NAGIOS CORE:

* Nagios is a tool for monitoring system; nagios watches computers and devices on the network and makes sure they are working as expected.
* System Monitoring in Nagios is divided into two parts:
  + Hosts: Represent the physical machine, VM, server on the network.
  + Services: these are the particular functionalities for specific hosts. Each service is associated with a host on which it’s running.
* Nagios only checks a service in 4 different states ok, Warning, critical & unknown.
* Nagios performs all of its checks using plugins. These are external plugins to which nagios passes information to on what should be checked and what are the threshold limits for the critical, warning states.
* Main Features: Nagios is very flexible and can be configured as per the requirements.
  + **Commands:** these are definitions on how nagios should perform certain type of checks. Commands are an abstraction layer on top of plugins and allow us to group similar type of actions together.
  + **Time Periods:** these are time periods over a certain action should or should not be performed.
  + **Hosts and Host Groups:** these are devices along with the possibility of grouping; a single host may be a part of multiple groups.
  + **Services:**  Functionalities or resources to be monitored on a HOST.
  + **Contact and Contact Groups:** People who should be notified when a services or host goes down or state is changed. A single contact can be a member of multiple groups
  + **Notifications:** These define who be notified of what
  + **Escalations:** These are extension of notifications. They define if a certain service is a state for a specific time then other people should be notified for the same.
* **Soft and Hard States:** nagios works on checking the state of the service and storing its value. If a service goes down, nagios performs a check on it several times to make sure this new state is permanent; this new state is termed as soft state. Once nagios is sure that this new state is permanent, it changes the state to hard state and sends out the notification to the contacts.
* **Installation of Nagios (CentOS 7):** The installation is done from the compiling the source code. Any customization made for changing the default directory for installation will be highlighted. For this installation I have used CentOS7 as base operating system and created a **sudo user nagios** 
  + **Pre-Requisites:**yum install gcc make imake binutils cpp postgresql-devel mysql-libs mysql-devel openssl openssl-devel pkgconfig gd gd-devel gd-progs libpng libpng-devel libjpeg libjpeg-devel perl perl-devel net-snmp net-snmp-devel net-snmp-perl net-snmp-utils httpd php
  + **Directory Structure:** /opt/nagios/nagios-core – install directory for Nagios, this is the location for all nagios binaries, plugins, and additional files.  
    /opt/nagios/nagios-core /var – data directory for nagios. This is where status of everything is stored. This can be a part of nagios binary install directory or a separate directory like /var/nagios.  
    /opt/nagios/nagios-core /etc– nagios configuration will reside in here, this directory will be created as a part of nagios installation.  
    create the following directories by running :  
    mkdir -p nagios/nagios-core/etc  
    mkdir -p nagios/nagios-core/var
  + **User and Groups**:   
    user – nagios  
     groups - nagios,nagioscmd  
    groupadd nagios  
    groupadd nagioscmd  
    useradd -g nagios -G nagioscmd (if user nagios is already added , you can user usermod instead of useradd to modify usergroups)
  + Create a source directory under /opt/nagios/source  
    mkdir –p /opt/nagios/source.
  + Extract the nagios core and plugins archive source , and you will get directories nagios-4.3.4 nagios-plugins-2.2.1 under /opt/nagios/source
  + Assuming you have already switched to user nagios and added nagios as sudo user. Change the owner and owner group for /opt/nagios by running command  
    sudo chown –R nagios.nagios /opt/nagios
  + Change directory to nagios-core source directory and run the configure script with below parameterized values to configure nagios install paths/options.  
    sh configure --prefix=/opt/nagios/nagios-core --sysconfdir=/opt/nagios/nagios-core/etc --localstatedir=/opt/nagios/nagios-core/var --libexecdir=/opt/nagios/nagios-core/plugins --with-command-group=nagioscmd --with-nagios-user=nagios --with-nagios-group=nagios  
    After running configure.sh, you should get the output like:  
    *===============================================  
    Creating sample config files in sample-config/ ...  
    \*\*\* Configuration summary for nagios 4.3.4 2017-08-24 \*\*\*:  
     General Options:*
  + Run make allexpected end output after successful compile looks like  
    *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
    Enjoy.*
  + Now we need to install nagios by running below commands.  
    make install  
    make install-commandmode  
    make install-config
  + **Install Nagios Plugins:**Go to source directory for nagios-plugins and run below command to configure makesh configure --prefix=/opt/nagios/nagios-core --sysconfdir=/opt/nagios/nagios-core/etc --localstatedir=/opt/nagios/nagios-core/var --libexecdir=/opt/nagios/nagios-core/plugins --with-nagios-user=nagios --with-nagios-group=nagios --enable-perl-modules  
    make all  
    make install
  + Once the installation of nagios plugin is complete, check the installation by running  
    /opt/nagios/nagios-core/bin/nagios /opt/nagios/nagios-core/etc/nagios.cfg

If the install is successful, we should get the output like   
*Nagios 4.3.4 starting... (PID=4593)*

*Local time is Mon Mar 12 15:55:17 IST 2018*

*nerd: Channel hostchecks registered successfully*

*nerd: Channel servicechecks registered successfully*

*nerd: Channel opathchecks registered successfully*

*nerd: Fully initialized and ready to rock!*

*wproc: Successfully registered manager as @wproc with query handler*

*wproc: Registry request: name=Core Worker 4597;pid=4597*

*wproc: Registry request: name=Core Worker 4594;pid=4594*

*wproc: Registry request: name=Core Worker 4595;pid=4595*

*wproc: Registry request: name=Core Worker 4596;pid=4596*

*Successfully launched command file worker with pid 4606*

* + After Successful install of nagios, configure nagios as service by runningmake install-init  
    sudo chkconfig --add nagios ; sudo chkconfig nagios on
  + Start Nagios service by running   
    sudo service nagios start
  + Check the logs at /opt/nagios/nagios-core/var  
    tail –f nagios.log
  + **Configuring Web Interface:**
    - We already installed apache2 when installing pre-requisites. To configure Nagios Web Interface, we will have to tweak the httpd configuration.
    - Go to cd /etc/httpd/httpd.conf/
    - Create a file touch nagios.conf and paste in below information  
        
      *ScriptAlias /nagios/cgi-bin /opt/nagios/nagios-core/sbin  
      Alias /nagios /opt/nagios/nagios-core/share  
      LogFormat "%h %l %u \"%r\" %>s %b %{Host}e %f" debuglog  
      CustomLog /var/log/apache2/access-debug.log debuglog  
      <DirectoryMatch /opt/nagios/nagios-core/share>  
       Options FollowSymLinks  
       AllowOverride AuthConfig  
       Order Allow,Deny  
       Allow From All  
       AuthName "Nagios Access"  
       AuthType Basic  
       AuthUserFile /opt/nagios/nagios-core/etc/htpasswd.users  
       AuthGroupFile /opt/nagios/nagios-core/etc/htpasswd.groups  
       require valid-user  
      </DirectoryMatch>  
      <DirectoryMatch /opt/nagios/nagios-core/sbin>  
       Options ExecCGI  
       AllowOverride AuthConfig  
       Order Allow,Deny  
       Allow From All*

*AuthName "Nagios Access"  
 AuthType Basic*

*AuthUserFile /opt/nagios/nagios-core/etc/htpasswd.users*

*AuthGroupFile /opt/nagios/nagios-core/etc/htpasswd.groups*

*require valid-user*

*</DirectoryMatch>*

* + - Run below commands to add the nagios admin user to apache authentication  
      cp /dev/null /opt/nagios/nagios-core/etc/htpasswd.groups
    - htpasswd -c /opt/nagios/nagios-core/etc/htpasswd.users nagiosadmin  
      this will prompt you to set the password, set the password as you will.
    - Check the httpd user and and its groups, if its apache user and not a part of nagios group , modify user groups by running command  
      sudo usermod apache –G nagios
    - Restart httpd and nagios service  
      sudo service httpd restart ; sudo service nagios restart
    - You can use check\_http plugin to verify if apache is running and nagios is is up or not by running below commands, the return status should be 200 OK in all of the commands, if it’s different you can check for the detailed error in the debug logs mentioned in nagios.conf file under /etc/httpd/httpd.conf/nagios.conf  
      /opt/nagios/nagios-core/plugins/check\_http -H 127.0.0.1   
      /opt/nagios/nagios-core/plugins/check\_http -H 127.0.0.1 -u /nagios -a nagiosadmin:admin  
      /opt/nagios/nagios-core/plugins/check\_http -H 127.0.0.1 -u /nagios/cgi-bin/tac.cgi -a nagiosadmin:admin
    - Go to your favorite browser and type in http://<ip\_of\_the box>:80/nagios and enjoy the WEB - INterface  
      if you got 200 ok in all the above commands and still the Web Page is unavailable make sure firewall for port 80 is open and apache service is running.
  + **Configuring Nagios:**  the configuration file for nagios should locate at /opt/nagios/nagios-core/etc/nagios.cfg
    - This cfg files contains all the information on the hosts and services to be defined
* Configure Email Notification on nagios
  + To configure email notification we will have to have a mail utility installed on the Linux system which can be used with nagios.
  + All the examples in these notes are for CentOS7.
  + Download sendEmail from internet and extract the tar by following below steps  
    wget <http://caspian.dotconf.net/menu/Software/SendEmail/sendEmail-v1.56.tar.gz>  
    tar –xzvf sendEmail\*\*\*.tar.gz  
    cp sendEmail-v1.56/sendEmail /usr/local/bin
  + Go to nagios install directory and modify the commands.cfg in our case  
     cd /opt/nagios/nagios-core/etc/objects  
    vi commands.cfg   
    comment the existing notify-host-by-email and notify-service-by-email commands and paste in  
      
    *define command{*

*command\_name notify-host-by-email*

*command\_line /usr/bin/printf "%b" "\*\*\*\*\* Nagios \*\*\*\*\*\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME$\n" | /usr/local/bin/sendEmail -f $USER5$ -s $USER7$:$USER8$ -u "\*\* $NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$ \*\*" -t $CONTACTEMAIL$ -xu $USER9$ -xp $USER10$*

*}*

*define command{*

*command\_name notify-service-by-email*

*command\_line /usr/bin/printf "%b" "\*\*\*\*\* Nagios \*\*\*\*\*\n\nNotification Type: $NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\nState: $SERVICESTATE$\n\nDate/Time: $LONGDATETIME$\n\nAdditional Info:\n\n$SERVICEOUTPUT$" | /usr/local/bin/sendEmail -f $USER5$ -s $USER7$:$USER8$ -u "\*\* $NOTIFICATIONTYPE$ Service Alert: $HOSTALIAS$/$SERVICEDESC$ is $SERVICESTATE$ \*\*" -t $CONTACTEMAIL$ -xu $USER9$ -xp $USER10$*

*}*

* + Now we need to make sure that the variables $USER5$,$USER7$, $USER8$, $USER9$,$USER10$, are defined on the resource.cfg located on <nagios\_install\_directory>/etc, where  
    $USER5$ = senders email address  
    $USER7$ = smtp server address/ip of smpt server (I preferred IP)

$USER8$ = SMTP PORT  
$USER9$ = UserName for SMTP authentication  
$USER10$ = Password for SMTP authentication

You can also execute below to check if the sendEmail is working as expected, where $USER$ variables are to be replaced with original values.  
/usr/bin/printf "%b" "\*\*\*\*\* Nagios \*\*\*\*\*\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME$\n" | /usr/local/bin/sendEmail -f $USER5$ -s $USER7$:$USER8$ -u "\*\* $NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$ \*\*" -t $CONTACTEMAIL$ -xu $USER9$ -xp $USER10$

* + Now All we need to have is to define contact, contact groups and have notifications enabled for service and hosts in corresponding cfg files
  + Once the Notifications have been enabled, check and verify nagios changes by running  
    <nagios\_install\_directory>/bin/nagios –v <nagios\_install\_directory>/etc/nagios.cfg