dnl #
dnl # Uncomment and edit the following line if your outgoing mail needs to
dnl # be sent out through an external mail server:
dnl #
define(`SMART_HOST', `smtp.gmail.com')dnl
FEATURE(`authinfo')
dnl #
define(`confDEF_USER_ID', ``8:12'')dnl
dnl define(`confAUTO_REBUILD')dnl
define(`confTO_CONNECT', `1m')dnl
define(`confTRY_NULL_MX_LIST', `True')dnl
define(`confDONT_PROBE_INTERFACES', `True')dnl
define(`PROCMAIL_MAILER_PATH', `/usr/bin/procmail')dnl
define(`ALIAS_FILE', `/etc/aliases')dnl
define(`STATUS_FILE', `/var/log/mail/statistics')dnl
define(`UUCP_MAILER_MAX', `2000000')dnl
define(`confUSERDB_SPEC', `/etc/mail/userdb.db')dnl
define(`confPRIVACY_FLAGS', `authwarnings,novrfy,noexpn,restrictqrun')dnl
define(`confAUTH_OPTIONS', `A')dnl
dnl #
dnl # The following allows relaying if the user authenticates, and disallows
dnl # plaintext authentication (PLAIN/LOGIN) on non-TLS links
dnl #
dnl define(`confAUTH_OPTIONS', `A p')dnl
dnl #
dnl # which realm to use in SASL database (sasldb2)
dnl #
define(`confAUTH_REALM', `mail')dnl
dnl #
dnl # PLAIN is the preferred plaintext authentication method and used by
dnl # Mozilla Mail and Evolution, though Outlook Express and other MUAs do
dnl # use LOGIN. Other mechanisms should be used if the connection is not
dnl # guaranteed secure.
dnl # Please remember that saslauthd needs to be running for AUTH.
dnl #
dnl TRUST_AUTH_MECH(`EXTERNAL DIGEST-MD5 CRAM-MD5 LOGIN PLAIN')dnl
dnl define(`confAUTH_MECHANISMS', `EXTERNAL GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN PLAIN')dnl
```

SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
dnl #
dnl # Basic sendmail TLS configuration with self-signed certificate for
dnl # inbound SMTP (and also opportunistic TLS for outbound SMTP).
dnl #
define(`confCACERT_PATH', `/etc/pki/tls/certs')dnl
define(`confCACERT', `/etc/pki/tls/certs/ca-bundle.crt')dnl
define(`confSERVER_CERT', `/etc/pki/tls/certs/sendmail.pem')dnl
define(`confSERVER_KEY', `/etc/pki/tls/private/sendmail.key')dnl
define(`confTLS_SRV_OPTIONS', `V')dnl
dnl #
dnl # This allows sendmail to use a keyfile that is shared with OpenLDAP's
dnl # slapd, which requires the file to be readble by group ldap
dnl #
dnl define(`confDONT_BLAME_SENDMAIL', `groupreadablekeyfile')dnl
dnl #
dnl define(`confTO_QUEUEWARN', `4h')dnl
dnl define(`confTO_QUEUERETURN', `5d')dnl
dnl define(`confQUEUE_LA', `12')dnl
dnl define(`confREFUSE_LA', `18')dnl
define(`confTO_IDENT', `0')dnl
dnl # If you're operating in a DSCP/RFC-4594 environment with QoS
dnl define(`confINET_QOS', `AF11')dnl
dnl FEATURE(delay_checks)dnl
FEATURE(`no_default_msa', `dnl')dnl
FEATURE(`smrsh', `/usr/sbin/smrsh')dnl
FEATURE(`mailertable', `hash -o /etc/mail/mailertable.db')dnl
FEATURE(`virtusertable', `hash -o /etc/mail/virtusertable.db')dnl
FEATURE(redirect)dnl
FEATURE(always_add_domain)dnl
FEATURE(use_cw_file)dnl
FEATURE(use_ct_file)dnl
dnl #
dnl # The following limits the number of processes sendmail can fork to accept
dnl # incoming messages or process its message queues to 20.) sendmail refuses
dnl # to accept connections once it has reached its quota of child processes.
dnl #
dnl define(`confMAX_DAEMON_CHILDREN', `20')dnl
dnl #
dnl # Limits the number of new connections per second. This caps the overhead
dnl # incurred due to forking new sendmail processes. May be useful against
dnl # DoS attacks or barrages of spam. (As mentioned below, a per-IP address
dnl # limit would be useful but is not available as an option at this writing.)
dnl #
dnl define(`confCONNECTION_RATE_THROTTLE', `3')dnl
dnl #
dnl # The -t option will retry delivery if e.g. the user runs over his quota.
dnl #
FEATURE(local_procmail, `', `procmail -t -Y -a $h -d $u')dnl
FEATURE(`access_db', `hash -T<TMPF> -o /etc/mail/access.db')dnl
FEATURE(`blocklist_recipients')dnl
EXPOSED_USER(`root')dnl
dnl #
dnl # For using Cyrus-IMAPd as POP3/IMAP server through LMTP delivery uncomment
dnl # the following 2 definitions and activate below in the MAILER section the
dnl # cyrusv2 mailer.
```

SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
dnl #
dnl define(`confLOCAL_MAILER', `cyrusv2')dnl
dnl define(`CYRUSV2_MAILER_ARGS', `FILE /var/lib/imap/socket/lmtp')dnl
dnl #
dnl # The following causes sendmail to only listen on the IPv4 loopback address
dnl # 127.0.0.1 and not on any other network devices. Remove the loopback
dnl # address restriction to accept email from the internet or intranet.
dnl #
DAEMON_OPTIONS(`Port=smtp,Addr=127.0.0.1, Name=MTA')dnl
dnl #
dnl # The following causes sendmail to additionally listen to port 587 for
dnl # mail from MUAs that authenticate. Roaming users who can't reach their
dnl # preferred sendmail daemon due to port 25 being blocked or redirected find
dnl # this useful.
dnl #
dnl DAEMON_OPTIONS(`Port=submission, Name=MSA, M=Ea')dnl
dnl #
dnl # The following causes sendmail to additionally listen to port 465, but
dnl # starting immediately in TLS mode upon connecting. Port 25 or 587 followed
dnl # by STARTTLS is preferred, but roaming clients using Outlook Express can't
dnl # do STARTTLS on ports other than 25. Mozilla Mail can ONLY use STARTTLS
dnl # and doesn't support the deprecated smtps; Evolution <1.1.1 uses smtps
dnl # when SSL is enabled-- STARTTLS support is available in version 1.1.1.
dnl #
dnl # For this to work your OpenSSL certificates must be configured.
dnl #
dnl DAEMON_OPTIONS(`Port=smtps, Name=TLSMTA, M=s')dnl
dnl #
dnl # The following causes sendmail to additionally listen on the IPv6 loopback
dnl # device. Remove the loopback address restriction listen to the network.
dnl #
dnl DAEMON_OPTIONS(`port=smtp,Addr=::1, Name=MTA-v6, Family=inet6')dnl
dnl #
dnl # enable both ipv6 and ipv4 in sendmail:
dnl #
dnl DAEMON_OPTIONS(`Name=MTA-v4, Family=inet, Name=MTA-v6, Family=inet6')
dnl #
dnl # We strongly recommend not accepting unresolvable domains if you want to
dnl # protect yourself from spam. However, the laptop and users on computers
dnl # that do not have 24x7 DNS do need this.
dnl #
FEATURE(`accept_unresolvable_domains')dnl
dnl #
dnl FEATURE(`relay_based_on_MX')dnl
dnl #
dnl # Also accept email sent to "localhost.localdomain" as local email.
dnl #
LOCAL_DOMAIN(`localhost.localdomain')dnl
dnl #
dnl # The following example makes mail from this host and any additional
dnl # specified domains appear to be sent from mydomain.com
dnl #
dnl MASQUERADE_AS(`mydomain.com')dnl
dnl #
dnl # masquerade not just the headers, but the envelope as well
```

SMTP Relay Mail Server Configuration Using Ansible

By: Er. Vikas Nehra (M. Tech, B. Tech), Experience: 15 + Years

```
dnl #
dnl FEATURE(masquerade_envelope)dnl
dnl #
dnl # masquerade not just @mydomainalias.com, but @*.mydomainalias.com as well
dnl #
dnl FEATURE(masquerade_entire_domain)dnl
dnl #
dnl MASQUERADE_DOMAIN(localhost)dnl
dnl MASQUERADE_DOMAIN(localhost.localdomain)dnl
dnl MASQUERADE_DOMAIN(mydomainalias.com)dnl
dnl MASQUERADE_DOMAIN(mydomain.lan)dnl
MAILER(smtp)dnl
MAILER(procmail)dnl
dnl MAILER(cyrusv2)dnl
```

Execute the ansible playbook to setup the mail server at the managed node(s).

$ ansible-playbook mail-server.yml

Check and verify the status of the sendmail service at the managed node(s).

$ ansible node1 -m command -a 'sudo systemctl status sendmail'

Use ansible ad-hoc command or go to the mail server to send a mail to another domain like gmail.com

$ ssh node1 ; sudo su -

 # echo "Nehra Classes Are Awesome" | mail -v -s "Nehra Classes Are Awesome" nehraclasses@gmail.com

Now, login to the recipient email account and check whether you have received the mail or not.

How to enable root login alert emails in Linux:

Go to home directory of root user & longlist the contents of it including hidden files.

 # cd

 # ls -al

Here you will see the .bashrc profile file, open it in vi editor and paste the line mentioned below there.

 # vim .bashrc

echo 'ALERT - Root Shell Access 192.168.229.129 (node1) on:' `date` `who` | mail -s " Alert: Root Access" nehraclasses@gmail.com

Now, go to inbox of the recipient and verify the mail delivery.

The SMTP Replay Mail Server is working perfectly as expected.

Thank You