



## Лабораторна робота №6

### Налаштування Nagios моніторингу на базі NRPE (Nagios Remote Plugin Executor).

**Мета:** формування практичних навичок налаштування та використання протоколу NRPE (Nagios Remote Plugin Executor) для моніторингу серверів Linux у системі Nagios, зокрема конфігурації NRPE на сервері моніторингу та віддаленому Linux-хості, а також додавання специфічних параметрів моніторингу.

**Інструменти:** гіпервізор VirtualBox, модель комп'ютерної мережі.

### Теоретичні відомості

На рис.6.1. наведена модель комп'ютерної мережі, побудована під час виконання попередніх лабораторних робіт. До серверу Serv-G-N-3 налаштовано SSH доступ через NAT Network для VirtualBox Host.

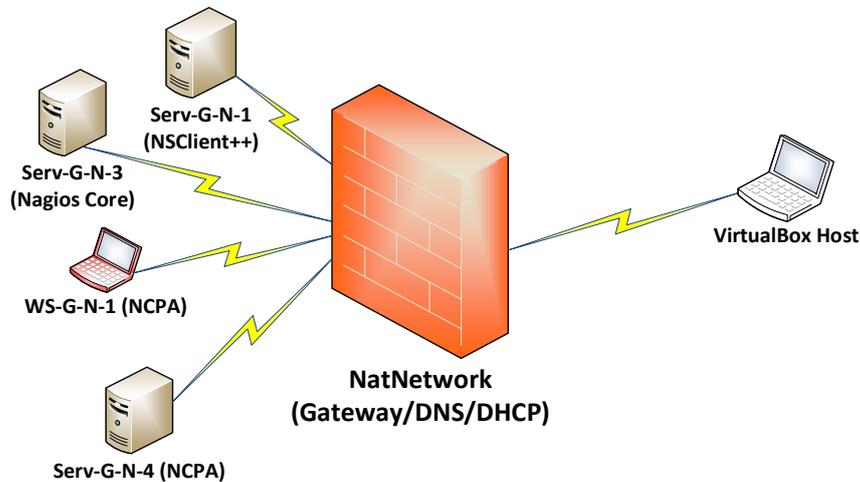


Рис. 6.1. Топологія мережі

На сервері Serv-G-N-3 розгорнуто систему моніторингу на базі Nagios 4.X. Моніторинг основних сервісів серверу Serv-G-N-1 виконується за допомогою NSClient++. Основні сервіси робочої станції WS-G-N-1 та Ubuntu-серверу Serv-G-N-4 відслідковуються за допомогою NCPA. Налаштовано підключення з хосту NAT Network по протоколу HTTP до систему моніторингу під користувачем nagios.

NRPE розроблений, щоб дозволити запускати плагіни Nagios на віддалених машинах Linux/Unix. Основна причина для цього полягає в тому, щоб дозволити Nagios контролювати «локальні» ресурси (наприклад, навантаження ЦП, використання пам'яті тощо) на віддалених машинах. Оскільки ці загальнодоступні ресурси зазвичай не доступні зовнішнім машинам, на віддалених машинах Linux/Unix потрібно встановити такий агент, як NRPE.

Плагіни Nagios можна запускати на віддалених машинах Linux/Unix через SSH. Існує плагін check\_by\_ssh, який дозволяє це зробити. Використання SSH є більш безпечним, ніж агент NRPE, але воно також накладає більші витрати (ЦП) як на моніторинг, так і на віддалені машини. Це може стати проблемою, коли ви починаєте стежити за сотнями або тисячами машин. Багато адміністраторів Nagios вибирають використання NRPE через менше навантаження.

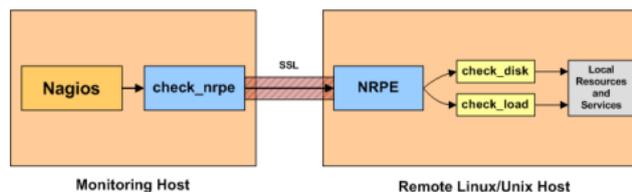


Рис. 6.2. Дизайн Nagios – SSL – NRPE.

NRPE складається з двох частин: плагіна check\_nrpe, який знаходиться на Nagios сервері моніторингу та демону NRPE, який працює на віддаленій машині Linux/Unix

Коли Nagios потрібно контролювати ресурс обслуговування віддаленої машини Linux/Unix, Nagios запускає плагін check\_nrpe та повідомляє, яку службу потрібно перевірити. check\_nrpe зв'язується з демоном NRPE на віддаленому хості через (опціонально) захищене з'єднання SSL.



Демон NRPE запускає відповідний плагін Nagios для перевірки служби чи ресурсу. Результати перевірки служби передаються від демона NRPE назад до плагіна check\_nrpe, який потім повертає результати перевірки процесу Nagios

Демон NRPE вимагає, щоб плагіни Nagios були встановлені на віддаленому хості Linux/Unix. Без них демон не міг би нічого контролювати.

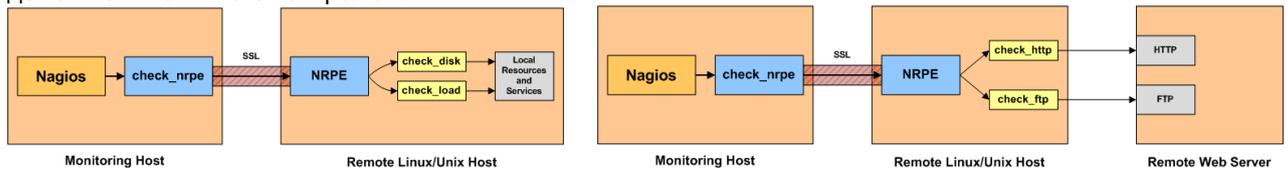


Рис. 6.3. Прямі та непрямі перевірки Nagios – NRPE.

Найбільш простим використанням NRPE є моніторинг «локальних» або «приватних» ресурсів на віддаленій машині Linux/Unix. Сюди входять такі речі, як навантаження ЦП, використання пам'яті, використання підкачки, поточні користувачі, використання диска, стани процесів тощо.

Також можливо використовувати NRPE для опосередкованої перевірки «загальнодоступних» служб і ресурсів віддалених серверів, які можуть бути недоступні безпосередньо з хосту моніторингу. Наприклад, якщо віддалений хост, на якому встановлено демон і плагіни NRPE, може спілкуватися з віддаленим веб-сервером (але хост моніторингу не може). Демон NRPE може бути налаштовано, щоб дозволити опосередковано контролювати віддалений веб-сервер. У цьому випадку демон NRPE по суті діє як проксі.

### **Nagios сервер. Serv-G-N-3.**

Для встановлення Nagios Remote Plugin Executor необхідно отримати лінк актуального релізу пакету на сторінці підтримки <https://github.com/NagiosEnterprises/nrpe/releases>

На момент написання цього документа це був NRPE version 4.1.3 Release. Завантажуємо інсталяційний пакет

```
cd /tmp
wget https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-4.1.3/nrpe-4.1.3.tar.gz
Розпаковуємо завантажений архів
tar -xzf nrpe-4.1.3.tar.gz
```

Переходимо в каталог, який був створений під час розпакування, оновлюємо репозиторій, та встановлюємо необхідні для роботи NRPE пакети та виконуємо встановлення NRPE:

```
cd nrpe-4.1.3
sudo apt-get update
sudo apt-get install -y autoconf automake gcc libc6 libmcrypt-dev make libssl-dev wget openssl
sudo ./configure --enable-command-args --with-ssl-lib=/usr/lib/x86_64-linux-gnu/
sudo make all
sudo cp /tmp/nrpe-4.1.3/src/check_nrpe /usr/local/nagios/libexec/check_nrpe
```

Виконуємо команду, щоб переконатися, що check\_nrpe встановлений:

```
/usr/local/nagios/libexec/check_nrpe -V
```

Результат має бути подібний до:

```
student@serv-22-45-3:/tmp/nrpe-4.1.3$ sudo cp /tmp/nrpe-4.1.3/src/check_nrpe /usr/local/nagios/libexec/check_nrpe
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -V
NRPE Plugin for Nagios
Version: 4.1.3
student@serv-22-45-3:/tmp/nrpe-4.1.3$
```

Рис. 6.4. check\_nrpe -V на Nagios-сервері Serv-22-45-3

Додаємо у конфігураційний файл команд `/usr/local/nagios/etc/objects/commands.cfg` секцію визначення для встановленої команди

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

«Звична», але обов'язкова операція – перевірка вірності внесених у конфігурацію змін та перезапуск сервісу Nagios:

```
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
sudo service nagios restart
```



#### Linux сервер. Serv-G-N-4.

Конфігурування серверної частини завершено, переходимо на Linux-сервер Serv-G-N-4 та повторюємо процедуру встановлення пакету NRPE. Завантажуємо інсталяційний пакет:

```
cd /tmp
```

```
wget https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-4.1.3/nrpe-4.1.3.tar.gz
```

Розпаковуємо завантажений архів

```
tar -xvzf nrpe-4.1.3.tar.gz
```

Переходимо в каталог, який був створений під час розпакування, оновлюємо репозиторій, та встановлюємо необхідні для роботи NRPE пакети:

```
cd nrpe-4.1.3
```

```
sudo apt-get update
```

```
sudo apt-get install -y autoconf automake gcc libc6 libmcrypt-dev make libssl-dev wget openssl
```

Встановлюємо NRPE:

```
sudo ./configure --enable-command-args --with-ssl-lib=/usr/lib/x86_64-linux-gnu/
```

```
sudo make all
```

```
sudo make install
```

Файли конфігурації.

```
sudo make install-config
```

Оновлюємо файл служб /etc/services, що використовується програмами для перекладу зрозумілих людині назв служб у номери портів під час підключення до машини через мережу.

```
sudo sh -c "echo >> /etc/services"
```

```
sudo sh -c "sudo echo '# Nagios services' >> /etc/services"
```

```
sudo sh -c "sudo echo 'nrpe 5666/tcp' >> /etc/services"
```

Це встановлює файли служби або демона.

```
sudo make install-init
```

```
sudo systemctl enable nrpe.service
```

Якщо на сервері увімкнений брандмауер, налаштуємо його. Перевірка стану брандмауера:

```
sudo ufw status.
```

Якщо отримали **Status: inactive** або **Invalid syntax**, правила можна не налаштовувати.

Налаштування правил брандмауер виконується лише у випадку його «активності»

Порт 5666 використовується NRPE, і його потрібно відкрити на локальному брандмауері.

```
sudo mkdir -p /etc/ufw/applications.d
```

```
sudo sh -c "echo '[NRPE]' > /etc/ufw/applications.d/nagios"
```

```
sudo sh -c "echo 'title=Nagios Remote Plugin Executor' >> /etc/ufw/applications.d/nagios"
```

```
sudo sh -c "echo 'description=Allows remote execution of Nagios plugins' >> /etc/ufw/applications.d/nagios"
```

```
sudo sh -c "echo 'ports=5666/tcp' >> /etc/ufw/applications.d/nagios"
```

```
sudo ufw allow NRPE
```

```
sudo ufw reload
```

З правилами брандмауера завершили, переходимо до оновлення файлу конфігурації **/usr/local/nagios/etc/nrpe.cfg**, де змінюємо два рядки:

```
allowed_hosts=127.0.0.1,:::1,192.168.45.136
dont_blame_nrpe=1
```

Allowed hosts – рядок адрес, дозволених для NRPE серверів. Крім локальної адреси додаємо через кому адресу Serv-G-N-3 (Nagios серверу).

dont\_blame\_nrpe визначає, чи дозволить демон NRPE клієнтам вказувати аргументи для команд, які виконуються.

Ці зміни файлу конфігурації **/usr/local/nagios/etc/nrpe.cfg** також можливо виконати наступними командами (якщо зміни внесено в «ручному» режимі, команди не виконуються☺):

```
sudo sh -c "sed -i '/^allowed_hosts=/s/$/,192.168.22.136/' /usr/local/nagios/etc/nrpe.cfg"
```

```
sudo sh -c "sed -i 's/^dont_blame_nrpe =.*/dont_blame_nrpe=1/g' /usr/local/nagios/etc/nrpe.cfg"
```

Запускаємо службу / демон:

```
sudo systemctl start nrpe.service
```

Перевіряємо, що NRPE слухає запити та відповідає на них.

```
/usr/local/nagios/libexec/check_nrpe -H 127.0.0.1
```

Результат має бути подібний до:

```
NRPE v4.1.3
```



```
student@serv-22-45-4:/tmp/nrpe-4.1.3$ sudo ufw status
Status: inactive
student@serv-22-45-4:/tmp/nrpe-4.1.3$ sudo vi /usr/local/nagios/etc/nrpe.cfg
student@serv-22-45-4:/tmp/nrpe-4.1.3$ sudo systemctl start nrpe.service
student@serv-22-45-4:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 127.0.0.1
NRPE v4.1.3
student@serv-22-45-4:/tmp/nrpe-4.1.3$
```

Рис. 6.5. `check_nrpe -H 127.0.0.1` на `Serv-22-45-4`

Для коректної роботи NRPE, необхідно встановити Nagios плагіни на сервер `Serv-22-45-4`. Плагіни допомагають NRPE виконувати різноманітні перевірки та збирати дані для подальшого відображення в Nagios.

Завантажуємо останню версію Nagios плагінів:

```
cd /tmp
```

```
wget https://nagios-plugins.org/download/nagios-plugins-2.4.9.tar.gz
```

Розпаковка, конфігурація та встановлення:

```
tar -xzvf nagios-plugins-2.4.9.tar.gz
```

```
cd nagios-plugins-2.4.9
```

```
./configure
```

```
make
```

```
sudo make install
```

Після цих кроків на сервері `Serv-22-45-4` встановлено Nagios плагіни, які можна використовувати для налаштування активних перевірок за допомогою NRPE.

### Nagios сервер. Serv-G-N-3.

Переходимо на Nagios-сервер та перевіряємо роботу віддаленого виклику процедур перевірок:

```
/usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_users
```

```
/usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_load
```

```
/usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_total_procs
```

```
/usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_zombie_procs
```

Відповідно – завантаження ЦП, кількість підключених користувачів, кількість процесів та кількість «зомбі» процесів.

```
student@serv-22-45-3:/tmp/nrpe-4.1.3$ sudo service nagios restart
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_users
USERS OK - 2 users currently logged in |users=2;5;10;0
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_load
CRITICAL - load average: 0.25, 0.34, 0.18|load1=0.250;0.150;0.300;0; load5=0.340;0.100;0.250;0; load15=0.180;0.050;0.200;0;
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_total_procs
PROCS OK: 111 processes | procs=111;150;200;0;
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_zombie_procs
PROCS OK: 0 processes with STATE = Z | procs=0;5;10;0;
```

Рис. 6.6. Виклик `check_nrpe` з `Serv-22-45-3` на `Serv-22-45-4`. Прямі перевірки.

### Linux сервер. Serv-G-N-4.

Розглянемо що ж це за команди та де їх можливо визначити. Відкриваємо файл конфігурації `/usr/local/nagios/etc/nrpe.cfg` та шукаємо рядки визначення описаних команд:

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

Спробуємо додати визначення команди. Візьмемо за основу нової команди плагін `/usr/local/nagios/libexec/check_dns`, який вже встановлено на `Serv-G-N-3`. Плагін дозволяє перевірити доступність домену. Наприклад перевірка домену `ztu.edu.ua`

```
/usr/local/nagios/libexec/check_dns -H ztu.edu.ua
```

```
student@serv-22-45-4:/tmp/nagios-plugins-2.4.9$ /usr/local/nagios/libexec/check_dns -H ztu.edu.ua
DNS OK: 0.050 seconds response time. ztu.edu.ua returns 46.4.92.154|time=0.050383s;;;0.000000
student@serv-22-45-4:/tmp/nagios-plugins-2.4.9$ /usr/local/nagios/libexec/check_dns -H ukr.net
DNS OK: 0.052 seconds response time. ukr.net returns 35.186.218.67|time=0.052185s;;;0.000000
student@serv-22-45-4:/tmp/nagios-plugins-2.4.9$
```

Рис. 6.7. `Serv-22-45-4`. `check_dns` по двом доменним іменам.



Наступні дії виконуються тільки, якщо сервер «не бачить» домени.

Якщо при вірних мережевих налаштуваннях сервер Serv-G-N-4 «не бачить» домени може знадобитись редагування конфігураційних файлів `/etc/resolv.conf` та `/etc/hosts`. Відкриваємо файл `/etc/resolv.conf` для редагування та заміняємо існуючі записи `nameserver` записами DNS-серверів, що використовуються у лабораторній роботі:

`nameserver 192.168.45.129`

`nameserver 10.10.5.86`

Вводимо зміни у файлі `/etc/resolv.conf` командою

`sudo systemctl restart systemd-resolved`

Редагуємо у конфігураційному файлі `/etc/hosts` записи серверів:

`127.0.0.1 localhost`

`127.0.1.1 serv-22-45-4`

`192.168.40.135 serv-22-45-1`

`192.168.40.136 serv-22-45-3`

Звичайно можна додати для кожного домену, що нам необхідно перевіряти, окрему команду у файл конфігурації `/usr/local/nagios/etc/nrpe.cfg`, а можна створити одну, універсальну для будь якого домену. Вона буде виглядати наступним чином:

```
command[check_dns]=/usr/local/nagios/libexec/check_dns -H $ARG1$
```

```
student@serv-22-45-4: /tmp/nagios-plugins-2.4.9
# The following examples use hardcoded command arguments...
# This is by far the most secure method of using NRPE

command[check_users]=/usr/local/nagios/libexec/check_users -w 5 -c 10
command[check_load]=/usr/local/nagios/libexec/check_load -r -w .15,.10,.05 -c .30,.25,.20
command[check_hda1]=/usr/local/nagios/libexec/check_disk -w 30% -c 10% -p /dev/hda1
command[check_zombie_procs]=/usr/local/nagios/libexec/check_procs -w 5 -c 10 -s Z
command[check_total_procs]=/usr/local/nagios/libexec/check_procs -w 150 -c 200
command[check_dns]=/usr/local/nagios/libexec/check_dns -H $ARG1$

# The following examples allow user-supplied arguments and can
```

Рис. 6.8. Serv-22-45-4. Додавання `check_dns` у файл `/usr/local/nagios/etc/nrpe.cfg`.

Не забуваємо, що після цих налаштувань необхідно перезавантажити службу `nrpe` командою `sudo systemctl restart nrpe`

### Nagios сервер. Serv-G-N-3.

Переходимо на Nagios та виконуємо виклик налаштованої команди. У команді `check_nrpe`, аргументи передаються після опції `-a`, і кожен аргумент розділяється пробілом. `check_nrpe` для домену `ztu.edu.ua`

`/usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_dns -a ztu.edu.ua`

```
student@serv-22-45-3: /tmp/nrpe-4.1.3
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_dns -a ztu.edu.ua
DNS OK: 0.041 seconds response time. ztu.edu.ua returns 46.4.92.154|time=0.040863s;;0.000000
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_dns -a ukr.net
DNS OK: 0.090 seconds response time. ukr.net returns 35.166.218.67|time=0.089718s;;0.000000
student@serv-22-45-3:/tmp/nrpe-4.1.3$ /usr/local/nagios/libexec/check_nrpe -H 192.168.45.132 -c check_dns -a meta.ua
DNS OK: 0.058 seconds response time. meta.ua returns 172.66.40.158,172.66.43.98|time=0.057745s;;0.000000
student@serv-22-45-3:/tmp/nrpe-4.1.3$
```

Рис. 6.9. Віддалений виклик `check_dns` для кількох доменних імен на Serv-22-45-4 з серверу Serv-22-45-3

Додаємо налаштовані команди до конфігураційного файлу сервера `/usr/local/nagios/etc/objects/linux/serv-22-45-4.cfg`.

```
define service {
    host_name          serv-22-45-4
    use                generic-service
    service_description Check users
    check_command      check_nrpe!check_users
    max_check_attempts 5
    check_interval     5
    retry_interval     1
    check_period       24x7
    notification_interval 60
    notification_period 24x7
}
define service {
    host_name          serv-22-45-4
```



```
use generic-service
service_description Check load CPU
check_command check_nrpe!check_load
max_check_attempts 5
check_interval 5
retry_interval 1
check_period 24x7
notification_interval 60
notification_period 24x7
}
define service {
    host_name serv-22-45-4
    use generic-service
    service_description Check Total procs
    check_command check_nrpe!check_total_procs
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period 24x7
    notification_interval 60
    notification_period 24x7
}
define service {
    host_name serv-22-45-4
    use generic-service
    service_description Check Zombie procs
    check_command check_nrpe!check_zombie_procs
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period 24x7
    notification_interval 60
    notification_period 24x7
}
define service {
    host_name serv-22-45-4
    use generic-service
    service_description Check domain falkovsky.net
    check_command check_nrpe!check_dns -a meta.ua
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period 24x7
    notification_interval 60
    notification_period 24x7
}
define service {
    host_name serv-22-45-4
    use generic-service
    service_description Check domain ztu.edu.ua
    check_command check_nrpe!check_dns -a ztu.edu.ua
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period 24x7
    notification_interval 60
    notification_period 24x7
}
define service {
    host_name serv-22-45-4
    use generic-service
    service_description Check domain learn.ztu.edu.ua
    check_command check_nrpe!check_dns -a learn.ztu.edu.ua
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period 24x7
    notification_interval 60
    notification_period 24x7
}
define service {
    host_name serv-22-45-4
    use generic-service
    service_description Check domain portal.ztu.edu.ua
    check_command check_nrpe!check_dns -a portal.ztu.edu.ua
    max_check_attempts 5
    check_interval 5
    retry_interval 1
    check_period 24x7
    notification_interval 60
    notification_period 24x7
}
}
```

Перевірка вірності внесених у конфігурацію змін та перезапуск сервісу Nagios:

```
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
sudo service nagios restart
```

Переглядаємо роботу виконаних налаштувань (рис. 6.10).

Тимчасово, щоб запобігти можливому ефекту DDoS-атак на доменах ztu.edu.ua, коментарями вимикаємо їх налаштований моніторинг.

```
#define service {
#   host_name serv-22-40-3
#   use generic-service
#   service_description Check domain ztu.edu.ua
#   check_command check_nrpe!check_dns -a meta.ua
#   max_check_attempts 5
#   check_interval 5
#   retry_interval 1
#   check_period 24x7
#   notification_interval 60
#   notification_period 24x7
#}
```



```
#define service {
#   host_name          serv-22-40-3
#   use                generic-service
#   service_description Check domain learn.ztu.edu.ua
#   check_command      check_nrpe!check_dns -a learn.tu.edu.ua
#   max_check_attempts 5
#   check_interval     5
#   retry_interval     1
#   check_period       24x7
#   notification_interval 60
#   notification_period 24x7
#}
#define service {
#   host_name          serv-22-1-3
#   use                generic-service
#   service_description Check domain portal.ztu.edu.ua
#   check_command      check_nrpe!check_dns -a portal.tu.edu.ua
#   max_check_attempts 5
#   check_interval     5
#   retry_interval     1
#   check_period       24x7
#   notification_interval 60
#   notification_period 24x7
#}
```

Виконуємо перевірку вірності внесених у конфігурацію змін та перезапускаємо сервіс Nagios:

```
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
sudo service nagios restart
```

Переглядаємо отриману «картину» моніторингу хосту Serv-22-45-4 (рис.6.11)

Service Status Details For Host 'serv-22-45-4'

Host	Service	Status	Last Check	Duration	Attempt	Status Information
serv-22-45-4	CPU Usage	OK	07-15-2025 09:24:12	0d 0h 58m 4s	1/5	OK: Percent was 0.00 %
	Check Total procs	OK	07-15-2025 09:24:44	0d 0h 20m 59s	1/5	PROCS OK: 112 processes
	Check Zombie procs	OK	07-15-2025 09:25:15	0d 0h 20m 28s	1/5	PROCS OK: 0 processes with STATE = Z
	Check domain falkovskynet	OK	07-15-2025 09:24:48	0d 0h 0m 55s	1/5	DNS OK: 0.125 seconds response time. meta.ua returns 172.66.40.158;72.66.43.98
	Check domain learn.ztu.edu.ua	OK	07-15-2025 09:21:18	0d 0h 19m 25s	1/5	DNS OK: 0.040 seconds response time. learn.ztu.edu.ua returns 46.4.92.154
	Check domain portal.ztu.edu.ua	OK	07-15-2025 09:21:50	0d 0h 18m 53s	1/5	DNS OK: 0.036 seconds response time. portal.ztu.edu.ua returns 46.4.92.154
	Check domain ztu.edu.ua	OK	07-15-2025 09:22:22	0d 0h 18m 21s	1/5	DNS OK: 0.033 seconds response time. ztu.edu.ua returns 46.4.92.154
	Check load CPU	OK	07-15-2025 09:22:53	0d 0h 17m 50s	1/5	OK - load average: 0.00, 0.00, 0.00
	Check users	OK	07-15-2025 09:23:25	0d 0h 17m 18s	1/5	USERS OK - 2 users currently logged in
	Disk SDA, Read bytes	OK	07-15-2025 09:21:10	0d 2h 24m 33s	1/5	OK: Read_bytes was 0.00 MB/s
	Disk SDA, Read time	OK	07-15-2025 09:21:53	0d 2h 23m 50s	1/5	OK: Read_time was 0.00 ms/s
	Disk SDA, Write bytes	OK	07-15-2025 09:22:36	0d 2h 23m 7s	1/5	OK: Write_bytes was 0.01 MB/s
	Ethernet, Received bytes	OK	07-15-2025 09:25:20	0d 0h 56m 24s	1/5	OK: Bytes_recv was 0.14 KB/s
	Ethernet, Sent bytes	OK	07-15-2025 09:24:03	0d 2h 21m 40s	1/5	OK: Bytes_sent was 0.18 KB/s
	Logical disk	OK	07-15-2025 09:23:48	0d 1h 56m 55s	1/5	OK: Free was 1.35 GB
	Logical disk used percent	OK	07-15-2025 09:22:45	0d 2h 22m 58s	1/5	OK: Used_percent was 65.40 %
	Memory Usage	OK	07-15-2025 09:20:43	0d 2h 25m 0s	1/5	OK: Memory usage was 25.60 % (Available: 1.14 GB, Total: 1.54 GB, Free: 0.18 GB, Used: 0.22 GB)
	Process Count	OK	07-15-2025 09:21:50	0d 2h 23m 53s	1/5	OK: Process count was 108
	System operation time	OK	07-15-2025 09:21:56	0d 2h 23m 47s	1/5	OK: Uptime was 3 hours 15 minutes 9 seconds

Results 1 - 19 of 19 Matching Services

Рис. 6.10. Перегляд сервісів Serv-22-45-4

Service Status Details For Host 'serv-22-40-3'

Host	Service	Status	Last Check	Duration	Attempt	Status Information
serv-22-40-3	CPU Usage	OK	11-27-2024 17:50:40	0d 1h 17m 31s	1/5	OK: Percent was 0.00 %
	Check Total procs	OK	11-27-2024 17:50:58	0d 0h 21m 15s	1/5	PROCS OK: 110 processes
	Check Zombie procs	OK	11-27-2024 17:51:27	0d 0h 20m 44s	1/5	PROCS OK: 0 processes with STATE = Z
	Check domain falkovskynet	OK	11-27-2024 17:51:58	0d 0h 20m 13s	1/5	DNS OK: 0.035 seconds response time. falkovskynet returns 192.168.4.0131
	Check load CPU	OK	11-27-2024 17:51:01	0d 0h 12m 10s	1/5	OK - load average: 0.00, 0.00, 0.00
	Check users	OK	11-27-2024 17:49:31	0d 0h 17m 40s	1/5	USERS OK - 2 users currently logged in
	Disk SDA, Read bytes	OK	11-27-2024 17:49:54	0d 3h 32m 17s	1/5	OK: Read_bytes was 0.00 MB/s
	Disk SDA, Read time	OK	11-27-2024 17:47:28	0d 3h 31m 39s	1/5	OK: Read_time was 0.00 ms/s
	Disk SDA, Write bytes	OK	11-27-2024 17:47:15	0d 3h 31m 1s	1/5	OK: Write_bytes was 0.00 MB/s
	Ethernet, Received bytes	OK	11-27-2024 17:52:09	0d 1h 17m 17s	1/5	OK: Bytes_recv was 0.51 KB/s
	Ethernet, Sent bytes	OK	11-27-2024 17:47:27	0d 3h 29m 44s	1/5	OK: Bytes_sent was 0.17 KB/s
	Logical disk	OK	11-27-2024 17:52:08	0d 0h 0m 5s	1/5	OK: Free was 5.21 GB
	Logical disk used percent	OK	11-27-2024 17:48:43	0d 3h 28m 28s	1/5	OK: Used_percent was 54.30 %
	Memory Usage	OK	11-27-2024 17:47:24	0d 3h 49m 47s	1/5	OK: Memory usage was 42.30 % (Available: 0.58 GB, Total: 1.00 GB, Free: 0.07 GB, Used: 0.28 GB)
	Process Count	OK	11-27-2024 17:48:16	0d 3h 48m 56s	1/5	OK: Process count was 107
	System operation time	OK	11-27-2024 17:47:34	0d 3h 29m 37s	1/5	OK: Uptime was 6 hours 14 minutes 13 seconds

Results 1 - 16 of 16 Matching Services

Рис. 6.11. Перегляд сервісів Serv-22-45-4



## Завдання до лабораторної роботи

1. Виконайте налаштування NRPE на Nagios-сервері Serv-G-N-3.
2. Виконайте налаштування NRPE-серверу на лінукс-сервері Serv-G-N-4.
3. Додайте кілька, на Ваш вибір, параметрів NRPE-моніторингу що виконуються на Serv-G-N-4.

## Звіт має містити:

- лістинг використаних команд;
- скріншоти отриманих результатів моніторингу у Nagios 4;
- короткий опис редагування файлів конфігурації Nagios 4.

Додаток 1.

### Виправлення помилки з запуском клієнта Nagios на Ubuntu.

Виправлення помилки з роботою ncpa. Перевіряємо статус служби клієнта.

```
sudo service ncpa status
```

Вимикаємо автозапуск служби NCPA, щоб вона не запускалася під час перезавантаження системи.

```
sudo systemctl disable ncpa
```

```
sudo reboot
```

```
sudo service ncpa status
```

Змінюємо права доступу до файлу процесу ncpa.pid, щоб дозволити його видалення та видаляємо.

```
sudo chmod 777 /usr/local/ncpa/var/run/ncpa.pid
```

```
sudo rm /usr/local/ncpa/var/run/ncpa.pid
```

Перезапускаємо службу та відновлюємо її автозапуск.

```
sudo service ncpa restart
```

```
sudo systemctl enable ncpa
```

Це універсальний алгоритм відновлення та працює також з nrpe. Файл процесу nrpe-клієнта знаходиться по шляху /usr/local/nagios/var/nrpe.pid

### Корисні посилання

- Nagios Add-Ons Projects

<https://www.nagios.org/downloads/nagios-core-addons/>

- NRPE - How To Install NRPE v4 From Source

<https://support.nagios.com/kb/article/nrpe-how-to-install-nrpe-v4-from-source-515.html>

- NRPE - How to install NRPE

<https://support.nagios.com/kb/article/nrpe-how-to-install-nrpe-8.html>

- Index of /downloads/nagiosxi/agents

<https://assets.nagios.com/downloads/nagiosxi/agents/>

- Exchange Nagios. NRPE - Nagios Remote Plugin Executor

<https://exchange.nagios.org/directory/Addons/Monitoring-Agents/NRPE--2D-Nagios-Remote-Plugin-Executor/details>

- Using NSClient++ with check\_nrpe

<https://nsclient.org/docs/howto/nrpe/>

- The Nagios Plugins. Category: Operating Systems

<https://exchange.nagios.org/directory/Plugins/Operating-Systems>