功能強大、開源免費之網路管理軟體 Nagios 安裝與應用實務介紹

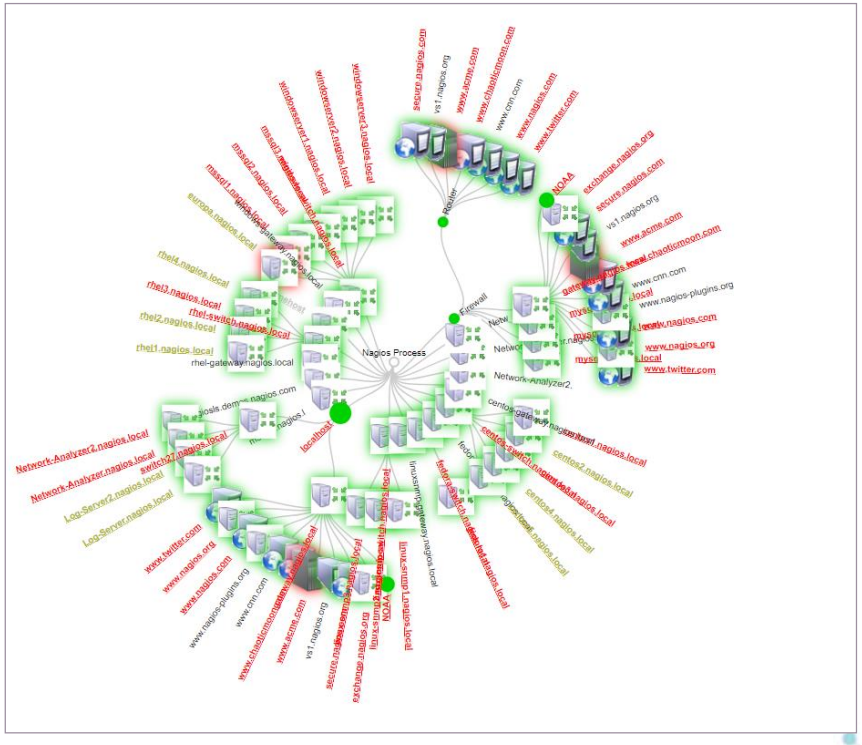
中山大學(高屏澎區網中心)
王聖全

Nagios® Core™

Agenda

- Nagios Core 簡介及安裝
- Nagios Core 主機及服務檢測機制
- Nagios Core 擴充與Plugin整合
- Nagios Core Notification機制實例應用

Nagios Users and Story



Nagios Saves News Year Eve For Admins at Argentina Economics Ministry

Some admins – including myself – have to work on New Years Eve. On December 31st of 2006 things went very wrong with our datacenter. I was at home at 3:00 pm and an SMS arrived. I thought that someone was sending nice happy New Year message, but i was wrong! Nagios was telling that the datacenter was going down because the power went out!

Thanks to Nagios I could solve the problem before midnight and celebrate New Year with my family!

Luis Aparicio, Ministerio de Economia – Argentina

MECON

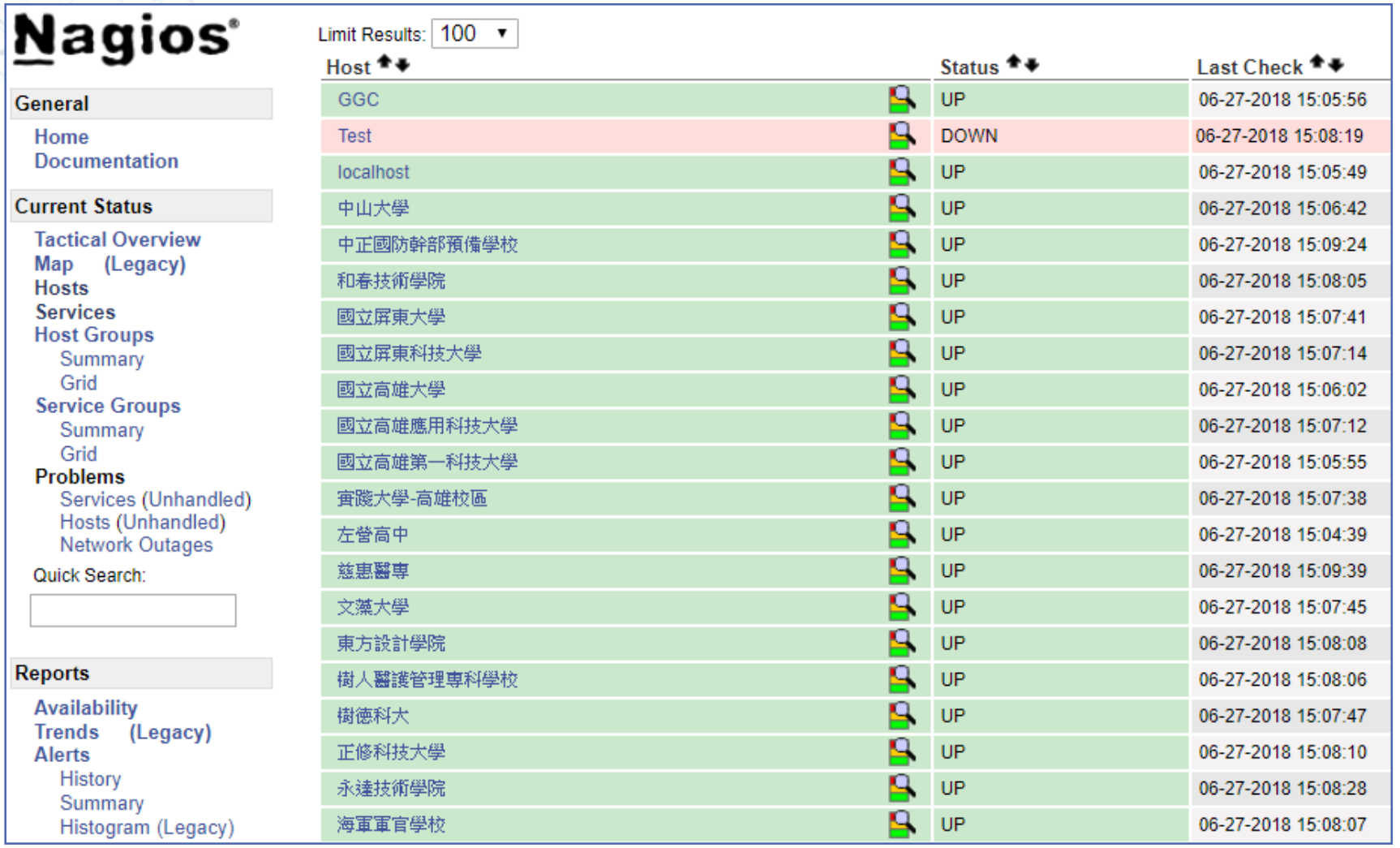
https://www.nagios.com/casestudies/

Nagios Core introduction and Installation

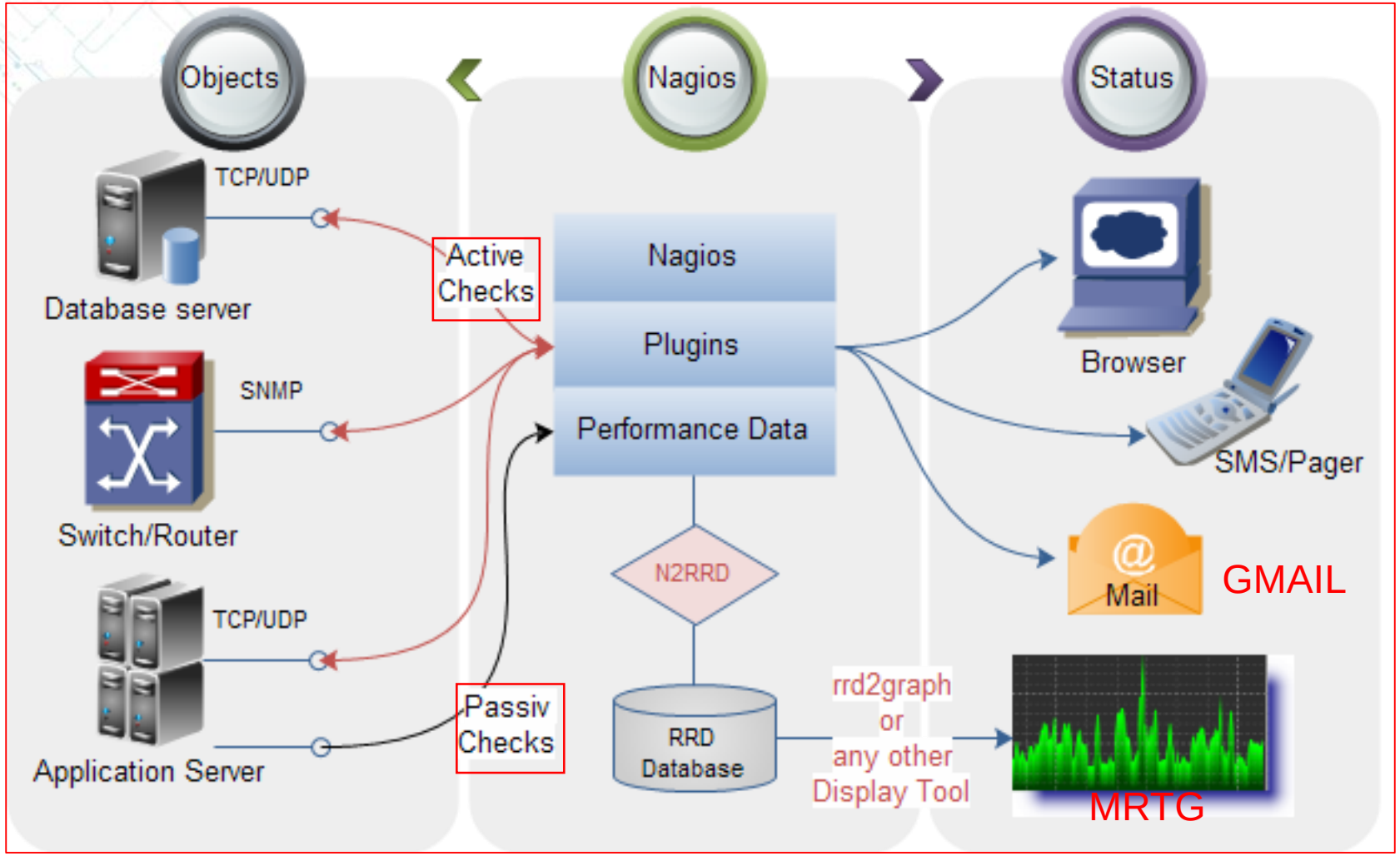
Nagios Core Introduction

- Open source monitoring tools for troubleshooting
- Network monitoring
  - Availability
  - Reliability
- Server monitoring
  - CPU
  - DISK
- Application monitoring
- Alerting service
- Nagios plugins

Nagios Monitoring Example



Overview



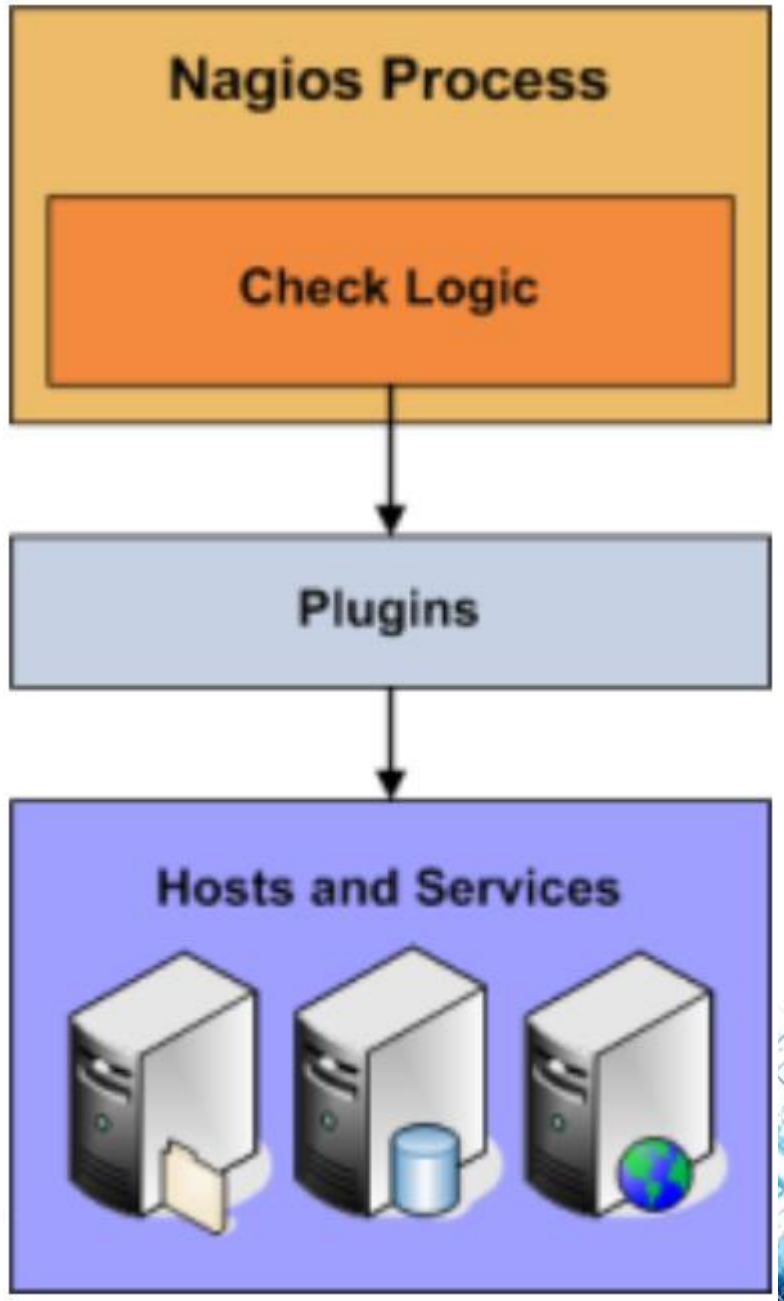
https://en.wikipedia.org/wiki/Nagios#/media/File:Monitoring.png

Two ways of monitoring

- Host and Service monitoring
  - Active Checks (主動檢測)
  - Passive Checks (被動檢測)

Active Checks

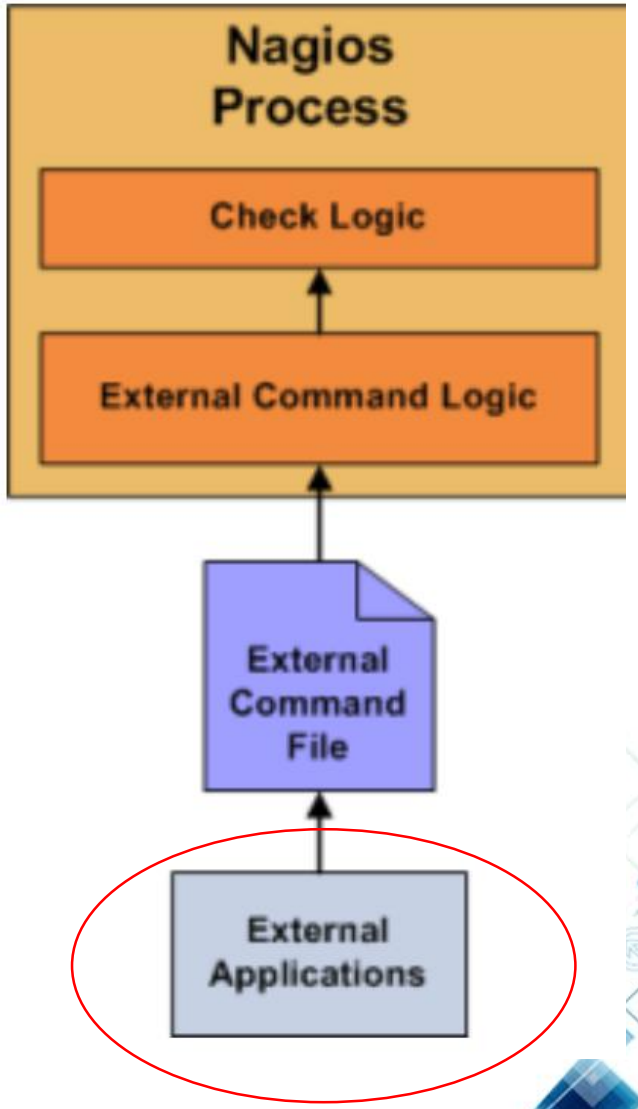
- How
  - Initiated by the check logic
  - Execute a plugin
- When
  - Regular scheduled interval
    - Check_interval
    - Retry_interval
    - On-demand as need



https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/activechecks.html

Passive Checks

- Initiated and performed by external applications/processes
- Asynchronous
- Why passive
  - Service located in a firewall
  - Asynchronous in nature
    - SNMP traps
    - security alerts



State Concept

- Soft State
  - Maybe broken
  - Need to confirm
  - Retry many times
- Hard State
  - A host or service has been re-checked
  - Definitely dead.

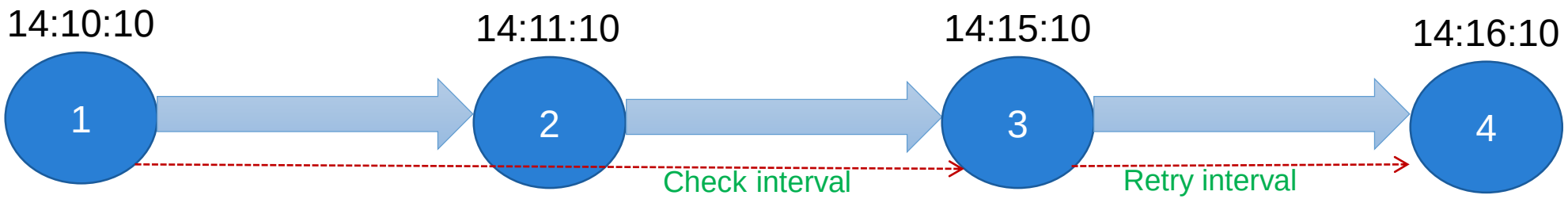
States Transition

- Service and Host Check Retries
  - *max_check_attempts* parameter
  - Prevent false alarm from transient problems
- SOFT states
  - Not yet to *max_check_attempts* checks
  - Recovery from soft_error
- HARD states
  - Already checked *max_check_attempts* times
  - etc
- Basically

  HARD OK → SOFT → HARD non-OK → HARD OK.

State Transition Example

- check_interval 5
- retry_interval 1
- max_check_attempts 3



- Nagios check host1
- Result = OK
- HARD state
- Attempt 1/3
- Next scheduled check 14:15:10

- HOST1 died
- Nagios doesn't know

Host 1

- Nagios check host1
- Check timed out
- Result = CRITICAL
- SOFT state
- Attempt 1/3
- No notification sent
- Next scheduled check 14:16:10

- Nagios check host1
- Check timed out
- Result = CRITICAL
- SOFT state
- Attempt 2/3
- No notification sent
- Next scheduled check 14:17:10

State Transition Example (cont.)

- check_interval 5
- retry_interval 1
- max_check_attempts 3

- Nagios check host1
- Check timed out
- Result = CRITICAL
- HARD state
- Attempt 3/3
- Notification SENT
- Next scheduled check 14:22:10

- Nagios check host1
- Check timed out
- Result = CRITICAL
- HARD state
- Attempt 3/3
- Next scheduled check 14:27:10

- host1 recovers
- Nagios does not know about this yet

Host 1

- Nagios check host1
- Result = OK
- HARD state
- Attempt 1/3
- Next scheduled check 14:22:10

Host Checks

- Check performed
  - At regular intervals
    - *check_interval* (in host definition)
    - *retry_interval* (in host definition)
- On-demand cases
  - when a service associated with the host changes state
  - as needed as part of the
  - host reachability logic
  - as needed for predictive host dependency checks
- Host States
  - UP
  - DOWN
  - UNREACHABLE

Service Checks

- Check performed
  - At regular intervals
    - check_interval (in service definitions)
    - retry_interval (in service definitions)
- Service States
  - OK
  - WARNING
  - UNKNOWN
  - CRITICAL

Installation

- OS Version: Ubuntu 18.04.1 LTS Version
- Nagios Version: Nagios-4.4.2 (Lastest Stable Version, 2018/8/16 released)
  - Check changelog: https://github.com/NagiosEnterprises/nagioscore/blob/master/Changelog

Nagios Core

Latest version

Latest Version 4 Releases

Version	Date	Notes	Type	Link
4.4.2	2018-08-16	Latest stable release	Source code	nagios-4.4.2.tar.gz
4.4.1	2018-06-25	Previous stable release	Source code	nagios-4.4.1.tar.gz
4.4.0	2018-06-19	Previous stable release	Source code	nagios-4.4.0.tar.gz
4.3.4	2017-08-24	Previous stable release	Source code	nagios-4.3.4.tar.gz
4.2.4	2016-12-07	Previous stable release	Source code	nagios-4.2.4.tar.gz

Installation (Download the source)

- sudo apt-get update
- sudo apt-get install -y autoconf gcc libc6 make wget unzip apache2 php libapache2-mod-php7.2 libgd-dev
- cd ~/Download
- wget -O nagioscore.tar.gz https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.2.tar.gz
- tar xzf nagioscore.tar.gz

https://support.nagios.com/kb/article/nagios-core-installing-nagios-core-from-source-96.html#Ubuntu

Installation (Compile the source)

- cd ~/Downloads/nagioscore-nagios-4.4.2
- sudo ./configure --with-httpd-conf=/etc/apache2/sites-enabled
- sudo make all

--with-httpd-conf=<path_to_conf>
sets path to Apache conf.d directory

```
kpprc@kpprc-nagios:/etc/apache2/sites-enabled$ ls
000-default.conf
```

Installation (Create User And Group)

- sudo make install-groups-users
  - creates the nagios user and group

```
make install-groups-users
    - This adds the users and groups if they do not exist
```

- sudo usermod -a -G nagios www-data
  - www-data user is also added to the nagios group

Installation (Binaries and configs)

- sudo make install
  - installs the binary files, CGIs, and HTML files.
- sudo make install-daemoninit
  - installs the service or daemon files and also configures them to start on boot
- sudo make install-commandmode
  - installs and configures the external command file.
- sudo make install-config
  - installs the *SAMPLE* configuration files

Installation (Apache config files)

- sudo make install-webconf
  - installs the *Apache* web *server* configuration files

```
kpprc@kpprc-nagios:~/Downloads/nagioscore-nagios-4.4.2$ ls /etc/apache2/sites-enabled/
000-default.conf  nagios.conf
```

- sudo a2enmod rewrite
  - Enabling module rewrite
- sudo a2enmod cgi
  - Enabling module cgi.

Installation (Firewall and user account)

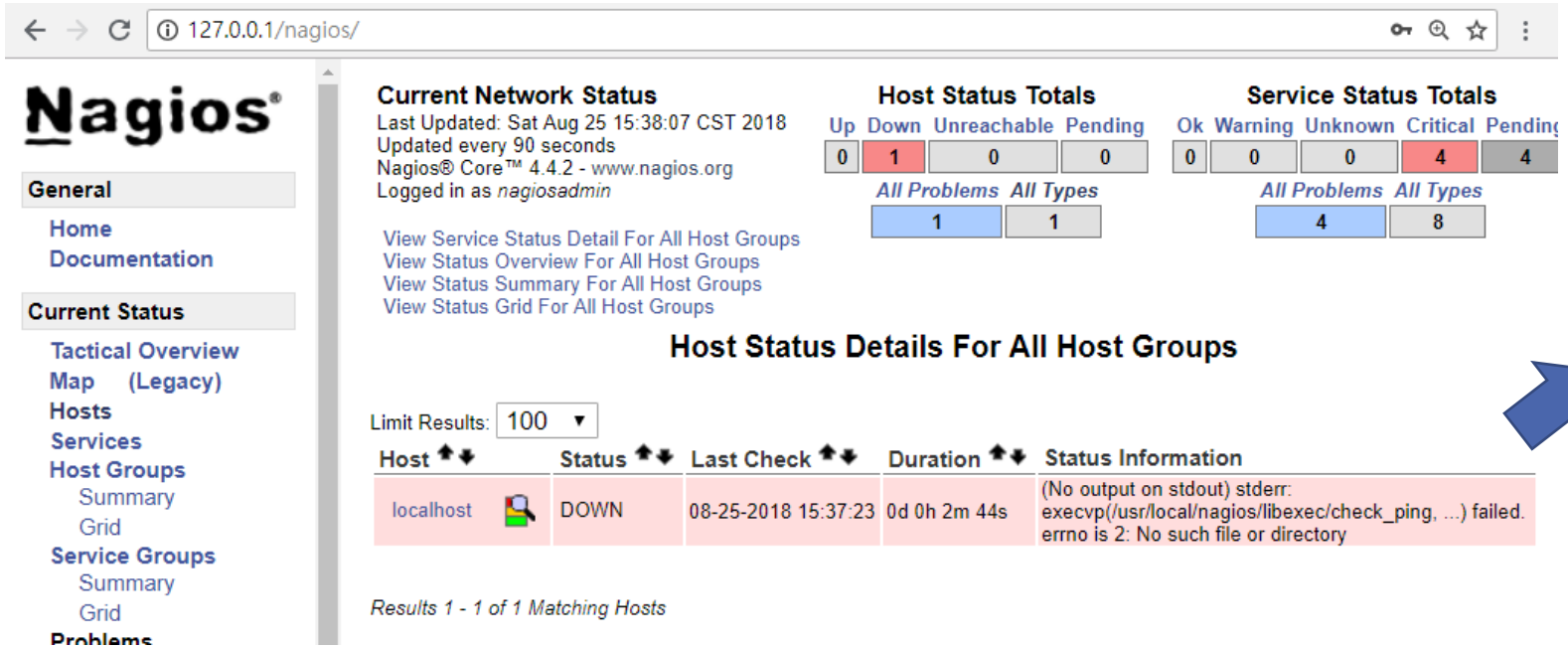
- sudo ufw allow Apache
  - Add port 80 for web interface access
- sudo ufw reload
- sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
  - create an Apache user account to be able to log into Nagios

Installation (Apache server)

- sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
  - create an Apache user account to be able to log into Nagios
  - Set your own login password
- sudo systemctl restart apache2.service
  - Restart web server
- sudo systemctl start nagios.service
  - Start Service / Daemon

Check Nagios Status

- http://127.0.0.1/nagios
  - Test Nagios status



Installing The Nagios Plugins

Nagios Plugin Installation

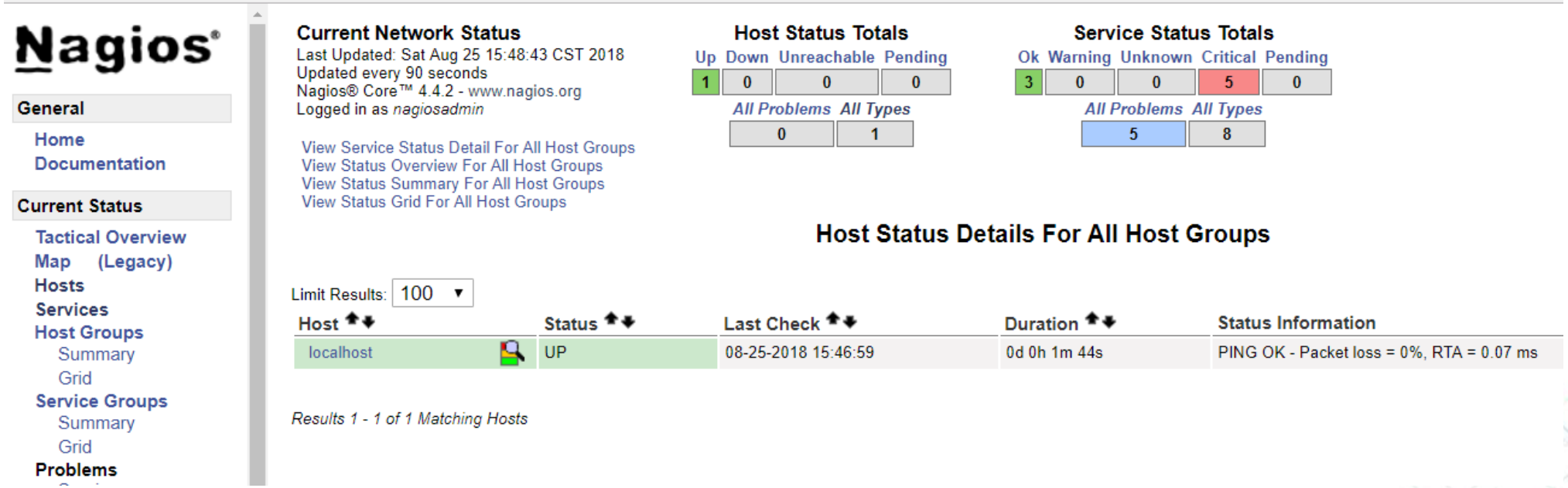
- sudo apt-get install -y autoconf gcc libc6 libmcrypt-dev make libssl-dev wget bc gawk dc build-essential snmp libnet-snmp-perl gettext
  - Prerequisites
- cd ~/Downloads/
- wget --no-check-certificate -O nagios-plugins.tar.gz https://github.com/nagios-plugins/nagios-plugins/archive/release-2.2.1.tar.gz
- tar zxf nagios-plugins.tar.gz

Nagios Plugin Installation (cont.)

- cd ~/Downloads/nagios-plugins-release-2.2.1
- sudo ./tools/setup
- sudo ./configure
- sudo make
- sudo make install

Check Nagios Plugin Status

- http://127.0.0.1/nagios

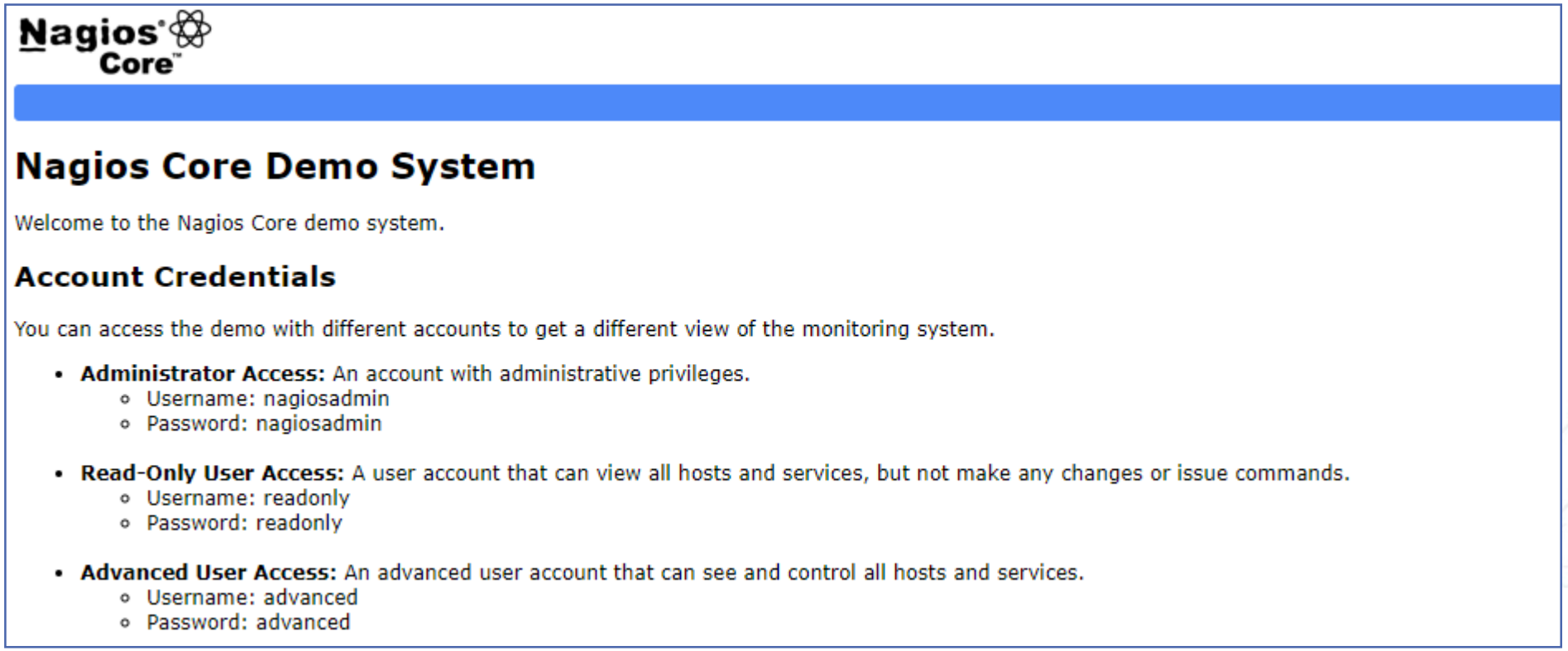


Nagios Service Commands

- sudo systemctl start nagios.service
- sudo systemctl stop nagios.service
- sudo systemctl restart nagios.service
- sudo systemctl status nagios.service

Nagios Online Demo

- http://nagioscore.demos.nagios.com/



Monitor HOST

- Host Definition
- define a physical server, workstation, device, etc

mandatory field

```
define host {
host_name                       host_name
max_check_attempts              #
check_period                    timeperiod_name
contacts                        contacts
contact_groups                  contact_groups
notification_interval           #
notification_period             timeperiod_name
…
}
```


https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/objectdefinitions.html#host

Linux HOST Template

```
define host{
        name                    linux-server         ; The name of this host template
        use                     generic-host         ; This template inherits other values from the generic-host template
        check_period            24x7                 ; By default, Linux hosts are checked round the clock
        check_interval          5                    ; Actively check the host every 5 minutes
        retry_interval          1                    ; Schedule host check retries at 1 minute intervals
        max_check_attempts      10                   ; Check each Linux host 10 times (max)
        check_command           check-host-alive ; Default command to check Linux hosts
        notification_period     workhours          ; Linux admins hate to be woken up, so we only notify during the day
                                                   ; Note that the notification_period variable is being overridden from
                                                   ; the value that is inherited from the generic-host template!
        notification_interval   120                ; Resend notifications every 2 hours
        notification_options    d,u,r              ; Only send notifications for specific host states
        contact_groups          admins             ; Notifications get sent to the admins by default
        register                0                  ; DONT REGISTER THIS DEFINITION - ITS NOT A REAL HOST, JUST A TEMPLATE!
        }
```

Host Definition Example

```
define host {
        use                     linux-server
        host_name               cacti
        alias                    cacti
        address                 163.28.129.100
        max_check_attempts      5
        check_period             24x7
        notification_interval    30
        notification_period      24x7
}
```

IP address (or FQDN)

Max_check_attempts:
define the number of times that Nagios will retry the host check command if it returns any state other than an OK state

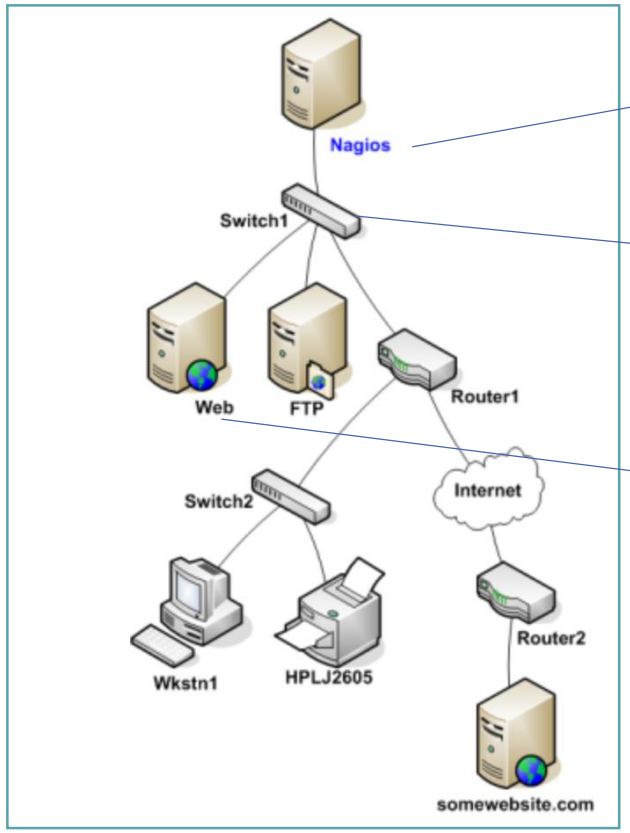
Lab1 - Let’s Start Monitoring

- Create HOSTs and Services

```
define host {
        use                     linux-server
        host_name               Suricata
        alias                   Suricata
        address                 10.0.2.6
        max_check_attempts      3
        notification_interval   30
        notification_period     24x7
}

define service{
        use generic-service
        host_name Suriata
        service_description SSH
        check_command check_ssh
}
```

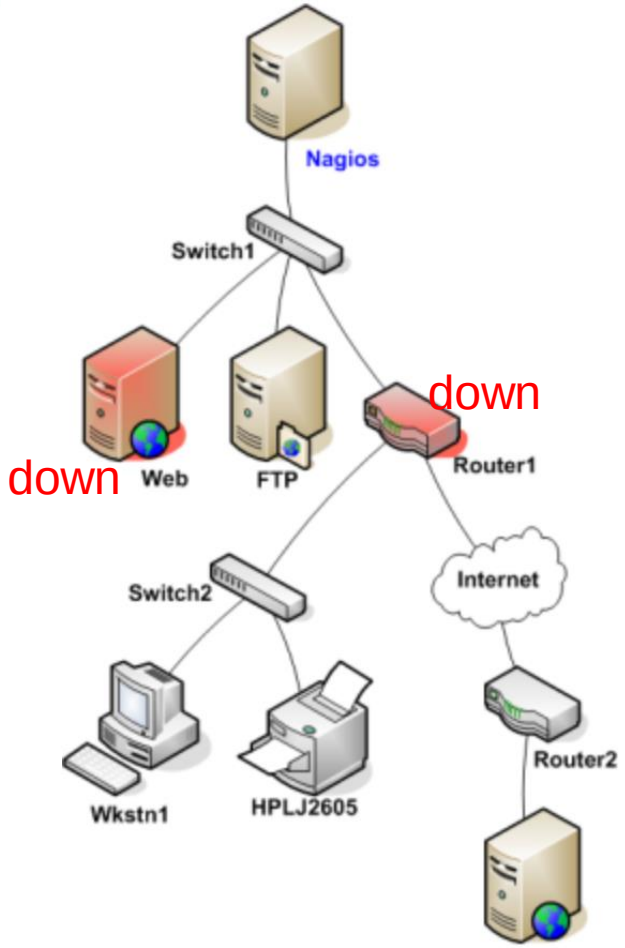
Status and Reachability of Network Hosts

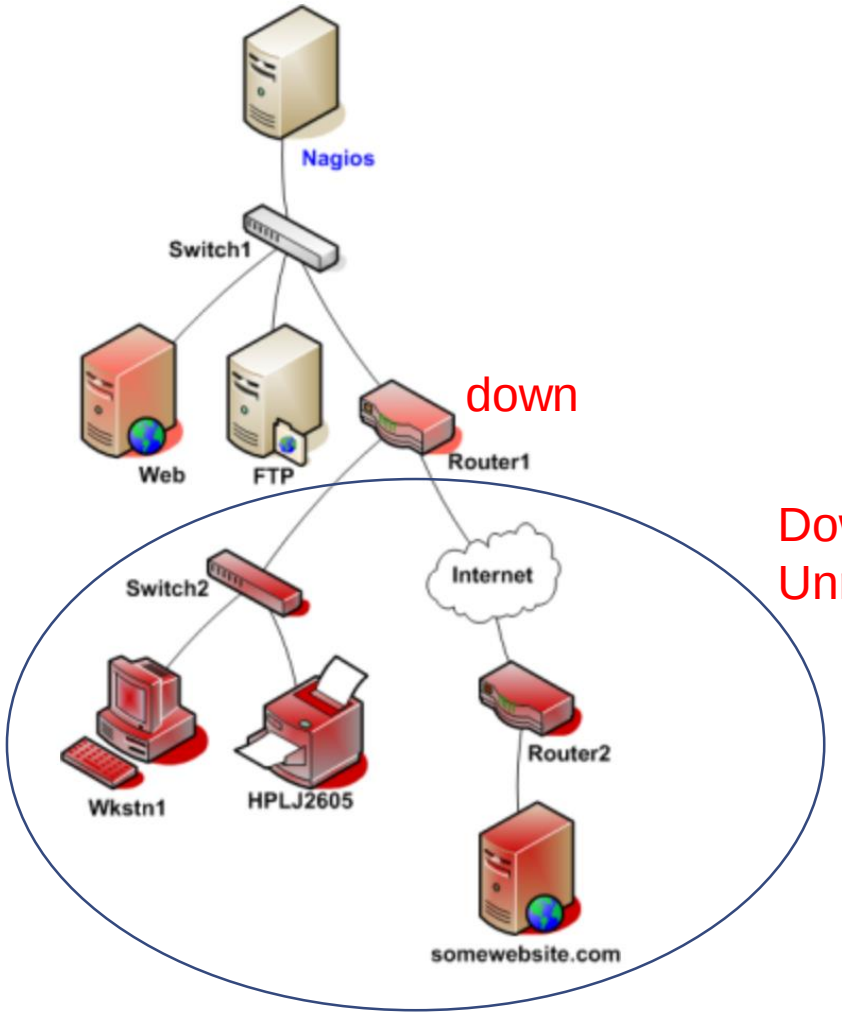
- Distinguish between DOWN and UNREACHABLE states
  - Defining Parent & Child Relationships




```
define host {
    host_name   Nagios
}

define host {
    host_name   Switch1
    parents     Nagios
}

define host {
    host_name   Web
    parents     Switch1
}

define host {
    host_name   FTP
    parents     Switch1
}
```

Unreachable State





Down or
Unreachable ?

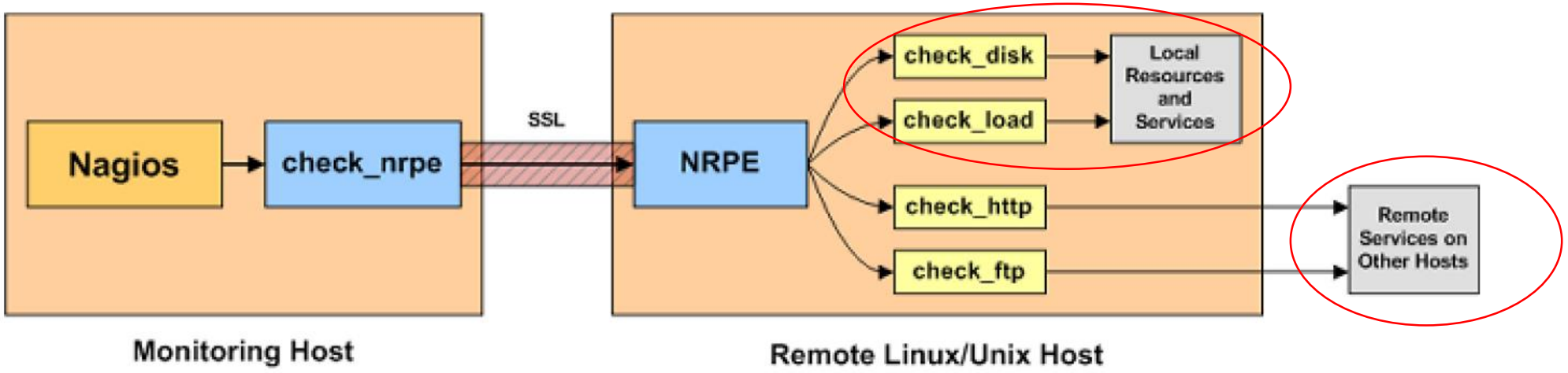
Nagios NRPE addon

Running Nagios plugins remotely

- Nagios Server
- Install NRPE Plugin

- The Remote host be monitored
- **Install NRPE Plugin**
- **Install NRPE Server**

NRPE addon (1/6)



IP: 10.0.2.5

IP: 10.0.2.6

```
nagios@kpprc-nagios:/usr/local/nagios/libexec$ ./check_nrpe -H 10.0.2.6
NRPE v3.2.1
```

```
nagios@kpprc-nagios:/usr/local/nagios/libexec$ ./check_nrpe -H 10.0.2.6 -c check_load
WARNING - load average: 0.07, 0.06, 0.11|load1=0.070;0.150;0.300;0; load5=0.060;0.100;0.250;0; load15=0.110;0.050;0.200;0;
```

NRPE addon (Monitoring HOST) (2/6)

- NRPE (Nagios Remote Plugin Executor)
- install the check_nrpe plugin on our Nagios server.
- Dependence: SSL headers and libraries
  - sudo apt install libssl-dev
- curl -L -O https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-3.2.1/nrpe-3.2.1.tar.gz
- /configure
- make check_nrpe (builds check_nrpe only)
- sudo make install-plugin

NRPE addon (Remote HOST) (3/6)

- OS Environment: Ubuntu 18.04.1 LTS
- sudo apt-get update
- sudo apt-get install nagios-nrpe-server nagios-plugins
- /etc/nagios/nrpe.cfg
  - allowed_hosts=127.0.0.1, 10.0.2.5 (<Your Monitoring Nagios Server IP>)
- Verify Connection (Server Side)
  - /usr/local/nagios/libexec/check_nrpe –H (Monitored Client IP)
- Configuration file
  - /etc/nagios/nrpe.cfg

NRPE addon (Remote HOST) (4/6)

- Remote Host Side
  - /etc/nagios/nrpe.cfg

```
command[check_users]=/usr/lib/nagios/plugins/check_users -w 5 -c 10
command[check_load]=/usr/lib/nagios/plugins/check_load -r -w .15,.10,.05 -c .30,.25,.20
command[check_hda1]=/usr/lib/nagios/plugins/check_disk -w 20% -c 10% -p /dev/hda1
command[check_zombie_procs]=/usr/lib/nagios/plugins/check_procs -w 5 -c 10 -s Z
command[check_total_procs]=/usr/lib/nagios/plugins/check_procs -w 150 -c 200
```

Commands we can call from monitoring sides

NRPE addon (Monitoring HOST) (5/6)

- Monitoring Nagios Host
  - /usr/local/nagios/etc/objects/commands.cfg
  - Add nrpe command definition

```
define command
{
 command_name check_nrpe command_line $USER1$/check_nrpe -H
$HOSTADDRESS$ -c $ARG1$
 }
```

NRPE addon (Monitoring HOST) (6/6)

- Define Service to monitor

```
define service{
    use generic-service
    host_name Suricata
    service_description CPU Load
    check_command check_nrpe!check_load
}
```

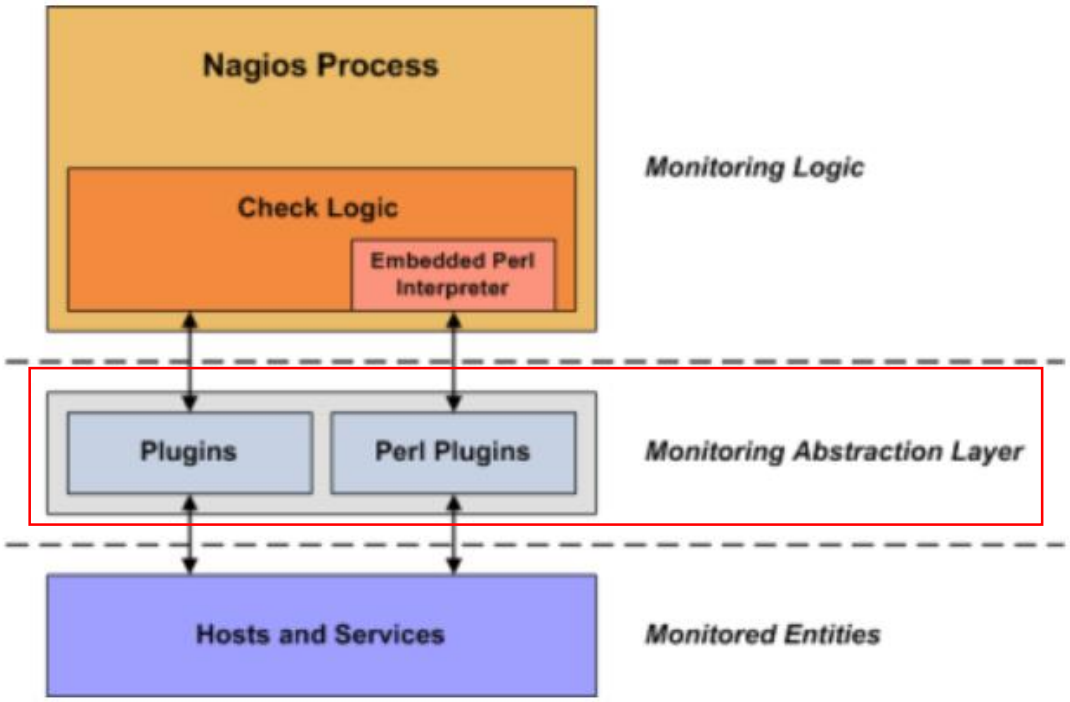
Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
Suricata	CHECK_LOAD	WARNING	08-28-2018 15:13:55	0d 0h 4m 8s	2/3	WARNING - load average: 0.07, 0.11, 0.07
	CHECK_SDA1	OK	08-28-2018 15:13:46	0d 0h 2m 17s	1/3	DISK OK - free space: /var/tmp 19411 MB (67% inode=85%):
localhost	Current Load	OK	08-28-2018 15:14:15	0d 5h 33m 35s	1/4	OK - load average: 0.34, 0.46, 0.44
	Current Users	OK	08-28-2018 15:15:22	2d 23h 28m 26s	1/4	USERS OK - 1 users currently logged in
	FTP	OK	08-28-2018 15:13:30	0d 0h 7m 33s	1/4	FTP OK - 0.005 second response time on 127.0.0.1 port 21 [220 (vsFTPd 3.0.3)]
	HTTP	OK	08-28-2018 15:12:35	2d 23h 27m 49s	1/4	HTTP OK: HTTP/1.1 200 OK - 11192 bytes in 0.001 second response time
	PING	OK	08-28-2018 15:12:09	2d 23h 27m 11s	1/4	PING OK - Packet loss = 0%, RTA = 0.04 ms
	Root Partition	OK	08-28-2018 15:14:49	2d 23h 26m 34s	1/4	DISK OK - free space: / 18763 MB (65.70% inode=85%):
	SSH	OK	08-28-2018 15:10:55	2d 23h 25m 56s	1/4	SSH OK - OpenSSH_7.6p1 Ubuntu-4 (protocol 2.0)
	Swap Usage	OK	08-28-2018 15:12:02	2d 23h 25m 19s	1/4	SWAP OK - 32% free (444 MB out of 1425 MB)
	Total Processes	OK	08-28-2018 15:13:09	2d 23h 24m 41s	1/4	PROCS OK: 46 processes with STATE = RSZDT

Nagios Plugins

Nagios Plugins Overview

- Plugins
  - Compiled executables or scripts
  - Scripts: Perl scripts, shell scripts, Python, PHP, Ruby, etc.
- Abstraction Layer



Nagios Core Plugin API

- Plugin Return Code

Return Code	Service State	Host State
0	OK	UP
1	WARNING	UP or DOWN/UNREACHABLE*
2	CRITICAL	DOWN/UNREACHABLE
3	UNKNOWN	DOWN/UNREACHABLE

https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/pluginapi.html

Simple bash plugin example

```
#!/bin/bash

filename=$1

#first check if the file exists. this is the first and most basic check with which you should begin
if [ ! -e $filename ]; then
    echo "CRITICAL status - file $filename doesn't exist"
    exit 2 #the file doesn't even exist

#if the previous condition passes (file exists) then next check if it is readable
elif [ ! -r $filename ]; then
    echo "WARNING status - file $filename is not readable."
    exit 1 #returns warning status because this state is better than having no file at all

#if the previous condition passes, check if it is a regular file and not a directory or device file
elif [ ! -f $filename ]; then
    echo "UNKNOWN status - file $filename is not a file."
    exit 3 #returns unknown status

#if all of the above checks pass then it's ok
else
    echo "OK status - file is OK"
    exit 0 #Return OK status
fi
```

Check SSL Certificate expiration example

```
#!/bin/bash
SitePort=$1
OutDate=$(echo | openssl s_client -connect ${SitePort}:443 2>/dev/null | openssl x509 -noout -enddate | awk -F'='
'{print $2}')
OutputDate=$(date -d "${OutDate}" "+%s")
SystemDate=$(date '+%s')
DiffTime=$(expr "${OutputDate}" - "${SystemDate}")
if [ "${DiffTime}" -ge 864000 ]
then
echo "OK - Certificate will expire on ${OutDate}"
exit 0
elif [ "${DiffTime}" -le 863999 -a "${DiffTime}" -ge 5 ]
then
echo "WARNING - Certificate is going to expire in less then 10 days on ( ${OutDate} ). Please Renew!"
exit 1
elif [ "${DiffTime}" -le 4 ]
then
echo "CRITICAL - Certificate has EXPIRED on ( ${OutDate} )! Please replace!"
exit 2
else
echo "UNKNOWN - ERROR! Certificate NOT found!"
exit 3
fi
```

Nagios Plugin實作 (1/2)

- Nagios plugin directory: /usr/local/nagios/libexec
- Define Nagios command
  - /usr/local/nagios/etc/objects/command.cfg

```
define command{
        command_name    check_ssl
        command_line    $USER1$/check_ssl $HOSTADDRESS$
}
```

Sets $USER1$ to be the path to the plugins
$USER1$=/usr/local/nagios/libexec

Address of the host. This value is taken from the address directive in the host definition.

https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/macrolist.html#hostname

Nagios Plugin實作 (2/2)

- Define your plugin as a service in your Nagios configuration
  - /usr/local/nagios/etc/servers

```
define host {
        use                     linux-server
        host_name               cacti2
        alias                   cacti2
        address                 cacti2.kpprc.edu.tw
        max_check_attempts      5
        check_period            24x7
        notification_interval   30
        notification_period     24x7
}
```

```
define service{
        use generic-service
        host_name cacti2
        service_description SSL_Expiration_Check
        check_command check_ssl
}
```

Check_ssl plugin實際運作

Status	Last Check	Duration	Attempt	Status Information
OK	06-04-2018 13:57:36	1d 23h 33m 38s	1/3	OK - Certificate will expire on Jul 15 02:33:33 2018 GMT
OK	06-04-2018 13:54:44	5d 21h 2m 31s	1/3	OK - load average: 0.00, 0.00, 0.00
CRITICAL	06-04-2018 13:53:42	1d 23h 31m 30s	3/3	CRITICAL - Certificate has EXPIRED on (May 27 06:20:04 2018 GMT)! Please replace!

NRPE Plugin (1/5)

- Example: check_volume plugin
- Download URL
  - https://raw.githubusercontent.com/jonschipp/nagios-plugins/master/check_volume.sh

```
frank@suricata:/usr/lib/nagios/plugins$ ./check_volume.sh -v /var/data/

Check the capacity of a volume using df.

     Options:
        -v          Specify volume as mountpoint
        -c          Critical threshold as an int (0-100)
        -w          Warning threshold as an int (0-100)
        -s          Skip threshold checks
```

```
frank@suricata:/usr/lib/nagios/plugins$ ./check_volume.sh -v /var/data/ -c 95 -w 80
/var/data/ is at 44% capacity, 2.0T of 4.5T
```

NRPE Plugin (2/5)

- Allow Nagios server to run commands on the client by adding it to the allowed_hosts entry in ***/etc/nagios/nrpe.cfg***

```
# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostnames
# that are allowed to talk to the NRPE daemon. Network addresses with a bit mask
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently
# supported.
#
# Note: The daemon only does rudimentary checking of the client's IP
# address.  I would highly recommend adding entries in your /etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

allowed_hosts=127.0.0.1, 163.28.129.121
```

NRPE Plugin (3/5)

- 1. Check include the necessary configuration directory

```
frank@suricata:/etc/nagios/nrpe.d$ cat /etc/nagios/nrpe.cfg | grep include | grep -v ^#
include=/etc/nagios/nrpe_local.cfg
include_dir=/etc/nagios/nrpe.d/
```

- 2. Define the checks on /etc/nagios/nrpe_local.cfg or /etc/Nagios/nrpe.d**/<cfg-name>**

```
frank@suricata:/etc/nagios/nrpe.d$ ls
check_volume.cfg
frank@suricata:/etc/nagios/nrpe.d$ sudo vim check_volume.cfg
```

```
#custom
    command[check_volume]=/usr/lib/nagios/plugins/check_volume.sh -v /var/data -c 95 -w 85
~
```

Critical threshold

warning threshold

Mount point

- 3. Restart Nagios nrpe server service
  - sudo systemctl restart nagios-nrpe-server.service

NRPE Plugin (4/5)

- Test on the Nagios server side

```
kpprc@kpprc-nagios:~$ /usr/local/nagios/libexec/check_nrpe -H 163.28.129.32 -c check_volume
/var/data is at 44% capacity, 2.0T of 4.5T
```

- Set the NRPE Check on the Server Configuration Files
  - */usr/local/nagios/etc/objects/commands.cfg*:

```
# check_npre command definition
define command{
        command_name check_nrpe
        command_line $USER1$/check_nrpe -H $HOSTADDRESS$ -c $ARG1$
}
```

NRPE Plugin (5/5)

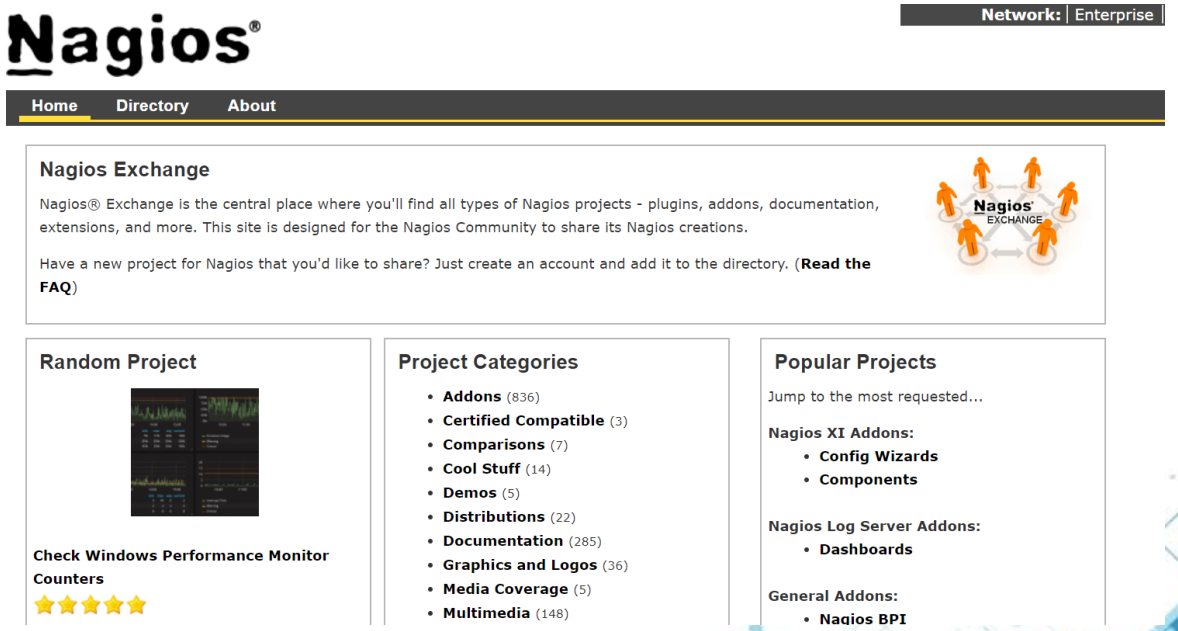
- Add the plugin into the service

```
define service{
        use generic-service
        host_name Suricata
        service_description nrpe_check_volume
        check_command check_nrpe!check_volume
}
```

- Check the configuration
  - */usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg*
- Restart the Nagios server
  - *sudo systemctl restart nagios*

Obtaining Plugins

- Nagios Plugins Project
  - https://github.com/nagios-plugins/nagios-plugins
- Nagios Downloads Page
  - https://www.nagios.org/download/
- Nagios Exchange
  - 尋找好用的Nagios Plugin
  - https://exchange.nagios.org



Google Calendar API Integration with Nagios

Gcalcli Introduction

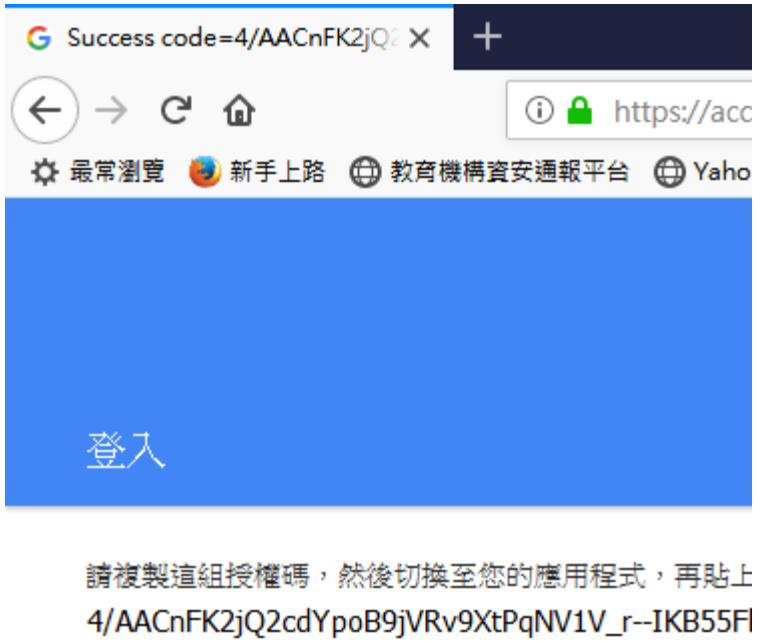
- Gcalcli
  - Google Calendar Command Line Interface
  - Python application
  - Easily to access google calendar
- Install gcalcli
  - sudo apt-get update
  - sudo apt-get install gcalcli

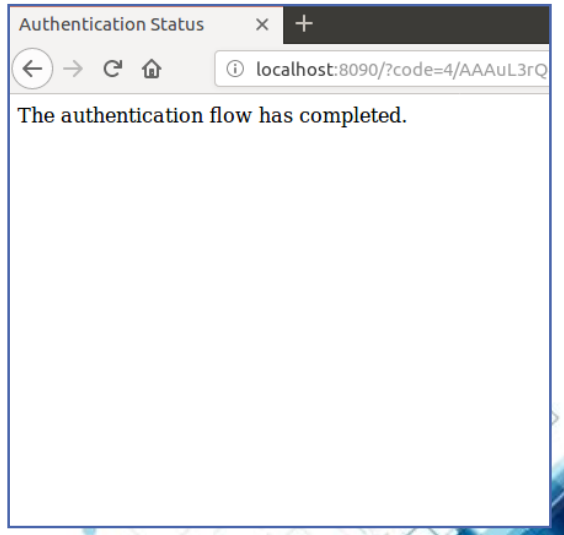
Gcalci Access(1/2)

- Switch to nagios account first
- gcalcli list --noauth_local_webserver

```
kpprc@kpprc-nagios:~$ gcalcli list  --noauth_local_webserver
Go to the following link in your browser:

    https://accounts.google.com/o/oauth2/auth?scope=https%3A%2F%2Fw
r&redirect_uri=urn%3Aietf%3Awg%3Aoauth%3A2.0%3Aoob&response_type=co
```

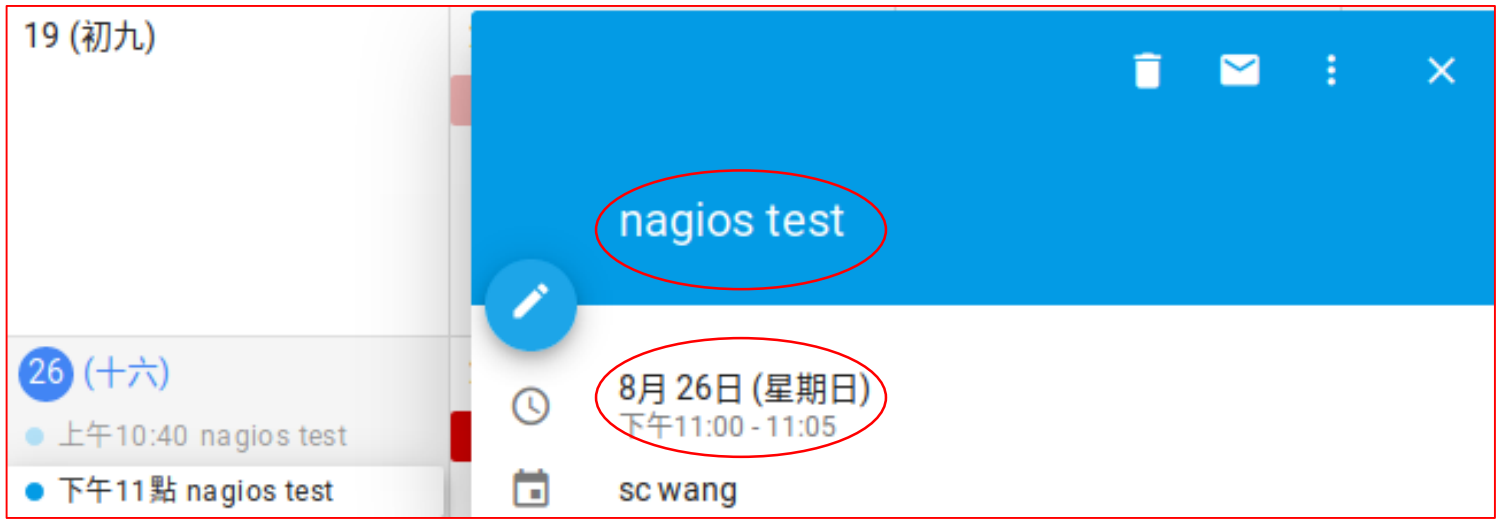
Gcalci Access(2/2)





Gcalci command test

- Run gcalcli command test firstly
  - gcalcli --calendar scwang@g-mail.nsysu.edu.tw add --title 'nagios test' --when '8/26/2018 23:00:00' --duration 5



Gcalci Integration with Nagios (1/3)

- /usr/local/nagios/etc/objects/command.cfg
- Add notify host commands

```
define command{
        command_name    notify-host-by-gcalcli
        command_line        /usr/bin/gcalcli --calendar 'scwang@g-mail.nsysu.edu.tw' add --title
'$NOTIFICATIONTYPE$ Host Alert: $HOSTALIAS$' --where 'KPPRC' --when '$LONGDATETIME$' --
duration 5 --description '**Nagios** Type: $NOTIFICATIONTYPE$ IP:$HOSTADDRESS$' --reminder '5m
email'

}
```

Note: the command path and syntax maybe different for different versions

Gcalci Integration with Nagios (2/3)

- /usr/local/nagios/etc/objects/command.cfg
- Add notify service commands

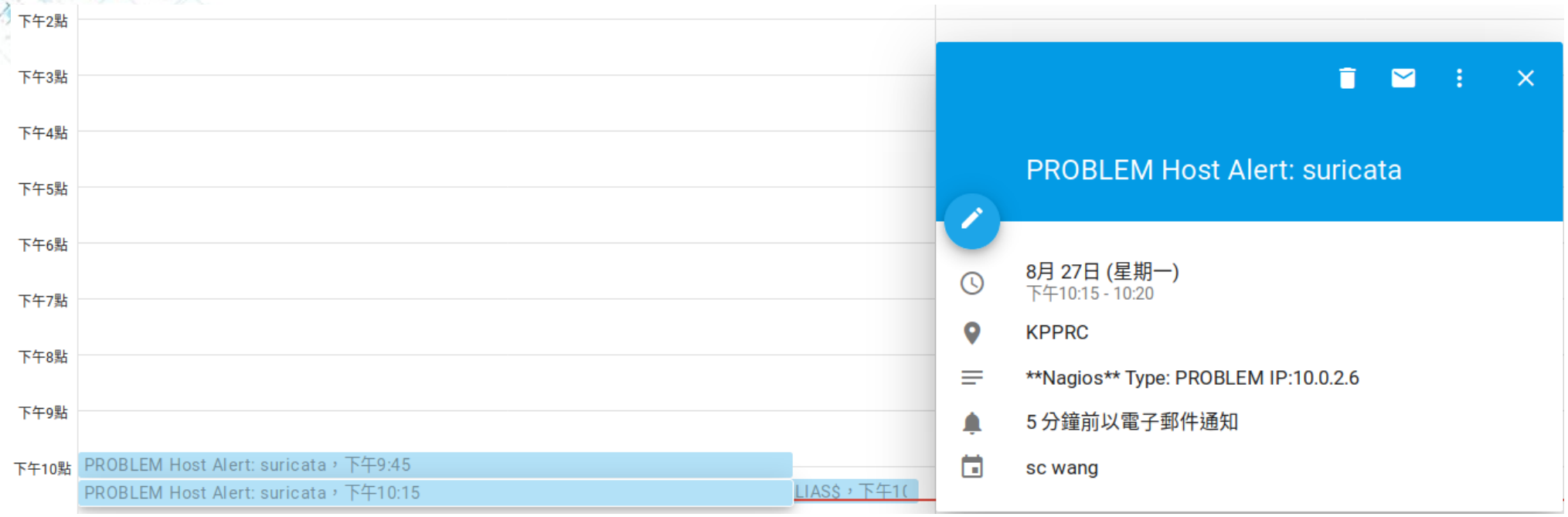
```
define command{
        command_name    notify-service-by-gcalcli
        command_line    /usr/bin/gcalcli --calendar 'scwang@g-mail.nsysu.edu.tw' --title
'$NOTIFICATIONTYPE$ Service Alert: $HOSTALIAS$ - $SERVICEDESC$' --where ‘KPPRC' --when
'$LONGDATETIME$' --duration 5 --description '**Nagios** Type: $NOTIFICATIONTYPE$
IP:$HOSTADDRESS$' --reminder '5m email' add

}
```

Gcalci Integration with Nagios (3/3)

- /usr/local/nagios/etc/objects/contact.cfg
- Change email address and notification commands

```
define contact {

    contact_name                    nagiosadmin             ; Short name of user
    use                             generic-contact         ; Inherit default values from generic-contact template
    alias                           Nagios Admin            ; Full name of user
    email                           scwang@g-mail.nsysu.edu.tw ; <<*****YOUR EMAIL  ADDRESS ******
    service_notification_commands       notify-service-by-gcalcli
    host_notification_commands          notify-host-by-gcalcli
}
```

Check the results




```
kpprc@kpprc-nagios:/usr/local/nagios/etc/objects$ gcalcli search suricata
2018-08-27  12:46朝    PROBLEM Host Alert: Suricata
2018-08-27   9:45暮    PROBLEM Host Alert: suricata
2018-08-27  10:15暮    PROBLEM Host Alert: suricata
```

Event Handlers

- Optional commands
- Run when state changes
- Try to fix problems before anyone is notified
- Restarting a failed service
- Logging event information to a database
- Example
  - https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/eventhandlers.html

Lab2

- Write your own Event Handler
- Linux vsftpd as example
- You may need to add sudo command first

Nagios log message

- /usr/local/nagios/var/nagios.log

```
kpprc@kpprc-nagios:/usr/local/nagios/var$ ll
total 1400
drwxrwxr-x 5 nagios nagios    4096  六   12 15:25 ./
drwxr-xr-x 9 root   root      4096  五   24 15:51 ../
drwxrwxr-x 2 nagios nagios    4096  六   11 23:59 archives/
-rw------- 1 nagios nagios  275354  六   12 15:25 nagios.debug
-rw------- 1 nagios nagios 1000068  六   12 15:20 nagios.debug.old
-rw------- 1 root   root     16384  六   12 15:23 .nagios.debug.swp
-rw-r--r-- 1 nagios nagios   21986  六   12 14:47 nagios.log
-rw-r--r-- 1 nagios nagios   21367  六   12 14:43 objects.cache
-rw------- 1 nagios nagios   25888  六   12 14:43 retention.dat
drwxrwsr-x 2 nagios nagcmd    4096  六   12 14:43 rw/
drwxr-xr-x 3 root   root      4096  五   23 22:16 spool/
-rw-rw-r-- 1 nagios nagios   26817  六   12 15:25 status.dat
```

Nagios log message (cont.)

- More readable format
  - cat /usr/local/nagios/var/nagios.log | perl -pe 's/(\d+)/localtime($1)/e'

```
[1528786056] wproc: Core Worker 14172: job 4 (pid=14357) timed out. Killing it
[1528786056] wproc: NOTIFY job 4 from worker Core Worker 14172 timed out after 60.01s
[1528786056] wproc:   host=kpprc_194; service=SSL expiration check; contact=nagiosadmin
[1528786056] wproc:   early_timeout=1; exited_ok=0; wait_status=0; error_code=62;
[1528786056] wproc:   stderr line 01: WARNING:root:This function, oauth2client.tools.run(),
```

```
[Tue Jun 12 14:43:58 2018] wproc: Registry request: name=Core Worker 14172;pid=14172
[Tue Jun 12 14:43:58 2018] wproc: Registry request: name=Core Worker 14171;pid=14171
[Tue Jun 12 14:43:58 2018] wproc: Registry request: name=Core Worker 14174;pid=14174
[Tue Jun 12 14:43:58 2018] wproc: Registry request: name=Core Worker 14173;pid=14173
[Tue Jun 12 14:43:58 2018] Successfully launched command file worker with pid 14176
[Tue Jun 12 14:44:36 2018] SERVICE ALERT: kpprc_194;SSL expiration check;CRITICAL;SOFT;1
```

Using The Nagiostats Utility

https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/nagiostats.html

Nagiostats utility

- Tuning performance is helpful
- MRTG Integration
  - --mrtg and –data arguments
- /usr/local/nagios/bin/nagiostats –c <config file>

Nagiostats utility

```
nagios@kpprc-nagios:/home/kpprc$ /usr/local/nagios/bin/nagiostats -c /usr/local/nagios/etc/nagios.cfg

Nagios Stats 4.3.4
Copyright (c) 2003-2008 Ethan Galstad (www.nagios.org)
Last Modified: 2017-08-24
License: GPL

CURRENT STATUS DATA
------------------------------------------------------
Status File:                            /usr/local/nagios/var/status.dat
Status File Age:                        0d 0h 0m 2s
Status File Version:                    4.3.4

Program Running Time:                   0d 11h 3m 21s
Nagios PID:                             25620

Total Services:                         14
Services Checked:                       14
Services Scheduled:                     14
Services Actively Checked:              14
Services Passively Checked:             0
Total Service State Change:             0.000 / 0.000 / 0.000 %
Active Service Latency:                 0.000 / 0.002 / 0.001 sec
Active Service Execution Time:          0.003 / 4.064 / 0.328 sec
Active Service State Change:            0.000 / 0.000 / 0.000 %
Active Services Last 1/5/15/60 min:     5 / 11 / 14 / 14
Passive Service Latency:                0.000 / 0.000 / 0.000 sec
Passive Service State Change:           0.000 / 0.000 / 0.000 %
Passive Services Last 1/5/15/60 min:    0 / 0 / 0 / 0
Services Ok/Warn/Unk/Crit:              13 / 0 / 0 / 1
```

MRTG Integration (1/2)

- Copy Nagios mrtg sample config
  - sudo cp nagios-4.4.2/sample-config/nagios.cfg /usr/local/nagios/etc/
- Create a folder for graphs and files
  - mkdir -p /usr/local/nagios/share/mrtg
- Configure MRTG
  - vim /usr/local/nagios/etc/mrtg.cfg
    - WorkDir: /usr/local/nagios/share/mrtg
- Initial run
  - env LANG=C /usr/bin/mrtg /usr/local/nagios/etc/mrtg.cfg

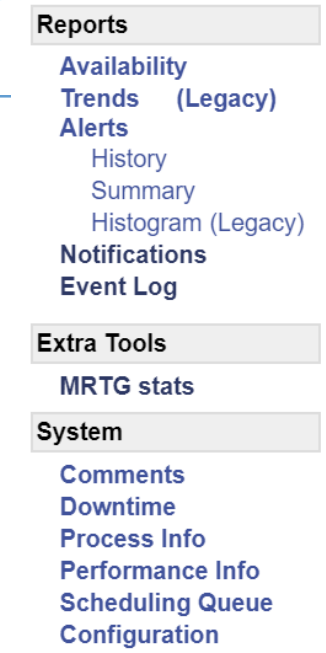
MRTG Integration (2/2)

- Create index.html page
  - /usr/bin/indexmaker /usr/local/nagios/etc/mrtg.cfg --output=/usr/local/nagios/share/mrtg/index.html
- Add into crontab
  - vim /etc/cron.d/mrtg-nagios
  - */5 * * * * env LANG=C /usr/bin/mrtg /usr/local/nagios/etc/mrtg.cfg
- Check result
  - http://<Nagios-server-ip>/nagios/mrtg

Add MRTG Links into Nagios menu

/usr/local/nagios/share/side.php

```
<div class="navsectiontitle">Extra Tools</div>
    <div class="navsectionlinks">
        <ul class="navsectionlinks">
            <li><a href="/nagios/mrtg" target="<?php echo $link_target;?>">MRTG stats</a></li>
        </ul>
    </div>
</div>
```



Nagios®

General
- Home
- Documentation

Current Status
- Tactical Overview
- Map (Legacy)
- Hosts
- Services
- Host Groups
  - Summary
  - Grid
- Service Groups
  - Summary
  - Grid
- Problems
  - Services (Unhandled)
  - Hosts (Unhandled)
  - Network Outages

Quick Search:

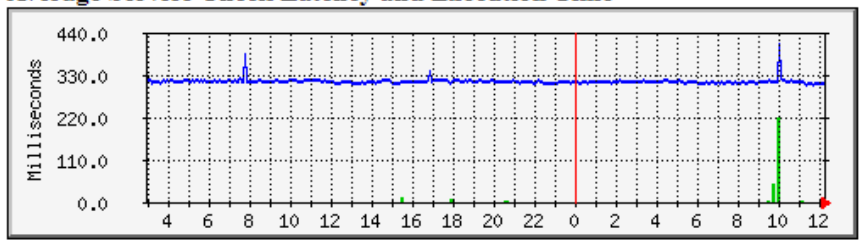
Reports
- Availability
- Trends (Legacy)
- Alerts
  - History
  - Summary
  - Histogram (Legacy)
- Notifications
- Event Log

Extra Tools
- MRTG stats

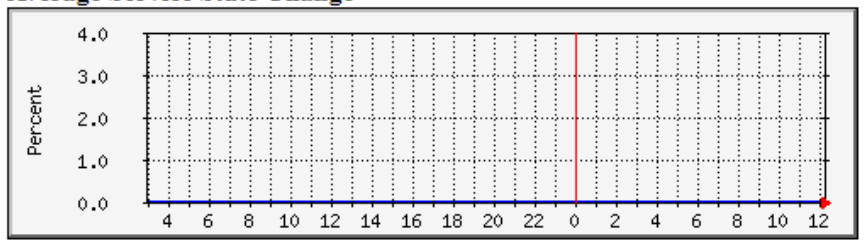
System
- Comments
- Downtime
- Process Info
- Performance Info
- Scheduling Queue
- Configuration

MRTG Index Page

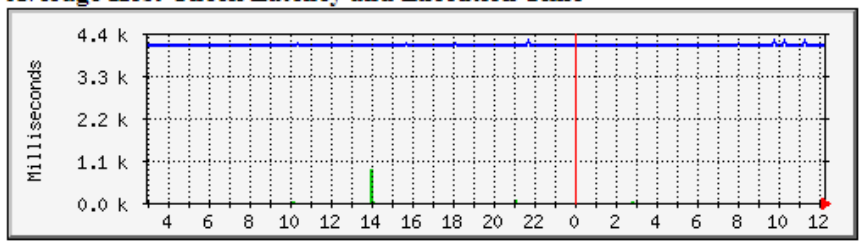
Average Service Check Latency and Execution Time



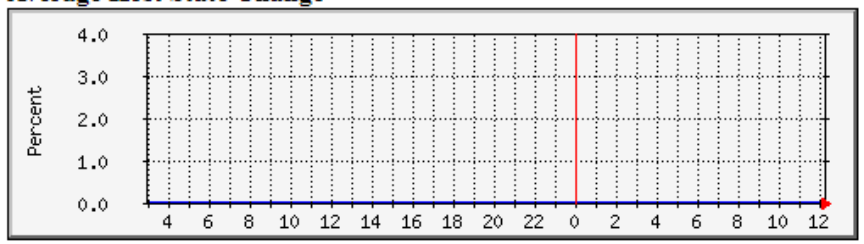
Average Service State Change



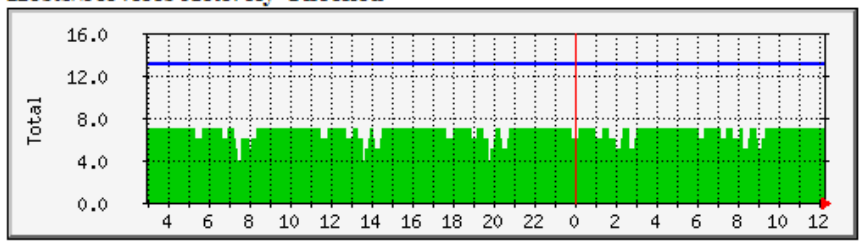
Average Host Check Latency and Execution Time



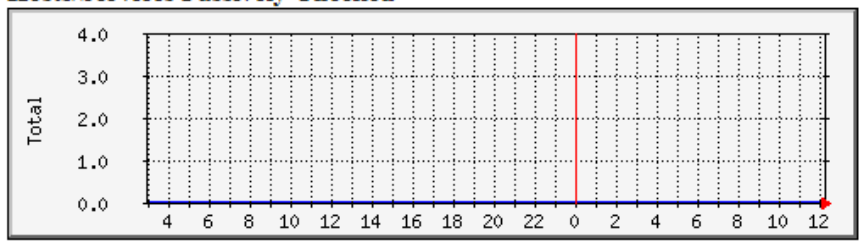
Average Host State Change



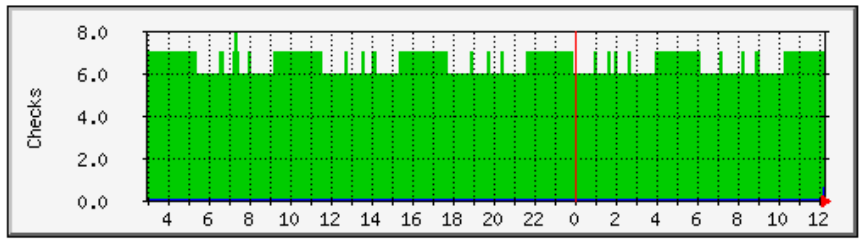
Hosts/Services Actively Checked



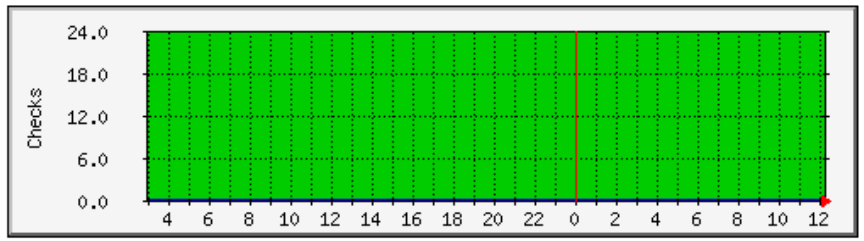
Hosts/Services Passively Checked



Active Host Checks



Active Service Checks



MRTG REFERENCE

- http://www.ubuntugeek.com/install-and-configure-mrtg-on-ubuntu-16-04-server.html
- Nagiosstats command
- https://nazarenolatella.myblog.it/tag/mrtg/
- https://raymii.org/s/tutorials/Nagios_Core_4_Installation_on_Ubuntu_12.04.html
- http://gree2.github.io/ubuntu/2015/07/19/install-mrtg-on-ubuntu

Nagios Plugin Reference

- https://nagios-plugins.org/doc/guidelines.html
- https://www.networkworld.com/article/2224083/opensource-subnet/how-to-extend-nagios-for-custom-monitoring.html
- https://www.howtoforge.com/tutorial/write-a-custom-nagios-check-plugin/

Nagios Core Reference

- Nagios Core Documentation
  - https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/toc.html
- Nagios github page
  - https://github.com/NagiosEnterprises/nagioscore
- Nagios NRPE
  - https://assets.nagios.com/downloads/nagioscore/docs/nrpe/NRPE.pdf