

Nagiosinstallation auf virt. Maschine mit Ubuntu 5.04

- Boot-CD starten
- Grundinstallation von CD wird ausgeführt
- System mit apt auf den neuesten Stand bringen
 - *apt-get update*
 - *apt-get upgrade*
- sources.list von apt erweitern, um weitere Software einzuspielen
 - *locate sources.list*
 - *vi /etc/apt/sources.list*
 - Server auskommentieren, die in die Suche mit eingeschlossen werden sollen
 - vi mit *:wq* speichern und beenden
 - erneut *apt-get update* und *apt-get upgrade* ausführen
- Anlegen des Benutzers nagios und der Gruppe nagios
 - In Gnome unter *System > Systemverwaltung > Benutzer und Gruppen* den Benutzer und die Gruppe *nagios* anlegen
 - Den Benutzer anschließend den Gruppen *nagios* und *www-data* hinzufügen um die Funktion von Nagios zu gewährleisten, da Nagios unter dem Benutzer nagios ausgeführt werden soll und da die cgi's mit PHP und der Gruppe www-data bearbeitet werden
- Das System sollten nun auf dem neuesten Stand sein und es kann weitergehen mit der Installation von Apache2, PHP4 und Nagios

Weitere essentielle Programme installieren

- openssh
- openssl
- gcc (C-Compiler)
- ntp
- ntp-server
- ntp-simple
- ntpdate
- apt-show-versions
- apt-show-source

Installation von Apache2

- *apt-cache search apache* ausführen um das Paket zu finden
- *apt-get install apache2* ausführen um Apache2 zu installieren
- mit *locate -u* kann die Indexierung von Dateien erneuert werden um neu installierte Dateien in die Suche einzubinden
- Apache sollte nun installiert, gestartet und im Browser über localhost aufrufbar sein

Verzeichnispfad	Datei
/etc/apache2	httpd.conf (alt) apache2.conf (aktuell)
/var/lock/	
/var/run/	apache2.pid
/var/log/apache2/	Apache-Log-Dateien
/usr/lib/apache2/modules/	Apache-Module
/var/www/	Apache Web-root

Installation von PHP4

- *apt-cache search php* um die passenden Pakete zu finden
- *apt-get install php4* um das Paket zu installieren
- *locate -u* ausführen um den Index zu aktualisieren
- In der *php.ini* den Wert *Register Globals* auf *On* setzen
- *vi /etc/php4/apache2/php.ini*
- PHP sollte damit schon funktionieren

Installation von Nagios

- *apt-cache search nagios* um die passenden Pakete zu finden
- *apt-get install nagios-text* um die Pakete zu installieren
- *nagios-plugins* und *nagios-common* werden automatisch installiert
- vorgeschlagen werden die *nagios-plugin-extras*
- empfohlen wird: *libnet-snmp.perl snmp ntp-simple*
- Es wird automatisch der Nutzer *nagiosadmin* angelegt und nach einem Passwort gefragt
- Dabei wird in */etc/nagios/* die Datei *htpasswd.users* angelegt, in der die Benutzer für Nagios verwaltet werden
- *locate -u* ausführen um den Index zu aktualisieren

Verzeichnispfad	Datei
/etc/nagios/	Konfigurationsdateien
/usr/sbin/	Startdatei
/var/log/nagios/	Nagios-Log-Dateien
/usr/lib/nagios/plugins/	Nagios-Plugins
/usr/lib/cgi-bin/nagios/	Nagios-CGI's
/usr/share/nagios/htdocs/	Nagios Web-root

- Nun muss die apache.conf aus dem /etc/nagios/ Ordner noch in der apache2.conf included werden
- *vi /etc/apache2/apache2.conf*
- Dann bei den Includes folgende Zeile hinzufügen
- *Include /etc/nagios/apache.conf*
- Neustart von Apache2 mit */etc/init.d/apache2 restart*
- Somit sollte die Authentifizierung funktionieren
- Localhost/nagios im Browser öffnen
- Es soll nun die Aufforderung zur Authentifizierung kommen
- Name und Passwort eingeben
- Nun sollte man Zugriff auf die verschiedenen CGI's haben

Typische Fehler und deren Lösung

- Zugriffsrechte für Ordner und Dateien
- Für cgi's und cfg's sollten folgende Rechte eingestellt werden
- *chown -R nagios.www-data .* im aktuellen Verzeichnis ausführen
- *chmod -R 775 ** im aktuellen Verzeichnis ausführen
- Commandfile-Probleme von Nagios
- Sollten beim Start von Nagios Probleme mit dem nagios.cmd File auftauchen, wie folgt vorgehen
- Ist der Ordner */var/spool/nagios/* nicht vorhanden
- Ordner mit den Rechten wie oben genannt anlegen, dann sollte Nagios starten
- Sollte der Ordner und die Datei existieren, es aber dennoch zu Problemen beim Schreibzugriff kommen, dann sollte einfach die nagios.cmd gelöscht werden. Beim nächsten Zugriff durch Nagios wird diese automatisch neu angelegt

- Log-Files werden nicht korrekt ausgewertet
- Sollte der Ordner `/var/log/nagios/archives/` nicht vorhanden sein, dann muss dieser ebenfalls mit den oben genannten Rechten für `nagios.www-data` erstellt werden. Steht nun die Log-Rotation z.B. auf `daily`, so sollte jeden Tag eine neue Logdatei angelegt werden und die alte automatisch im Ordner `archives` landen.

Postfix konfigurieren

- Postfix sollte standardmäßig installiert sein und als `forwarder` eingestellt sein
- Es muss also nur noch der ausgehende `smtp`-Server eingerichtet werden
- `vi /etc/postfix/main.cf`
- hier den `outgoing smtp-server` setzen
- es muss nun in der `nagios.cfg` nur der `MailAdmin` auf einen aktiven und gültigen Account gesetzt werden, damit Nagios die Benachrichtigungsmails automatisch versendet

VMware-Tools installieren

- C-Compiler muss installiert sein
- aktuelle Kernelversion mit `uname -a` herausfinden
- Kernelheader installieren
- `apt-get install linux-headers-kernelversion`
- `tar.gz` File von der VMwareCD auf die HD kopieren
- `tar xvzf vmware-tools.tar.gz` entpackt das Paket
- in erstellten Ordner wechseln
- `./vmware-install.pl` ausführen
- eventuell `/usr/bin/vmware-config-tools.pl` von Hand ausführen
- `/usr/bin/vmware-toolbox` startet ein X-Window für die Tools
- Zeit mit Host synchronisieren aktivieren

IP-Zugriffsbeschränkung

So sollte die `/etc/nagios/apache.conf` aussehen:

```
ScriptAlias /cgi-bin/nagios /usr/lib/cgi-bin/nagios
ScriptAlias /nagios/cgi-bin /usr/lib/cgi-bin/nagios
<DirectoryMatch /usr/lib/cgi-bin/nagios> ->
    Options ExecCGI
    AllowOverride
    AuthConfig Order Deny,Allow
    Deny From All
    Allow From 127.0.0.1 ...
    AuthName "Nagios Access"
    AuthType Basic
    AuthUserFile /etc/nagios/htpasswd.users require valid-user </DirectoryMatch>
```

SSL für Apache2 einrichten

1. `/usr/bin/openssl req -new > new.cert.csr`
 - Passphrase und weitere Daten eingeben
2. `openssl rsa -in privkey.pem -out new.cert.key`
 - Passphrase eingeben
3. `openssl x509 -in new.cert.csr -out new.cert.cert -req -signkey new.cert.key -days 365`
4. Link zu `/etc/apache2/ssl/new.cert.cert` SSLCertificateFile
5. Link zu `/etc/apache2/ssl/new.cert.key` SSLCertificateKeyFile
6. `cp /etc/apache2/sites-available/default /etc/apache2/sites-available/ssl`
7. `ln -s /etc/apache2/sites-available/ssl /etc/apache2/sites-enabled/ssl`
8. `vi /etc/apache2/sites-enabled/ssl`
 - Die Ports an SSL anpassen
 - `NameVirtualHost *:443`
 - `<VirtualHost *:443>`
9. `vi /etc/apach2/ports.conf`
 - `Listen 433` hinzufügen damit an diesem Port Anfragen angenommen werden
10. `vi /etc/apache2/sites-enabled/ssl`
 - Folgendes muss hinzugefügt werden
 - `SSLEngine On`
 - `SSLCertificateFile /etc/apache2/ssl/new.cert.cert`
 - `SSLCCertificateKeyFile /etc/apache2/ssl/new.cert.key`

`/etc/init.d/apache2 force-reload` startet Apache neu und somit sollte https verfügbar sein