WSEAS TRANSACTIONS on COMPUTERS
DOI: 10.37394/23205.2020.19.31

Using Nagios on a Raspberry Pi to monitor a network with emphasis on security

ANTONIOS ANDREATOS, NIKOLAOS CHATZIPANTOU
Department of Aeronautical Sciences
Hellenic Air Force Academy
Dekeleia, Attica 13671
GREECE

Abstract: - The objective of this paper is to present an advanced use of Nagios core on Raspberry Pi to monitor a network. Nagios is a popular network monitoring software. Raspberry Pi is a tiny, inexpensive but powerful computer board with many applications. In this work the free version of Nagios has been installed on a Raspberry Pi in order to minimise cost. Bash scripts automating the complex processes of installing Nagios server software on Raspberry Pi, as well as Nagios client software on Linux hosts have been developed. In order to elevate the security level of Network monitoring, a triple set of Raspberry Pi Nagios servers has been proposed; each Raspberry Pi monitors hosts and servers, as well as the other two Nagios servers (triple modular redundancy). Custom scripts providing useful information about monitored hosts, such as their operating system, hardware and networking. Finally, a special script examining and rating the security level of Apache web servers has been developed and incorporated into the network monitoring process.

Key-Words: - Network monitoring, Raspberry Pi, Nagios Server, NRPE, Apache Web Server, plugins, custom scripts, triple modular redundancy.

Received: May 29, 2020. Revised: November 8, 2020. Accepted: December 7, 2020. Published: December 31, 2020.

1 Introduction

1.1 About network monitoring

Network monitoring is the use of a system that constantly monitors a computer network for slow or failing components, improper bandwidth utilization and other anomalies, and notifies the network administrator (via email, SMS or other alarms) in case of outages or other trouble. Effective network monitoring allows organisations to quickly detect application, service or process problems, and take action to eliminate downtime for application users. Network monitoring is part of network management (1).

All parts of a network such as servers, routers, switches, virtual machines, printers, etc., are being monitored every minute to ensure their correct operation and availability. The benefits of continuous network monitoring are:

- Detects all server and network problems timely.
- Finds the causes of failures.
- Reduces maintenance costs.
- Detects performance issues.
- Facilitates infrastructure update.
- Helps to automatically correct problems as soon as they are detected.
- Ensures that servers, services and applications of the network are seamlessly running.

1.2 About Nagios

Nagios is one of the best network monitoring software on the market (2). Nagios offers flexibility because it supports both agent-based and agentless monitoring.

Nagios provides tools for monitoring of applications, including Linux applications, UNIX applications, Windows applications and Web applications. Nagios server uses a web interface to display Network monitoring information in a user-friendly way (Figure 1).

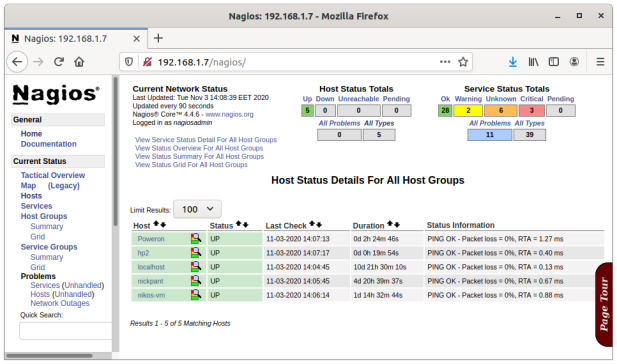

Figure 1: Nagios web interface

Nagios software is offered in two versions: the commercial version is called Nagios XI and offers additional features, while the free version is called Nagios core (3). In this work the Nagios core is used.

1.3 Client-server architecture

Nagios uses a client-server architecture. Nagios core software is installed on the Nagios server (a Raspberry Pi in this work).

The Nagios agent which run checks on remote machines is the Nagios Remote Plugin Executor, commonly known as NRPE. NRPE allows the Nagios server to run plugins on other machines remotely (Figure 2). Via NRPE remote machine metrics such as disk usage and CPU load can be monitored (4). For Windows machines the NSClient++ is used (5).

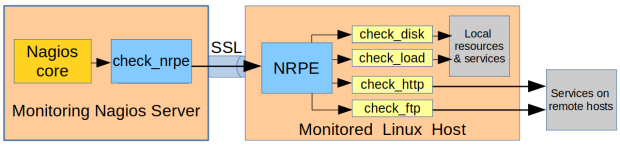

Figure 2: Nagios client-server architecture

In addition, the SNMP protocol must be installed and configured on monitored devices.

1.4 Configuring Nagios

After installation, configuration follows. Nagios configuration is a complicated task and depends on the number and type of monitored hosts. The Nagios server must have specific information for all monitored devices.

For each client the Nagios server needs a configuration file containing an alias, IP address, group it belongs to, and the desired services to be monitored (including services defined in custom plugins). Figure 3 shows custom services definitions in a configuration file of a monitored host from our work. An example host configuration file can be found in (4).

The path of the client configuration files must be added to the Nagios server main configuration file (nagios.cfg). In addition, the commands that will be used for monitoring must be declared in a special configuration file (commands.cfg).

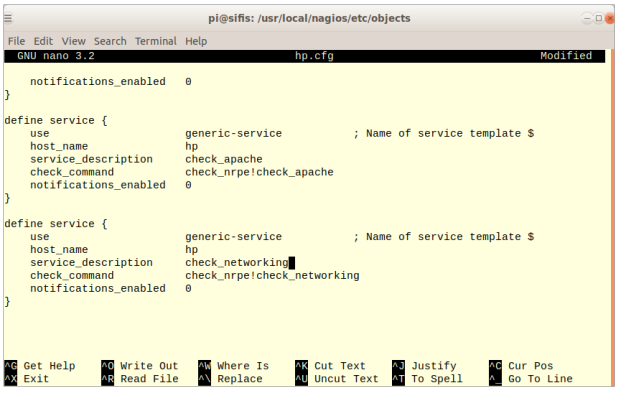

Figure 3: View of configuration file of a monitored host

On the other hand, for each monitored device using NRPE, a configuration file with the IP addresses of the monitoring server(s), as well as the monitoring services that will be used for this specific device must be edited (nrpe.cfg). All these services will be listed in the web interface of the Nagios server (Figure 4).

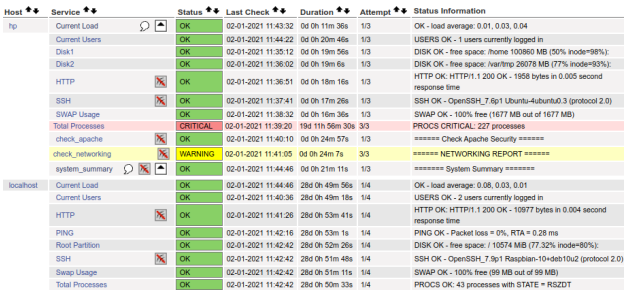

Figure 4: Services running on monitored hosts

1.5 Nagios Plugins

Nagios accepts 'plugins', i.e., compiled executables or scripts (usually in Perl), which extend its functionality to monitor servers and hosts. Plugins help Nagios server to monitor databases, operating systems, applications, network equipment, protocols, etc. (6). There are three types of Nagios plugins:

1. Official Nagios Plugins. More than 50 official Nagios Plugins, developed and maintained by the official Nagios Plugins Team, are available.
2. Community Plugins. There are over 5000 third party Nagios plugins that have been developed by hundreds of Nagios community members.
3. Custom Plugins. Users can also write their own custom plugins. There are certain guidelines that must be followed in order to write custom plugins.

Figure 5 shows a view of the plugins folder in a monitored host. There are 71 official plugins plus three custom plugins.

1.6 Raspberry Pi

Raspberry Pi is a tiny, low-cost yet powerful computer board, based on a 'system on a chip', first released in February 2012 (7). Figure 6 shows a Raspberry Pi 4 Model B (2019).

To date the Raspberry Pi Foundation has produced four generations of boards with several models (8) which cover a wide range of open-source projects (9) (Figure 7). Raspberry Pi can run several operating systems (OS) which are offered as ready-made images for downloading (10), with preinstalled software which converts Pi to a Print Server, Media Server, Game Sever, LAMP Web Server with WordPress, etc. In this work the Raspbian operating system has been

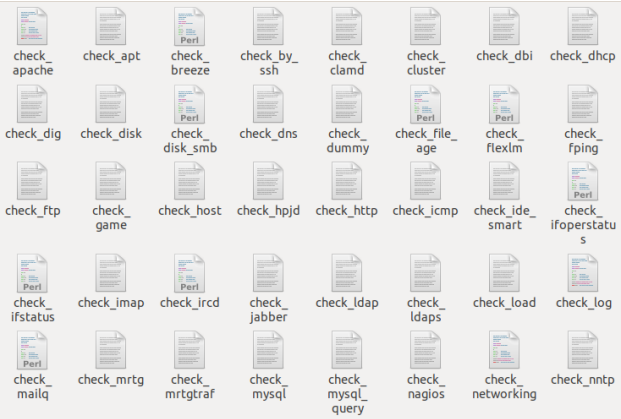

Figure 5: View of the plugins folder in a monitored host

Figure 6: Raspberry Pi 4 Model B

used. Raspbian is a free version of Debian optimised for the Raspberry Pi hardware (11).

Raspberry Pi's use an SD card in order to store the operating system, applications and additional software installed. SD cards are inexpensive and be easily cloned, greatly facilitating backup, replication and fast recovery of Raspberry Pi-based Nagios servers.

The I/O devices and peripherals vary from application to application, while a Raspberry Pi server can run even without peripherals (headless server) at minimal cost. In this case, the network administrator connects to the Nagios server via SSH, a protocol which is supported by Raspberry Pi operating systems.

One of the applications of Raspberry Pi is 'NagiosPi', that is, a Nagios server running on Raspberry Pi; unfortunately the project is not active anymore, as it has not been updated since 2013. However, instructions for installing Nagios server on Raspberry Pi can be found in (12), (13).

1.7 Experimental configuration

In order to develop our own version of NagiosPi, we have deployed an experimental network consisting of a Linux server with LAMP stack (14), a Windows

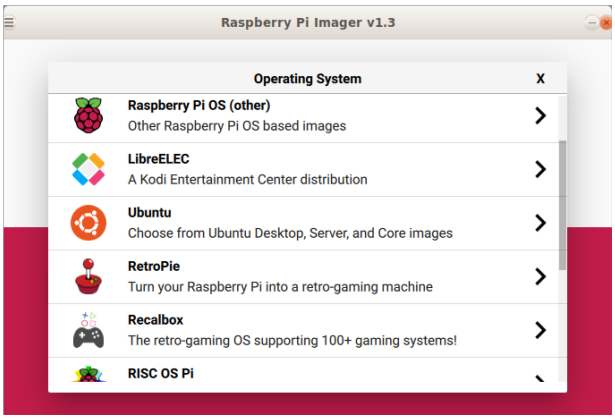

Figure 7: Raspberry Pi ready-made images

PC also running a Linux virtual machine, a modem-router, a Raspberry Pi 1B running Nagios server and a Raspberry Pi 4B also running Nagios server.

2 Enhancing functionality

2.1 Triple modular redundancy of Nagios servers

Triple modular redundancy (TMR) is a form of N-modular redundancy, in which three systems perform the same process and the result is processed by a majority-voting system to produce a single output. If any one of the three systems fails, the other two systems can correct and mask the fault (15). The TMR concept has many applications in computers, such as software redundancy in the form of N-version programming, and hardware redundancy, commonly used in error-correcting code (ECC) memories and elsewhere (16). TMR is used in fault-tolerant computer systems, communication systems, chronometers, etc.

Each Nagios server provides details also about itself ('localhost'). With a single Nagios server the reliability of monitoring information provided is low (what if the Nagios server fails or gets compromised?). Triplication of Nagios servers increases the reliability of Network monitoring.

In order to implement TMR of Nagios servers we need three Raspberry Pi's; the low cost of Raspberry Pi facilitates this issue. The whole software of the first Raspberry Pi (including the Operating System, the Nagios core software, and configuration files) can be easily replicated just by copying the contents of its SD card, saving lots of time and effort. NRPE, SNMP and the Nagios-client configuration files must also be installed to each Raspberry Pi. In our experimental configuration we have tested the monitoring of a Raspberry Pi 1B from a Raspberry Pi 4B.

2.2 Nagios core installation script

The installation of Nagios core is a tedious and time-consuming process. In order to facilitate Nagios server installation, a script automating this process has been developed. Using a script saves time, reduces potential errors, and prevents incorrect configuration. The script performs the following tasks:

- Collects network configuration details from the administrator (including monitored hosts data)
- Downloads and installs Apache web server and other prerequisite software
- Downloads and installs Nagios core software
- Downloads and installs Nagios plugins
- Downloads and installs SNMP
- Downloads and installs NRPE
- Configures the Firewall
- Adds monitored Linux hosts' configuration files
- Updates Nagios server 'commands.cfg' file in order to run NRPE commands
- Performs a sanity check and restarts Apache web server and Nagios server.

The Nagios server installation script contains about 270 lines of code.

2.3 Nagios client installation script

A script for preparing Nagios clients has also been developed. This script performs the following tasks:

- Installs snmpd
- Prepares snmpd.conf
- Restarts snmpd
- Installs nagios-plugins and nrpe-server
- Edits NRPE configuration file
- Restarts nagios-nrpe-server
- Displays useful information to the user.

The Nagios client installation script is 85 lines long.

3 Enhancing security

In this section some custom scripts developed for testing network and host security will be presented.

3.1 Custom scripts

In order to get specific information about Linux host clients connected to the organisational network, three custom scripts providing detailed information about the host systems and their network connections were developed. These scripts were inspired from 'Sherlinux' (17).

Script System_Summary provides a system summary with emphasis on security. It provides information about the hardware (CPU and its capabilities, main or physical memory, swap file, disk partitions, PCI and USB modules, etc.), software (operating system, hostname, kernel release, operating system architecture and much more). A snapshot is provided in Figure 8.

```
======= System Summary =======

System name:
G7
Date:
Sun Jan 24 22:56:30 EET 2021

Kernel version:
4.15.0-20-generic

CPU model: model name : AMD A4-9125 RADEON R3, 4 COMPUTE
CORES 2C+2G
A4-9125 RADEON R3, 4

Main memory: MemTotal: 3988864 kB

3988864 kB
SWAP Memory: SwapTotal: 1718088 kB
1718088 kB

********** Uptime ********
22:56:30 up 3:42, 1 user, load average: 0.00, 0.07, 0.05

******** FREE MEMORY ********
total used free shared buff/cache available
Mem: 3895 1305 1527 9 1062 2326
Swap: 1677 0 1677
Operating System Version
Linux G7 4.15.0-20-generic #21-Ubuntu SMP Tue Apr 24 06:16:15 UTC
2018 x86_64 x86_64 x86_64 GNU/Linux

Operating System Date
#21-Ubuntu SMP Tue Apr 24 06:16:15 UTC 2018

Operating System Release
bionic

Operating System Architecture:
x86_64
x86_64 ==> 64-bit kernel

Main memory: MemTotal: 3988864 kB
3988864 kB
```

Figure 8: System Summary (part)

Script System_Summary also rates the security levels of the various services running (such as Network Manager, dmesg, cups, gdm, etc.). A snapshot is provided in Figure 9.

Script check_Networking provides information about IP addresses, MAC addresses, services and corresponding ports of the local machine, known hosts, number of active connections, number of processes using sockets, number of open TCP and UDP connections, and socket statistics. A snapshot is depicted in Figure 10.

3.2 Apache security check script

In order to check and rate the security level of Apache Web Servers we have developed a special script. In this subsection we shall describe this script. The script implements the following tasks:

- Gets version information about OS and Apache

Figure 9: System security check

- Tests Apache configuration file for syntax errors
- Checks Apache access.log file
- Checks Apache error.log file
- Checks if Apache has a separate user
- Checks if FollowSymLinks is disabled
- Uses Nagiostats utility to get a human-readable output, etc.

According to the Apache script test results, some measures which will harden the Apache installation are proposed, with instructions (18), (19), (20):

- Hide Apache version and OS identity.
- Disable directory listing.
- Keep updating Apache regularly.
- Disable unnecessary modules.
- Use the Allow and Deny directives to restrict access to directories.
- Use mod_security and mod_evasive modules to secure Apache.

The Apache security check script is 75 lines long.

4 Conclusion and Future Work

4.1 Synopsis

In this paper we have presented an innovative network monitoring system based on Raspberry Pi's and Nagios core software.

The main features of the proposed network monitoring system are:

- Low cost because the free edition of Nagios core software, as well as inexpensive but fully functional hardware have been used.

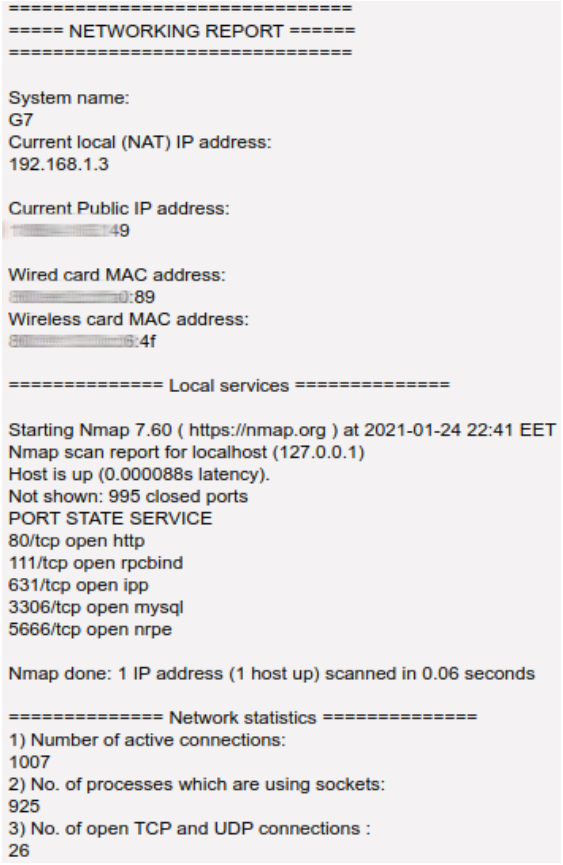

Figure 10: Networking report (part)

- Fast installation based on custom scripts and ready configuration files for Linux servers and clients.
- Focus on security based on custom NRPE plugins.
- Expandability with additional custom scripts.
- Enhanced reliability with triple modular redundancy of Nagios servers.
- Fast and easy backup, replication and recovery of Raspberry Pi-based Nagios servers.

Scripts facilitating the installation of Nagios software to servers, as well as Linux clients have been presented. The use of scripts automates the installation process which is time consuming and tricky to novel users, saving time and effort and minimising installation and configuration errors. In addition, the use of ready-made Linux client configuration files, customised to the needs of specific organisations, saves time and effort, and minimises misconfiguration errors.

Custom scripts to enhance the functionality and security of network monitoring have been presented. A script checking the security of Apache web servers has also been presented.

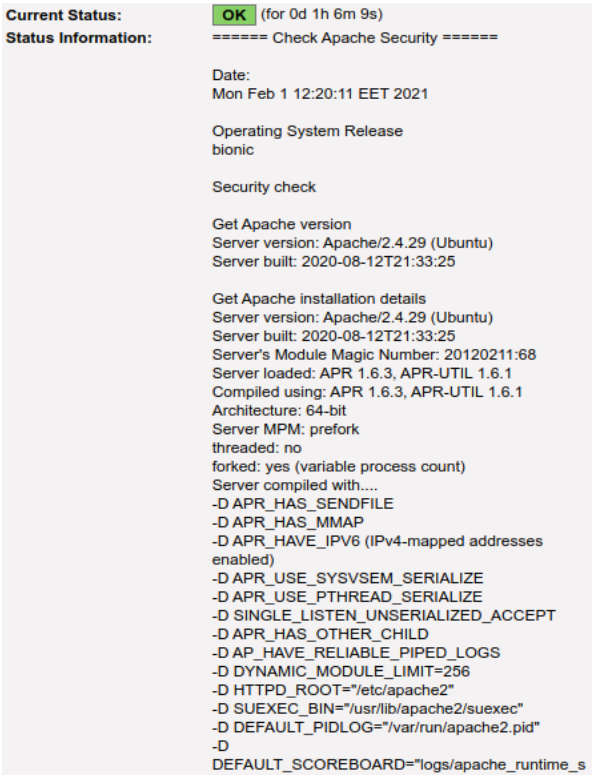

Figure 11: A screenshot of the Check Apache script results

4.2 Future Work

This work may be further developed in the following ways:

- A similar Nagios client installation script for Windows clients may be written.
- A diagnostic script detecting misconfiguration errors may be written.
- A script checking for anomalies in Apache server traffic may be written (21), (22), (23).
- A script checking for anomalies in network traffic may be written.
- Scripts checking a network for Indicators of Compromise (IoC) may be written (24).
- Finally, a Raspbian image with pre-installed Nagios server ('NagiosPi2') may be deployed.

References:

[1] V. Ratan and K.F. Li (2017) NetFlow: Network Monitoring and Intelligence Gathering. In: Xhafa F., Barolli L., Amato F. (eds), *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*, 3PGCIC 2016, Lecture Notes on Data Engineering and Communications Technologies, Vol.1, Springer, Cham.

[2] https://www.pcwdld.com/best-network-monitoring-tools-and-software. Accessed 4 Nov. 2020.

[3] https://www.nagios.com/products. Accessed 3 Nov. 2020.

[4] https://www.tutorialspoint.com/nagios/nagios_nrpe.htm. Accessed 4 Nov. 2020.

[5] http://www.nsclient.org/nsclient. Accessed 7 Nov. 2020.

[6] https://www.tutorialspoint.com/nagios/nagios_add_ons_plugins.htm. Accessed 3 Nov. 2020.

[7] https://www.raspberrypi.org/about. Accessed 7 Nov. 2020.

[8] https://www.raspberrypi.org/products. Accessed 29 Nov. 2020.

[9] https://itsfoss.com/raspberry-pi-projects. Accessed 11 Nov. 2020.

[10] https://www.raspberrypi.org/software/operating-systems. Accessed 11 Nov. 2020.

[11] https://www.raspbian.org. Accessed 7 Nov. 2020.

[12] https://raspberrytips.com/nagios-raspberry-pi. Accessed 4 Nov. 2020.

[13] https://support.nagios.com/kb/article/nagios-core-installing-nagios-core-from-source-96.html#Raspbian. Accessed 4 Nov. 2020.

[14] https://www.ibm.com/cloud/learn/lamp-stack-explained. Accessed 14 Nov. 2020.

[15] R. E. Lyons and W. Vanderkulk, The Use of Triple-Modular Redundancy to Improve Computer Reliability, *IBM Journal of Research and Development*, Vol.6, No.2, 1962, pp. 200–209.

[16] M. L. Shooman. N-Modular Redundancy. *Reliability of computer systems and networks: fault tolerance, analysis and design*. Wiley-Interscience, 2002, pp. 145–201.

[17] A. S. Andreatos, Sherlinux – A tool facilitating Linux forensics. In N.J. Daras (ed.), *Cryptography, Cyber-Security and Information Warfare*. Nova Science Publishers, NY, USA, 2019. Publication Date: December 2018.

[18] L. Welling and L. Thomson, *PHP and MySQL Web Development*, 5th ed., Addison-Wesley, 2017.

[19] J. C. Meloni, *Teach Yourself PHP, MySQL and Apache All in One*, 5th ed., Sams, 2012.

[20] *13 Apache Web Server Security and Hardening Tips*, https://www.tecmint.com/apache-security-tips. Accessed 3 Nov. 2020.

[21] A. P. Leros and A. S. Andreatos, Network Traffic Analytics for Internet Service Providers – Application in Early Prediction of DDoS Attacks. In: G. Tsihrintzis, D. Sotiropoulos and L. Jain (eds), *Machine Learning Paradigms. Intelligent Systems Reference Library*, Vol.149, Springer, Cham, 2019. https://link.springer.com/chapter/10.1007/978-3-319-94030-4_10

[22] A. S. Andreatos and V. C. Moussas, A Modular Intrusion Detection System based on Artificial Neural Networks, *International Journal of Neural Networks and Advanced Applications*. ISSN: 2313-0563, Vol.6, 2019, pp. 32-38.

[23] V. Moussas, Adaptive Traffic Modelling for Network Anomaly Detection, in 3rd Int'l Conf. on Cryptography, Cyber Security and Information Warfare (3rd CryCybIW), 26-27 May 2016, Hellenic Military Academy, Attica, Greece. http://users.uniwa.gr/vmouss/papers/P57.pdf

[24] https://www.forcepoint.com/cyber-edu/indicators-compromise-ioc. Accessed 15 Nov. 2020.