Mit Nagios das Netzwerk voll im Griff

Philipp Seidel

Chemnitzer StudentenNetz

8.11.2008

Outline

Netzwerkstruktur des Chemnitzer StudentenNetz

- Technik:
  - 6 Server
  - 10 virtuelle Server
  - ca. 80 Switches mit ca. 3700 aktiven Ports
  - zentraler Router
  - 11 W-LAN Accespoints
  - ca. 1800 - 2000 Nutzer (werden nicht geprüft)
- Dienste:
  - SSH
  - Webseite
  - SMTP-Server
  - DNS-Server
  - DHCP-Server
  - Netboot
  - Jabber

Problem und Lösung

Problem

- Netzwerkinfrastruktur wächst
- Prüfung von Hosts und Diensten wird schwerer
- frühzeitiges finden von Problemen erschwert

Lösung

- automatisierte Überwachung

Nagios

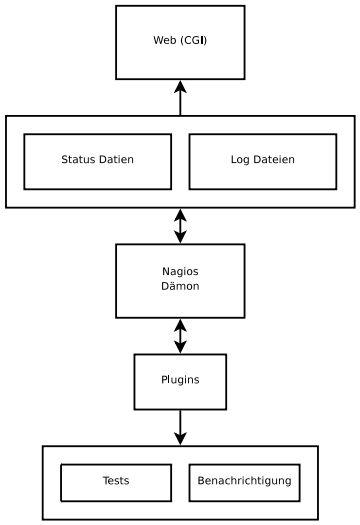
Vorteile

- sehr leicht konfigurierbar
- leicht erweiterbar
- sehr flexibel
- OpenSource
- verschiedene Benachrichtigungsformen
- Konfiguration in Textdateien

Nachteile

- gewöhnungsbedürftiges Web-Interface
- Konfiguration in Textdatein

Struktur



Was kann überwacht werden?

- Systeme:
  - Linux/Unix System
  - Windows Systeme
  - Routers, Switches, Hubs
  - Drucker
- Dienste:
  - öffentliche Dienste (SSH, HTTP, SMTP, ...)
  - private Dienste (CPU, RAM Nutzung, Festplattenplatz, ...)

Check Arten

aktive Checks

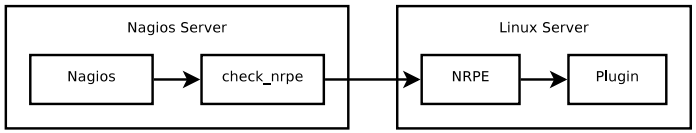
- vom Nagios Daemon ausgelöst
- in regelmässigen Abständen ausgeführt
- Daemon ruft Plugin auf und wertet Rückgabewert aus

passive Checks

- durch externe Programme durchgeführt
- liefern Status zur weiteren Verarbeitung an Nagios

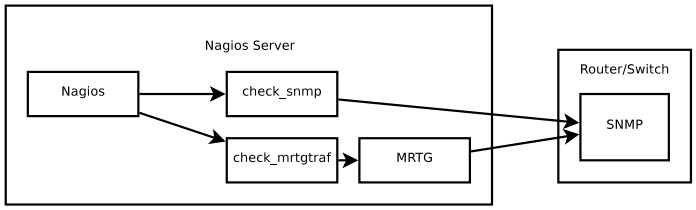
private Dienste: Linux/Unix

- zwei grundlegende Methoden
- SSH Verbindung:
  - benutzen von verteilten SSH-Keys
  - Plugin: check_by_ssh
  - sehr aufwendig
  - hohe CPU-Auslastung
- NRPE Addon:

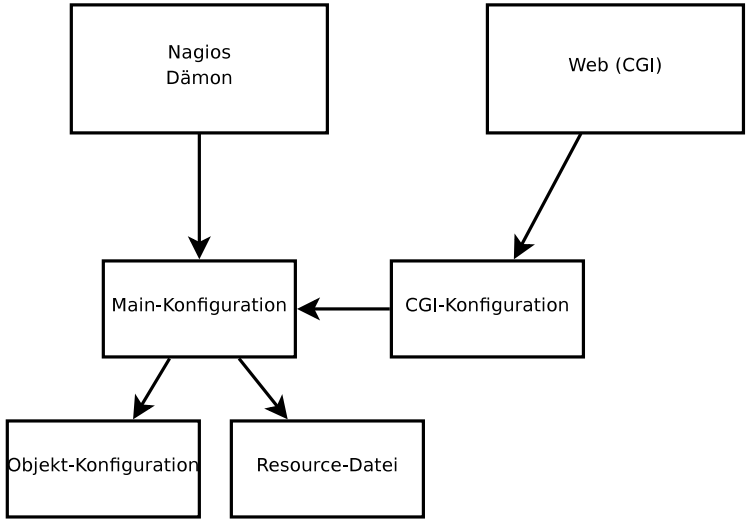


private Dienste: Router/Switch

- per SNMP
- per MRTG



Struktur der Konfiguration



Ordnerstruktur

1. lokale Installation in /usr/local/nagios
   - bin - enthält der eigentlichen Nagios Dämon
   - etc - Konfigurationsdateien
   - libexec - Plugins
   - sbin - CGI Scripte
   - share - statische Dateien für die Webseite
   - var - Daten und Log-Dateien
2. Paketgebundene Installation

Konfigurationsdateien

- nagios.cfg - die Hauptkonfigurationsdatei

```
...
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
...
cfg_dir=/usr/local/nagios/etc/servers
...
```

- resource.cfg - Pfadangaben, Passwörter

```
...
$USER1$=/usr/local/nagios/libexec
...
#$USER2$=/usr/local/nagios/libexec/eventhandlers
...
```

- cgi.cfg - CGI Konfiguration
- objects/ - weitere Objekte, z.B.: Komandos, Server

Konfigurationsdateien objects/

- commands.cfg - Kommandos zum Benachrichtigen und um Tests durchzuführen

```
...
# 'notify-host-by-email' command definition
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/
              bin/mail -s "** $NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$
              **" $CONTACTEMAIL$
        }
...
# 'check-host-alive' command definition
define command{
        command_name    check-host-alive
        command_line    $USER1$/check_ping -H $HOSTADDRESS$ -w 3000.0,80% -c 5000.0,100% -
              p 5
        }
...
```


Konfigurationsdateien objects/

- contacts.cfg - Personen die Benachrichtigt werden sollen

```
..
define contact{
        contact_name                    hans-muster
        alias                           Hans Muster
        service_notification_period     24x7
        host_notification_period        24x7
        service_notification_options    w,u,c,r
        host_notification_options       d,u,r
        service_notification_commands   notify-service-by-mail
        host_notification_commands      notify-host-by-mail
        email                           muster@example.com
        pager                           12345678
        }
...
define contactgroup{
        contactgroup_name       admins
        alias                   Nagios Administrators
        members                 hans-muster
        }
...
```


Konfigurationsdateien objects/

- localhost.cfg - Checks für Localhost
- printer.cfg - Checks für Drucker
- switch.cfg - Services und Hosts der Switches/Router
- templates.cfg - Templates für Hosts und Services

```
...
define host{
        name                            linux-server
        use                             generic-host
        check_period                    24x7
        check_interval                  5
        retry_interval                  1
        max_check_attempts              10
        check_command                   check-host-alive
        notification_period             24x7
        notification_interval           120
        notification_options            d,u,r
        contact_groups                  admins
        hostgroups                      servers-ping, ssh-servers
        register                        0
        }
...
```

Konfigurationsdateien objects/

- timeperiods.cfg

```
...
define timeperiod{
        timeperiod_name 24x7
        alias           24 Hours A Day, 7 Days A Week
        sunday          00:00-24:00
        monday          00:00-24:00
        tuesday         00:00-24:00
        wednesday       00:00-24:00
        thursday        00:00-24:00
        friday          00:00-24:00
        saturday        00:00-24:00
        }
...
```

- windows.cfg - Services und Hosts der Windows-Server

Konfigurationsdateien objects/

- servers.cfg - Services und Hostgroups für Server (selbst angelegt)

```
...
define hostgroup {
        hostgroup_name  servers-ping
        alias           Pingable servers
}
...
define service{
        hostgroup_name                  servers-ping
        service_description             PING
        check_command                   check_ping!100.0,20%!500.0,60%
        use                             generic-service
        process_perf_data               1
}
...
```


Konfigurationsdateien servers/

- test-server.cfg (selbst angelegt)

```
...
define host {
        host_name       test-server
        alias           Test Server
        address         12.34.56.78
        parents         test-router
        use             linux-server
        hostgroups      +dns-servers
        }
...
```


Plugin-Aufbau

- Plugin Struktur recht einfach aufgebaut
- Voraussetzung:
  - ausführbar
  - bestimmten Rückgabewert
  - mindestens eine Zeile auf STDOUT ausgeben

Rückgabewert	Service Status	Host Status
0	OK	UP
1	WARNING	UP or DOWN/UNREACHABLE
2	CRITICAL	DOWN/UNREACHABLE
3	UNKNOWN	DOWN/UNREACHABLE

Erstellen des Plugins

- Plugin muss standardmäßig in Verzeichnis libexec/
- Beispiel: md5sum_test.sh

```
1  #!/bin/sh
2
3  HOSTNAME=$1
4  FILE=$2
5  MD5SUM_ORIG=$3
6
7  MD5SUM=$(wget -q -O "-" "http://$HOSTNAME$FILE" | md5sum | awk '{print $1}')
8
9  if [ "$MD5SUM" = "$MD5SUM_ORIG" ]; then
10         echo "MD5SUM: $MD5SUM"
11         exit 0
12 else
13         echo "MD5SUM: none"
14         exit 2
15 fi
```


Erstellen des Plugins

- Neues Kommando in der Datei commands.cfg anlegen
- Beispiel:

```
1  define command {
2          command_name md5sum_test
3          command_line $USER1$/md5sum_test.sh $HOSTADDRESS$ $ARG1$ $ARG2$
4          }
```

- Neuen Service in der Datei server.cfg anlegen
- Beispiel:

```
1  define service {
2          use                     generic-service
3          host_name               test-server
4          service_descripton      MD5SUM Test
5          check_command           md5sum_test!/lit-banner-2008.jpg!97121
                 f38355f64fff92d2e719fa22033
6  }
```


vordefinierte Macros

- gekennzeichnet durch $MACRO_NAME$
- werden durch Nagios durch den entsprechenden Wert ersetzt
- Beispiel:

```
1  define service{
2    check_command           check_something!5!10
3    ...
4  }
5  define command {
6    command_name check_something
7    command_line ./something -a $ARG1$ -b $ARG2$
8  }
```

- Ergebnis:

```
./something -a 5 -b 10
```


eigene Macros

- in Definition mit mit führendem Unterstrich gekennzeichnet
- Zugriff mit:
  - $_HOSTvarname$
  - $_SERVICEvarname$
  - $_CONTACTvarname$
- Beispiel:

```
1  define host{
2    host_name           server
3    _MACADDRESS 00:01:02:03:04:05
4    ...
5  }
6  define command {
7    command_name check_something
8    command_line ./something -m $_HOSTMACADDRESS$
9  }
```

- Ergebnis:

```
./something -m 00:01:02:03:04:05
```

- Funktionsweise fast wie Plugin
- externem Programm wird zusendende Nachricht übergeben

- E-Mail
- SMS
- Anruf
- Pager
- Jabber
- IRC
- Elektroschocker ;-)

SMS-Versand per SMS Server Tools 3

- zum Versand der SMS einfach Datei in /var/spool/sms/outgoing/ anlegen
- Format der SMS-Datei:

```
1 To: 1234567
2
3 Test SMS
```


Script anlegen

Datei: /libexec/send_sms.sh

```
1  #!/bin/bash
2
3  NUMBER=$1
4  MESSAGE=$2
5
6
7  FILE="/var/spool/sms/outgoing/"$(echo "$MESSAGE-$NUMBER"$(date) | md5sum - | awk '{print $1}')".
       sms"
8
9  echo "To: $NUMBER" > $FILE
10 echo "" >> $FILE
11 echo $MESSAGE >> $FILE
12 chmod a+rw $FILE
```


Benachrichtungskommando im Nagios anlegen

Datei: /etc/objects/commands.cfg

```
1  define command{
2          command_name notify-service-by-sms
3          command_line $USER1$/send_sms.sh $CONTACTPAGER$ '$NOTIFICATIONTYPE$: $HOSTNAME$:
                $SERVICEDESC$ is $SERVICESTATE$ ($SERVICEOUTPUT$)'
4  }
5
6  define command{
7          command_name notify-host-by-sms
8          command_line $USER1$/send_sms.sh $CONTACTPAGER$ '$NOTIFICATIONTYPE$: $HOSTNAME$ is
                $HOSTSTATE$ ($HOSTOUTPUT$)'
9  }
```

Personen konfigurieren

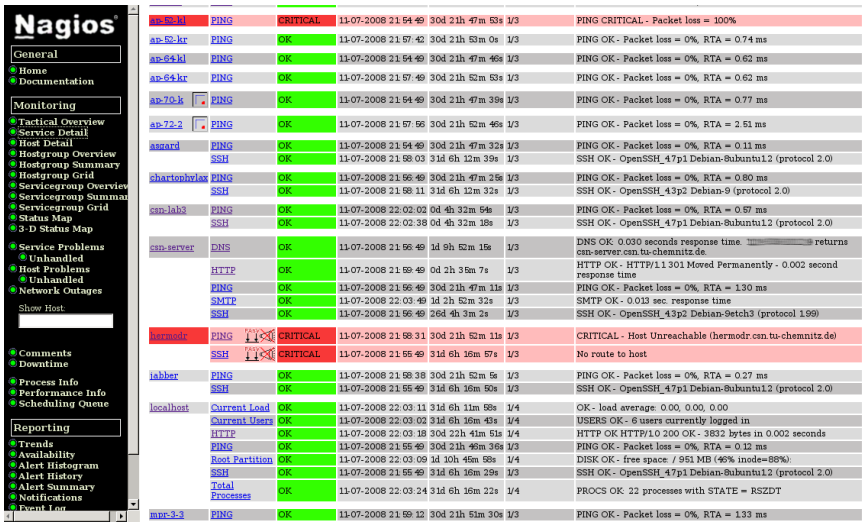
Datei: /etc/objects/contacts.cfg

```
define contact{
        contact_name                    hans-muster
        alias                           Hans Muster
        service_notification_period     24x7
        host_notification_period        24x7
        service_notification_options    w,u,c,r
        host_notification_options       d,u,r
        service_notification_commands   notify-service-by-mail, notify-service-by-sms
        host_notification_commands      notify-host-by-mail, notify-host-by-sms
        email                           muster@example.com
        pager                           12345678
        }
```

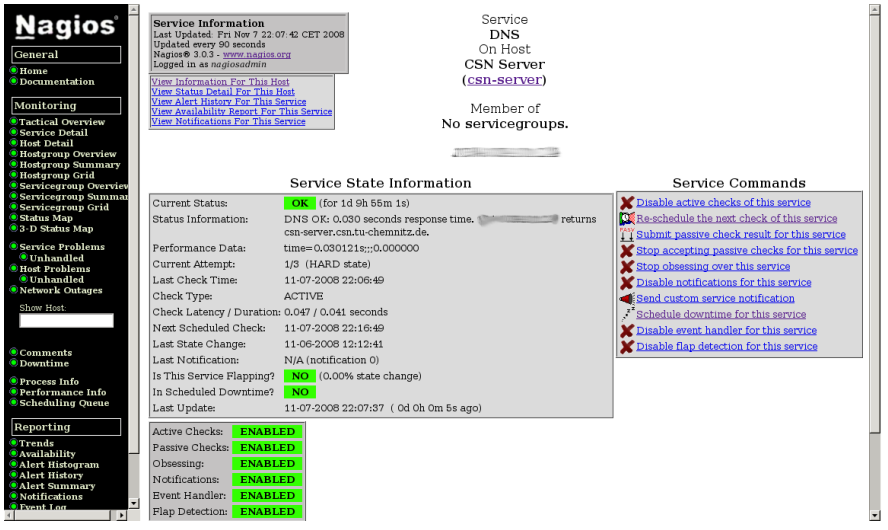

Demo SMS

- RECOVERY: test-server is UP (PING OK - Packet loss = 0%, RTA = 0.88 ms)
- PROBLEM: test-server is DOWN (CRITICAL - Host Unreachable (12.34.56.78))

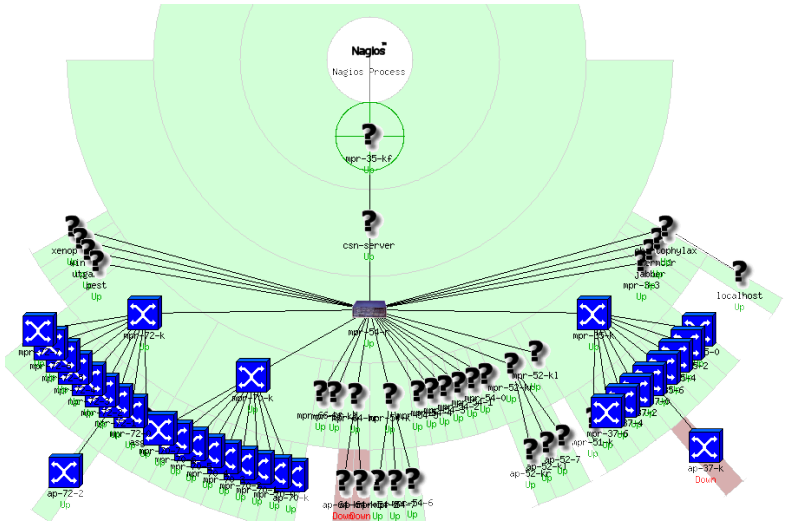
Host-Übersicht



Host-Ansicht



Status-Map



weitere Informationen

- http://www.nagios.org/
- http://nagiosplug.sourceforge.net/
- http://www.nagiosexchange.org/
- http://nagioswiki.org/
- http://smstools3.kekekasvi.com/
- http://www.dinotools.de/

Fragen

Fragen?