



"ШЛЮЗ" ЗІ СКЛАДУ "КОМПЛЕКСУ ПРОГРАМНОГО КРИПТОГРАФІЧНОГО ЗАХИСТУ ІНФОРМАЦІЇ "QUANT"

Настанова оператора
Версія 1/2018

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ	3
ВСТУП	4
СКЛАДОВІ КОМПОНЕТИ ЗАСОБУ	4
МІНІМАЛЬНІ ТЕХНІЧНІ ВИМОГИ	4
СУМІСНІСТЬ З ОПЕРАЦІЙНИМИ СИСТЕМАМИ	4
ВСТАНОВЛЕННЯ СЕРВЕРНОЇ КОМПОНЕНТИ МШ	5
1. Авторизація у операційній системі	5
2. Оновлення індексів пакетів операційної системи	5
3. Встановлення необхідного пакету бібліотек (програм)	6
4. Копіювання програмного модуля «Модуль шифрування» на сервер	7
5. Видалення попередньої версії програмного модуля «Модуль шифрування» (за необхідності)	7
6. Підготовка до інсталяції програмного модуля «Модуль шифрування»	7
7. Інсталяція програмного модуля «Модуль шифрування»	7
8. Перенесення ключових даних до відповідних підкаталогів	10
9. Перший запуск програмного модуля «Модуль шифрування»	12
10. Файл /etc/sysctl.conf	12
11. Конфігурування модуля шифрування. Файл ipsec.conf	13
12. Запуск модуля шифрування	15
13. Типові команди для адміністрування МШ	16
14. Автоматичний запуск МШ після завантаження/перезавантаження операційної системи	17
15. Використання та активація захищеного носія ключової інформації "ЕФІТ КЕЙ" (EfitKey)	17
16. Використання ключових даних наданих АЦСК	19
17. Використання ключових даних сформованих іншими стандартами криптографії (PFX та PKCS)	20
18. Журналювання подій роботи МШ	20
19. Приклад конфігураційних файлів	21
АДМІНСТРУВАННЯ КОНТЕНТУ	23
ПРИМІТКИ АДМІНІСТРАТОРУ	23

Пор. № зміни	Підпис відпов. особи	Дата внесення

ПЕРЕЛІК СКОРОЧЕНЬ

АРМ	Автоматизоване робоче місце
ОС	Операційна система
ПК	Персональний комп'ютер
Адміністратор	Уповноважена особа, щодо здійснення налаштування МШ
МШ	Модуль шифрування, а саме «Шлдюз» зі складу «Комплексу програмного криптографічного захисту інформації «QUANT»
ЗМ	Захищена мережа (локальна мережа МШ)
ЦГК	Центр генерації ключів
Ключові дані	Група файлів, що генерується у ЦГК для відповідного МШ. До цієї групи входить: кореневий сертифікат (ca.crt), локальний сертифікат (*.crt), контейнер (*.cnt), dke-файл (*.dke), файл з паролем до сертифікату (*.pwd)

Пор. № зміни	Підпис відпов. особи	Дата внесення

ВСТУП

Документ містить опис дій адміністратора безпеки (користувача) МШ, який є складовою частиною «Комплексу програмного криптографічного захисту інформації «QUANT» (далі – Комплекс) та призначений для побудови захищеної мережі та встановлюється на границі захищеної мережі (далі – ЗМ) або границі сегмента ЗМ, що функціонує в інтересах одного, декількох або всіх суб'єктів (об'єктів) даної ЗМ (сегмента ЗМ), що забезпечує створення захищених з'єднань із іншими довіреними модулями шифрування, що входять до складу певної автоматизованої системи.

Документ описує дії адміністратора безпеки (користувача, уповноваженої особи), щодо однієї з компонент Комплексу.

Даний документ містить опис послідовності дій адміністратора безпеки (користувача, уповноваженої особи) щодо розгортання та адміністрування МШ.

СКЛАДОВІ КОМПОНЕТИ ЗАСОБУ

МШ складається з наступних компонентів:

- ✓ Клієнтська складова
- ✓ Серверна складова

МІНІМАЛЬНІ ТЕХНІЧНІ ВИМОГИ

Центральний процесор: 2 Core, Intel® Xeon® 1800 Mhz

Графічний адаптер: наявності

Оперативна пам'ять: 4 Гб

Вільне місце на жорсткому диску: 60 Гб

Мережева карта: 2x100 Мбіт/с

Примітка: Зазначені технічні вимоги є мінімально необхідними для функціонування програмного забезпечення.

СУМІСНІСТЬ З ОПЕРАЦІЙНИМИ СИСТЕМАМИ

64-бітна ОС: Ubuntu 16.04.N-server-amd64 LTS

, де N – порядковий номер збірки операційної системи (1, 2, 3, тощо)

Пор. № зміни	Підпис відпов. особи	Дата внесення

ВСТАНОВЛЕННЯ СЕРВЕРНОЇ КОМПОНЕНТИ МШ

Розгортання серверної компоненти приведено на прикладі встановленої та відповідним чином налаштованої ОС єми Ubuntu 16.04.N-server-amd64 LTS (Xenial Xerus).

Встановлення (інсталяція) ОС Ubuntu 16.04.N-server-amd64 LTS (Xenial Xerus) не розглядається.

1. Авторизація у операційній системі

login as:
password:

2. Оновлення індексів пакетів операційної системи

Після авторизації необхідно оновити індекси пакетів ОС наступною командою:

sudo apt-get update

```

Ubuntu 16.04.2 LTS ubuntu tty1

ubuntu login: user
Password:
Last login: Mon Oct 23 14:04:27 EEST 2017 on tty1
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Могут быть обновлены 0 пакетов.
0 обновлений касаются безопасности системы.

user@ubuntu:~$ sudo apt-get update
[sudo] password for user: _

```

По завершенню процедури оновлення, система поінформує відповідним повідомленням о вдалій операції.

```

Пол:28 http://ua.archive.ubuntu.com/ubuntu xenial/universe Translation-en [4.354 kB]
Пол:29 http://ua.archive.ubuntu.com/ubuntu xenial/multiverse amd64 Packages [144 kB]
Пол:30 http://ua.archive.ubuntu.com/ubuntu xenial/multiverse i386 Packages [140 kB]
Пол:31 http://ua.archive.ubuntu.com/ubuntu xenial/multiverse Translation-ru [83,6 kB]
Пол:32 http://ua.archive.ubuntu.com/ubuntu xenial/multiverse Translation-en [106 kB]
Пол:33 http://ua.archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [642 kB]
Пол:34 http://ua.archive.ubuntu.com/ubuntu xenial-updates/main i386 Packages [610 kB]
Пол:35 http://ua.archive.ubuntu.com/ubuntu xenial-updates/main Translation-en [269 kB]
Пол:36 http://ua.archive.ubuntu.com/ubuntu xenial-updates/restricted amd64 Packages [7.972 B]
Пол:37 http://ua.archive.ubuntu.com/ubuntu xenial-updates/restricted i386 Packages [7.988 B]
Пол:38 http://ua.archive.ubuntu.com/ubuntu xenial-updates/restricted Translation-en [2.692 B]
Пол:39 http://ua.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [540 kB]
Пол:40 http://ua.archive.ubuntu.com/ubuntu xenial-updates/universe i386 Packages [516 kB]
Пол:41 http://ua.archive.ubuntu.com/ubuntu xenial-updates/universe Translation-en [220 kB]
Пол:42 http://ua.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [15,3 kB]
Пол:43 http://ua.archive.ubuntu.com/ubuntu xenial-updates/multiverse i386 Packages [14,5 kB]
Пол:44 http://ua.archive.ubuntu.com/ubuntu xenial-updates/multiverse Translation-en [7.544 B]
Пол:45 http://ua.archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [4.860 B]
Пол:46 http://ua.archive.ubuntu.com/ubuntu xenial-backports/main i386 Packages [4.852 B]
Пол:47 http://ua.archive.ubuntu.com/ubuntu xenial-backports/main Translation-en [3.220 B]
Пол:48 http://ua.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [5.896 B]
Пол:49 http://ua.archive.ubuntu.com/ubuntu xenial-backports/universe i386 Packages [5.896 B]
Пол:50 http://ua.archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en [3.060 B]
Получено 29,2 МБ за 10с (2.749 КБ/с)
Чтение списков пакетов... Готово
user@ubuntu:~$ _

```

Примітка: Для коректного виконання цієї команди обов'язково повинен бути вказаний DNS-сервер у мережевих налаштуваннях інтерфейсів.

Пор. № зміни	Підпис відпов. особи	Дата внесення

3. Встановлення необхідного пакету бібліотек (програм)

Встановлюємо необхідний пакет бібліотек (програм) за допомогою команди:

sudo apt-get install «бібліотека (програма)»

Бібліотеки:

- make
- gcc
- libgmp-dev
- bison
- flex
- g++

Програми:

- iperf3
- unzip
- wondershaper
- mc
- htop
- nload
- ntp
- nmap

Також, можливо вказати бібліотеки та програми однією рядок через інтервал

```
user@ubuntu:~$ sudo apt-get install make gcc libgmp-dev bison flex g++ iperf3 unzip wondershaper mc
htop nload ntp nmap
```

Деякі бібліотеки (програми) потребують підтвердження для її встановлення. Тому на такий запит необхідно натиснути клавішу «Y».

```
Необходимо скачать 54,9 МБ архивов.
После данной операции, объём занятого дискового пространства возрастёт на 199 МБ.
Хотите продолжить? [Д/Н] y_
```

Далі очікуємо завершення процесу встановлення даного програмного забезпечення.

Успішне виконання команди має вигляд:

Пор. № зміни	Підпис відпов. особи	Дата внесення

```

Настраивается пакет libgmpxx4ldbl:amd64 (2:6.1.0+dfsg-2) ...
Настраивается пакет libgmp-dev:amd64 (2:6.1.0+dfsg-2) ...
Настраивается пакет liblinear3:amd64 (2.1.0+dfsg-1) ...
Настраивается пакет liblua5.2-0:amd64 (5.2.4-1ubuntu1) ...
Настраивается пакет libssh2-1:amd64 (1.5.0-2ubuntu0.1) ...
Настраивается пакет libxslt1.1:amd64 (1.1.28-2.1ubuntu0.1) ...
Настраивается пакет lua-lpeg:amd64 (0.12.2-1) ...
Настраивается пакет make (4.1-6) ...
Настраивается пакет manpages-dev (4.04-2) ...
Настраивается пакет mc-data (3:4.8.15-2) ...
Настраивается пакет mc (3:4.8.15-2) ...
Настраивается пакет python-bs4 (4.4.1-1) ...
Настраивается пакет python-pkg-resources (20.7.0-1) ...
Настраивается пакет python-chardet (2.3.0-2) ...
Настраивается пакет python-six (1.10.0-3) ...
Настраивается пакет python-html5lib (0.999-4) ...
Настраивается пакет python-lxml (3.5.0-1build1) ...
Настраивается пакет unzip (6.0-2ubuntu1) ...
Настраивается пакет ndiff (7.01-2ubuntu2) ...
Настраивается пакет nload (0.7.4-1build1) ...
Настраивается пакет nmap (7.01-2ubuntu2) ...
Настраивается пакет wondershaper (1.1a-8) ...
Обрабатываются триггеры для libc-bin (2.23-0ubuntu5) ...
Обрабатываются триггеры для systemd (229-4ubuntu16) ...
Обрабатываются триггеры для ureadahead (0.100.0-19) ...
user@ubuntu:~$ _

```

4. Копіювання МШ на сервер

За допомогою ssh-з'єднання, або змонтованого носія даних копіюємо у розділ /root серверу МШ, що поставляється у вигляді архівного файлу: QUANT-UA-VPN.tar.gz

5. Видалення попередньої версії МШ (за необхідності)

Переходимо до каталогу /root за допомогою команди:

```
cd /root
```

Видаляємо каталог попередньої версії МШ за допомогою команди:

```
rm -rf /root/openswan-2.6.39
```

6. Підготовка до встановлення МШ

Розпакувати архів МШ за допомогою команди:

```
tar -xzf /root/QUANT-UA-VPN.tar.gz
```

Примітка: Каталог *openswan-2.6.39* обов'язково повинен розміщуватися у каталозі /root

7. Інсталяція МШ

7.1 Переходимо до каталогу, куди був розпакований файл QUANT-UA-VPN.tar.gz

Пор. № зміни	Підпис відпов. особи	Дата внесення

cd /root/openswan-2.6.39

```
root@ubuntu:~# cd /root/openswan-2.6.39/
root@ubuntu:~/openswan-2.6.39#
```

7.2 Вводимо команду «make clean» та чекаємо завершення її виконання.

```
root@ubuntu:~/openswan-2.6.39# make clean
```

Успішне виконання команди має вигляд:

```
done
grep: TESTLIST: No such file or directory
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/dnssec'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/scripts'
grep -v '^#' TESTLIST | while read testtype name status; \
do\
    rm -rf $name/OUTPUT; \
done
grep: TESTLIST: No such file or directory
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/scripts'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/packaging'
cat TESTLIST | while read testtype name status; \
do\
    rm -rf $name/OUTPUT;\
done
cat: TESTLIST: No such file or directory
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/packaging'
make[2]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing'
make[1]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64'
rm -rf /root/openswan-2.6.39/tmp.rpmbuild /root/openswan-2.6.39/rpms
rm -f out.*build out.*install # but leave out.kpatch
root@ubuntu:~/openswan-2.6.39#
```

7.3 Вводимо команду «make programs» та чекаємо завершення її виконання.

```
root@ubuntu:~/openswan-2.6.39# make clean
```

Успішне виконання команди має вигляд:

```
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/lib'
make[4]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/lib/libopenswan'
make[4]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/lib/libopenswan'
make[4]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/lib/libpluto'
make[4]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/lib/libpluto'
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/lib'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/utils'
make[4]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/utils/uml_netjig'
make[4]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/utils/uml_netjig'
make[4]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/utils/pcap2skb'
make[4]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/utils/pcap2skb'
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/utils'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/kliips'
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/kliips'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/pluto'
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/pluto'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/dnssec'
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/dnssec'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/scripts'
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/scripts'
make[3]: Entering directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/packaging'
true # not actually doing anything for this target, but thats OK.
make[3]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing/packaging'
make[2]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64/testing'
make[1]: Leaving directory '/root/openswan-2.6.39/OBJ.linux.x86_64'
root@ubuntu:~/openswan-2.6.39#
```

7.4 Вводимо команду «make KERNELSRC=/lib/modules/`uname -r`/build module» та чекаємо завершення її виконання.

Успішне виконання команди має вигляд:

Пор. № зміни	Підпис відпов. особи	Дата внесення


```
t-frame-pointer -fno-optimize-sibling-calls -fno-var-tracking-assignments -pg -mfentry -DCC_USING_FEN
NTRY -Wdeclaration-after-statement -Wno-pointer-sign -fno-strict-overflow -fconserve-stack -Werror=i
mplicit-int -Werror=strict-prototypes -Werror=date-time -DCC_HAVE_ASM_GOTO -include /root/openswan-2
.6.39/packaging/linux/config-all.h -DDISABLE_UDP_CHECKSUM -I/root/openswan-2.6.39/linux/include -I/r
oot/openswan-2.6.39/linux/net/ipsec/. -DIPCOMP_PREFIX -DKLIPS -Icrypto/ocf -D"KBUILD_STR(s)=#s" -D"
KBUILD_BASENAME=KBUILD_STR(ipsec.mod)" -D"KBUILD_MODULE=KBUILD_STR(ipsec)" -DMODULE -c -o /root/o
penswan-2.6.39/modobj26/ipsec.mod.o /root/openswan-2.6.39/modobj26/ipsec.mod.c
ld -r -m elf_x86_64 -T ./scripts/module-common.lds --build-id -o /root/openswan-2.6.39/modobj26/i
psec.ko /root/openswan-2.6.39/modobj26/ipsec.o /root/openswan-2.6.39/modobj26/ipsec.mod.o
make[2]: Leaving directory '/usr/src/linux-headers-4.4.0-62-generic'

=====

KLIPS26 module built successfully.
ipsec.ko is in /root/openswan-2.6.39/modobj26

-rw-r--r-- 1 root root 699440 Окт 23 14:49 ipsec.ko
text      data      bss      dec      hex filename
376163    21888    79536   477587   74993 ipsec.ko

use make mininstall as root to install it

=====

make[1]: Leaving directory '/root/openswan-2.6.39'
root@ubuntu:~/openswan-2.6.39#
```

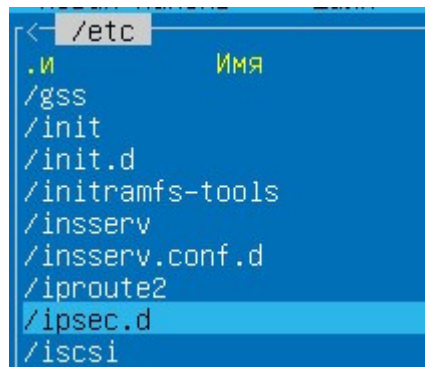
7.5 Вводимо команду «make KERNELSRC=/lib/modules/`uname -r`/build install mininstall» та чекаємо завершення її виконання.

```
root@ubuntu:~/openswan-2.6.39# sudo make KERNELSRC=/lib/modules/`uname -r`/build install mininstall
```

Успішне виконання команди має вигляд:

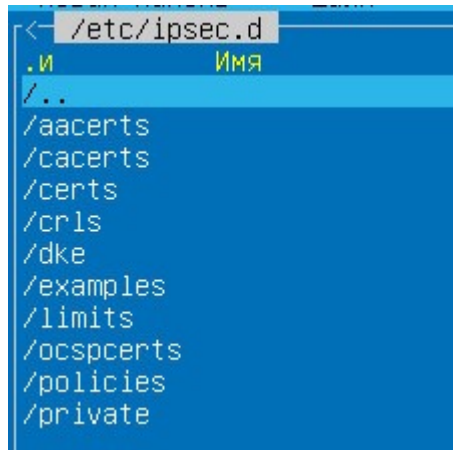
```
set -x ; \
mkdir -p $OSMODLIB/kernel/net/ipsec ; \
cp /root/openswan-2.6.39/modobj26/ipsec.ko $OSMODLIB/kernel/net/ipsec ; \
if [ -f /sbin/depmod ] ; then /sbin/depmod -a ; fi ; \
if [ -n "net/ipsec" ] ; then \
mkdir -p $OSMODLIB/kernel/net/ipsec ; \
if [ -f $OSMODLIB/kernel/ipsec.ko -a -f $OSMODLIB/kernel/net/ipsec/ipsec.ko ] ; then \
echo "WARNING: two ipsec.ko modules found in $OSMODLIB/kernel:" ; \
ls -l $OSMODLIB/kernel/ipsec.ko $OSMODLIB/kernel/net/ipsec/ipsec.ko ; \
exit 1 ; \
fi ; \
fi ; \
set -x ) ;
ls: cannot access './Documentation/DocBook/media/*_b64': No such file or directory
+ mkdir -p /lib/modules/4.4.0-62-generic/kernel/net/ipsec
+ cp /root/openswan-2.6.39/modobj26/ipsec.ko /lib/modules/4.4.0-62-generic/kernel/net/ipsec
+ '[' -f /sbin/depmod ']'
+ /sbin/depmod -a
+ '[' -n net/ipsec ']'
+ mkdir -p /lib/modules/4.4.0-62-generic/kernel/net/ipsec
+ '[' -f /lib/modules/4.4.0-62-generic/kernel/net/ipsec.ko -a -f /lib/modules/4.4.0-62-generic/kernel/ne
t/ipsec/ipsec.ko ']'
+ set -x
make[1]: Leaving directory '/root/openswan-2.6.39'
root@ubuntu:~/openswan-2.6.39#
```

Після успішного встановлення МШ у структурі розділів операційної системи додався каталог */ipsec.d*



з відповідними підкаталогами

Пор. № зміни	Підпис відпов. особи	Дата внесення

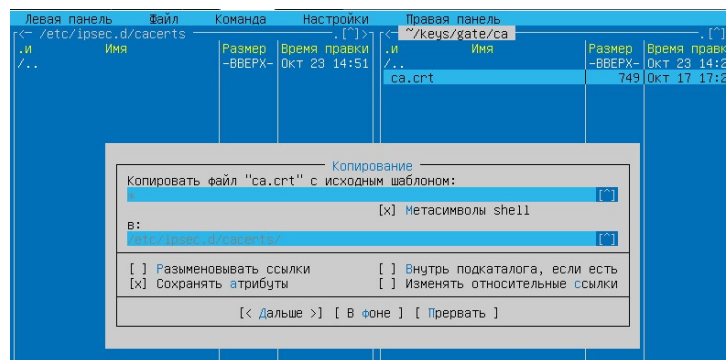


та конфігураційний файл /etc/ipsec.conf

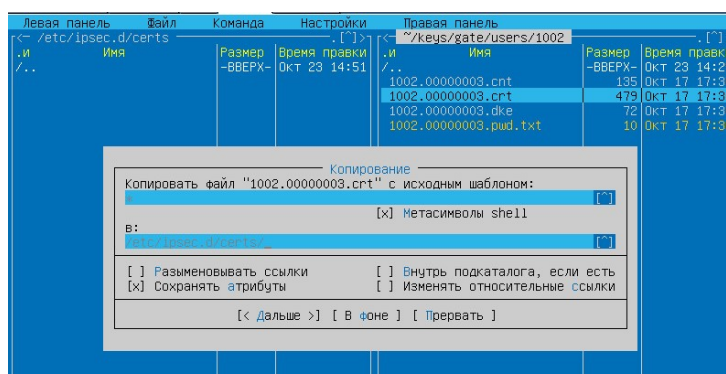
8. Завантаження ключових даних до відповідних підкаталогів

За допомогою файлового менеджера, як приклад «Midnight Commander» (mc), копіюємо ключові данні, які попередньо сформовані у ЦГК, до відповідних підкаталогів.

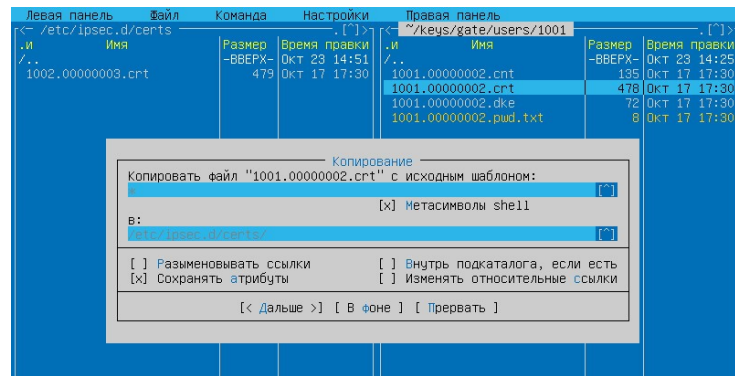
8.1 Скопіювати до підкаталогу /etc/ipsec.d/cacerts робочого каталогу МШ: кореневий сертифікат безпеки (файл з розширенням *.crt);



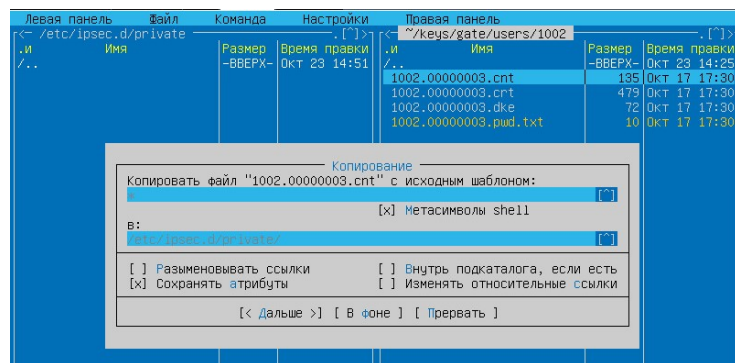
8.2 Скопіювати до підкаталогу /etc/ipsec.d/certs робочого каталогу МШ: локальний сертифікат безпеки даного МШ та сертифікати його партнерів, з якими він буде встановлювати ЗМ (файл з розширенням *.crt);



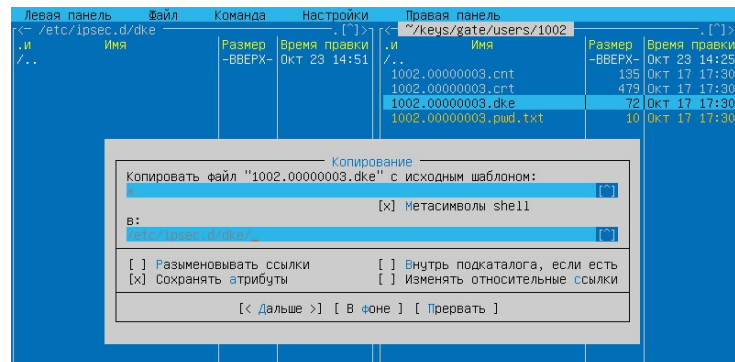
Пор. № зміни	Підпис відпов. особи	Дата внесення



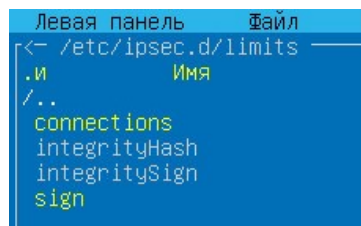
8.3 Скопіювати до підкаталогу /etc/ipsec.d/private робочого каталогу МШ: захищений контейнер локального сертифікату безпеки даного МШ (файл з розширенням *.cnt);



8.4 Скопіювати до підкаталогу /etc/ipsec/dke робочого каталогу МШ: dke-файл – файл заміни ключових даних (файл з розширенням *.dke);



8.5 Скопіювати до підкаталогу /etc/ipsec/limits робочого каталогу МШ: файли - ліцензії «connections» та «sign». Якщо схема з'єднання складається лише з двох МШ (МШ ← → МШ) – файли - ліцензії не потрібні.



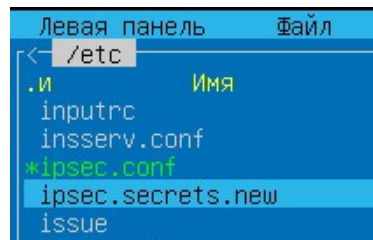
Пор. № зміни	Підпис відпов. особи	Дата внесення

9. Перший запуск МШ

Перший запуск МШ здійснюється за допомогою команди:

/etc/init.d/ipsec start

При першому запуску цієї команди формується файл `ipsec.secrets.new` в каталозі `/etc`.



Далі необхідно натиснути комбінацію клавіш «Ctrl+C», тим самим перервавши роботу програми.

Змінюємо ім'я файлу з «`ipsec.secrets.new`» на «`ipsec.secrets`» командою:

cp /etc/ipsec.secrets.new /etc/ipsec.secrets

```
root@ubuntu1604:/etc#
root@ubuntu1604:/etc# cp ipsec.secrets.new ipsec.secrets.new _
```

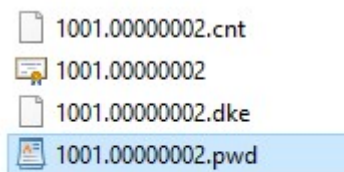
Прописуємо у файлі `./etc/ipsec.secrets` наступне:

```
: DSTU 1001.00000002.cnt "X631e8Hk"
```

, де:

1001.00000002.cnt - захищений контейнер локального сертифікату безпеки даного МШ

«X631e8Hk» – пароль до захищеного контейнера локального сертифікату безпеки даного МШ. Береться з файлу `1001.00000002.pwd`



10. Файл /etc/sysctl.conf

Вносимо зміни у файл `/etc/sysctl.conf`

Додаємо до вмісту файлу `/etc/sysctl.conf` наступні строки:

Пор. № зміни	Підпис відпов. особи	Дата внесення

```

/etc/sysctl.conf [BM--] 0 L:[ 1+ 5 6/ 80] *(166 /2579b) 0110 0x06E
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
net.ipv4.conf.default.rp_filter=1
net.ipv4.conf.all.rp_filter=1
net.ipv4.tcp_syncookies=1
net.ipv4.ip_forward=1
kernel.panic = 1
kernel.panic_on_oops = 1
kernel.panic_on_io_nmi = 1
kernel.panic_on_unrecovered_nmi = 1
net.ipv4.ip_nonlocal_bind = 1

```

11. Конфігурування модуля шифрування. Файл ipsec.conf

Типовий файл ipsec.conf має вигляд:

```

/etc/ipsec.conf [----] 0 L:[ 1+ 0 1/ 33] *(0 / 92
version 2.0
config setup
    dumpdir=/var/run/pluto/
    nat_traversal=yes
    virtual_private=%v4:10.0.0.0/8,%v4:20.0.0.0/8,%v4:
    protostack=klips
    interfaces="ipsec0=ens160"
    plutostderrlog=/var/log/pluto.log
    plutodebug = "none"
    klipsdebug = "none"
    gostbox_path=/etc/ipsec.d/dke/1001.00000002.dke
    gostbox_pass=X631e8HK

conn sample
    authby=dstusig
    left=192.168.50.80
    leftsubnet=10.10.10.0/24
    leftCert=1001.00000002.crt
    leftid=%fromcert
    right=192.168.50.75
    rightsubnet=20.20.20.0/24
    rightCert=1002.00000003.crt
    rightca=%same
    forceencaps=no
    pfs=no
    auto=add
    salifetime=24h
    ikelifetime=24h
    dpddelay=20
    dpdtimeout=30
    dpdaction=clear
    rightid=%fromcert

```

, де:

- Інформаційний блок:

Пор. № зміни	Підпис відпов. особи	Дата внесення

В цьому блоці зазначається номер версії конфігураційного файлу:

version 2.0	Версія специфікації «ipsec.conf»
--------------------	----------------------------------

Блок представлений у файлі налаштувань «ipsec.conf» в єдиному екземплярі.

- Блок загальних налаштувань:

Зазначений блок містить загальну політику налаштування, яка буде відноситись до всіх політик з'єднань розміщених в файлі налаштувань «ipsec.conf».

dumpdir=/var/run/pluto/	????
nat_traversal=yes	Використання «NAT-TRAVERSAL» необхідно для проходження пакетів UDP через NAT
virtual_private=%v4:10.0.0.0/8,%v4:192.168.0.0/16	Опис дозволених віртуальних підмереж для IKEcfg-адаптерів VPN-партнерів

:

Зазначений блок містить загальну політику налаштування, яка буде відноситись до всіх політик з'єднань розміщених в файлі ipsec.conf.

basic configuration

config setup

початок блоку

Do not set debug options to debug configuration issues!

Do not set debug options to debug configuration issues!

plutodebug / klipsdebug = "all", "none" or a combination from below:

"raw crypt parsing emitting control klips pfkey natt x509 dpd private"

eg:

plutodebug="control parsing"

Again: only enable plutodebug or klipsdebug when asked by a developer

#

enable to get logs per-peer

plutoopts="--perpeerlog"

Note: incorrect SELinux policies might prevent pluto writing the core

dumpdir=/var/run/pluto/

#

nat_traversal=yes

#

Використання NAT-TRAVERSAL необхідно для проходження пакетів UDP через NAT

virtual_private=%v4:10.0.0.0/8,%v4:192.168.0.0/16

#

Опис дозволених віртуальних підмереж для IKEcfg-адаптерів VPN-партнерів

protostack=klips

#

визначення зовнішнього інтерфейсу МШ при побудові ЗМ, де ens160 - логічне ім'я зовнішнього інтерфейсу

interfaces="ipsec0=ens160"

#

Use this to log to a file, or disable logging on embedded systems

#plutostderrlog=/dev/null

plutodebug = "none"

Визначення параметрів (рівня)

klipsdebug = "none"

#

журналювання. Можливі значення plutodebug, klipsdebug - "none", "all"

plutostderrlog=/var/log/pluto.log

#

Місце розміщення та назва файлу логування

gostbox_path=/etc/ipsec/1001.00000002.dke

#

Назва та шлях до розміщення ДКЕ-файлу для перевірки сертифікатів

gostbox_pass=X631e8Hk

#

Пароль від локального сертифікату безпеки МШ

Блок представлений у конфігураційному файлі ipsec.conf в єдиному екземплярі.

Пор. № зміни	Підпис відпов. особи	Дата внесення

- Блок налаштування певної ділянки ЗМ:

```

conn sample                # "sample" довільна назва ЗМ
authby=dstusig             # використання сертифікатів сформованих згідно ГОСТу для
                             аутентифікації партнерів
forceencaps=yes            # режим примусової UDP інкапсуляції ESP-пакетів
dpddelay=30                # Dead peer detection - 30 секунд - інтервал між
                             keepalive пакетами
dpdtimeout=120             # Dpd-таймаут 120 секунд, після якого партнер буде
                             вважатися недосяжним
dpdaction=restart_by_peer  # після інтервалу dpdtimeout МШ буде намагатися знову
                             побудувати з'єднання (можливі значення - none, clear,
                             route, restart_by_peer)
left=192.168.50.80         # IP-адреса зовнішнього інтерфейса локального МШ
leftsubnet=10.10.10.0/24   # захищена мережа локального МШ (внутрішня захищена
                             підмережа МШ)
leftCert=1001.00000002.crt # ім'я локального сертифікату цього шлюзу (розташований в
                             каталозі /etc/ipsec.d/cert даного МШ)
                             IP-адреса зовнішнього інтерфейса віддаленого партнера.
right=192.168.50.75        # Значення "%any" використовується при підключенні
                             клієнтів із не визначеними будь-якими) зовнішніми IP
                             адресами
rightsubnet=20.20.20.0/24 # захищена мережа партнера (внутрішня IP мережа партнера,
                             для VPN-партнерів використовується значення
                             "vnet:%no,%priv")
rightCert=1002.00000003.crt # назва сертифікату VPN-партнера (який знаходиться в
                             каталозі /etc/ipsec.d/cert даного МШ)
auto=add                   # add - загрузити конфігурацію з'єднання та відповідати
                             на приходячі запити (пасивний режим роботи),
                             start - загрузити конфігурацію з'єднання та посилати
                             запити на створення з'єднання (активний режим роботи)
salifetime=24h             # час життя одного сеансового ключу IPSEC з'єднання
pfs=no                     # не використовувати значення pfsgroup, при включенні pfs
                             (yes) значення pfsgroup береться із фази 1 IKE обміну
ikelifetime=24h            # час встановлення фази IKE
salifetime=1m              # час підтримки з'єднання без перевстановлення
leftid=%fromcert           #
rightca=%same              # використовувати перевірку у партнера на кореневий
                             сертифікат

```

Примітка: Один з двох серверів, який є складовою VPN-з'єднання, у файлі `ipsec.conf` повинен мати параметр `auto=add`, інший `auto=start`.

12. Запуск МШ

Запуск модуля шифрування здійснюється командою

`/etc/init.d/ipsec start`

```

root@VERSIA:~# /etc/init.d/ipsec start
<27>Nov  3 15:00:51 ipsec_setup: Starting SPEKTR IPsec kernel...
<27>Nov  3 15:00:51 ipsec_setup: ipsec0 -> NULL mtu=0(0) -> 0
set: wondershaper ipsec0 2048576 2048576
root@VERSIA:~#

```

Після запуску МШ в системі формується додатковий логічний інтерфейс `ipsec0` IP-адреса якого співпадає з IP-адресою зовнішнього інтерфейсу.

Пор. № зміни	Підпис відпов. особи	Дата внесення

```

ipsec0    Link encap:Ethernet  HWaddr 00:0e:c4:cd:86:86
          inet addr:192.168.50.80  Mask:255.255.255.255
          inet6 addr: fe80::20e:c4ff:fe86:8686/128 Scope:Link
          UP RUNNING NOARP  MTU:1500  Metric:1
          RX packets:8489 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8489 errors:0 dropped:2 overruns:0 carrier:0
          collisions:0 txqueuelen:10
          RX bytes:543296 (543.2 KB)  TX bytes:1273350 (1.2 MB)

```

13. Типові команди для адміністрування МШ

13.1 Після внесення будь-яких змін в конфігурацію МШ (файл `ipsec.conf`), потрібно перезавантажити службу «ipsec». Це виконується за допомогою команди:

/etc/init.d/ipsec restart

Примітка: При перезавантаженні даної служби усі побудовані захищені з'єднання будуть перезавантажені.

Для ручного запуску захищеного з'єднання з заданою політикою необхідно.

Для тимчасового зупинення служби, використовується команда:

/etc/init.d/ipsec stop

та набрати:

ipsec auto --up sample, де *sample* – це назва політики з'єднання.

13.2 Команди перевірки стану VPN-з'єднання:

- *sudo ipsec verify*
- *sudo ipsec auto --status*
- *sudo ipsec eroute*

```

root@MINI:~# ipsec eroute
866      10.10.10.0/24      -> 20.20.20.0/24      => tun0x1007@192.168.50.79

```

- *ping «IP-адреса №1» -I «IP-адреса №2»*

IP-адреса №1 - IP-адреса локального інтерфейсу МШ

IP-адреса №2 - IP-адреса локального інтерфейсу партнера МШ

```

root@VERSIA:~# ping 10.10.10.1 -I 20.20.20.1
PING 10.10.10.1 (10.10.10.1) from 20.20.20.1 : 56(84) bytes of data.
64 bytes from 10.10.10.1: icmp_seq=1 ttl=64 time=0.540 ms
64 bytes from 10.10.10.1: icmp_seq=2 ttl=64 time=0.545 ms
^C

```

Пор. № зміни	Підпис відпов. особи	Дата внесення

14. Автоматичний запуск МШ після завантаження/перезавантаження операційної системи.

14.1 Створюємо скрипт-файл запуску МШ в каталозі /etc/init.d з відповідними правами доступу:

```
touch /etc/init.d/ipsec_auto.sh
chmod 755 /etc/init.d/ipsec_auto.sh
chown root /etc/init.d/ipsec_auto.sh
```

14.2 Вносимо у скрипт-файл наступну інформацію:

```
/etc/init.d/ipsec_auto.sh
#!/bin/sh -e

/etc/init.d/ipsec start

exit 0
```

14.3 Створюємо у командному рядку link-посилання на скрипт-файл

```
ln -s /etc/init.d/ipsec_auto.sh /etc/rc2.d/S03ipsec_auto
```

15. Використання та активація захищеного носія ключової інформації "ЕФІТ КЕЙ" (EfitKey)

15.1 Для використання захищеного носія ключової інформації "ЕФІТ КЕЙ" (EfitKey) у операційній системі Linux необхідно мати носій версія якого не нижче 4.16.

15.2 Для Linux i686x64 існує скомпільована версія бібліотеки libefitkeynxt.so

15.3 Встановлення необхідних специфікацій для роботи бібліотеки libefitkeynxt.so та носія EfitKey

15.2.1 Зв'язок з захищеним носієм ключової інформації у операційній системі Linux забезпечується за допомогою набору відповідних специфікацій (pcsc та libccid). Специфікації регламентують програмний інтерфейс користувача з однієї сторони та програмним інтерфейс драйверів захищених носіїв з іншої сторони. Для їх встановлення необхідно виконати наступні команди:

```
sudo apt-get install pcscd
sudo apt-get pcsc-tools
sudo apt-get libccid
sudo apt-get libpcsclite1
```

Примітка: Рекомендовано додати поточного користувача у групу pcscd для керування демоном pcscd:

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
sudo adduser root pcscd
```

15.2.2 Встановлення пакету OpenSC

Пакет OpenSC є набором бібліотек та утіліт для роботи з захищеними носіями ключової інформації, які підтримують криптографічні операції.

Для його встановлення необхідно виконати наступну команду:

```
sudo apt-get install opensc
```

15.4 Додавання захищеного носія EfitKey до списку pcsc-пристроїв що підтримуються системою.

Для того щоб додати (включити) існуючий pcsc-пристрій (захищеного носія EfitKey) до відповідного списку pcsc-пристроїв, що підтримуються системою, необхідно скорегувати файл Info.plist

```
/usr/lib/pcsc/drivers/ifd-ccid.bundle/Contents/Info.plist
```

а саме здійснити наступні кроки:

15.3.1 Знайти строку "<key>ifdVendorID</key>" і після елементу "<array>" додати "<string>0xC1A6</string>"

15.3.2 Знайти строку "<key>ifdProductID</key>" і після елементу "<array>" додати "<string>0x0151</string>"

15.3.3 Знайти строку "<key>ifdFriendlyName</key>" і після елементу "<array>" додати "<string>Efit Technologies EfitKey</string>"

15.5 Скопіювати бібліотеку libefitkeynxt.so до каталогу /usr/lib.

15.6 Зконфігурувати програмний модуль «Модуль шифрування» для роботи носієм EfitKey.

Редагуємо файл ipsec.secrets:

```
: DSTU EFK4160030085 «12345678»
```

, де:

EFK4160030085 – серійний номер захищеного носія ключової інформації "ЕФІТ КЕЙ";

12345678 – пароль (pin code) для доступу до захищеного носія ключової інформації "ЕФІТ КЕЙ".

Редагуємо файл ipsec.conf:

```
gostbox_path=/etc/ipsec.d/dke/1001.00000002.dke
```

```
gostbox_pass=12345678
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

15.7 Запуск демона смарткарт.

Для роботи libefitkeynxt.so и EfitKey необхідно мати запущений демон pcsd.

15.8 Під'єднуємо захищений носій ключової інформації "ЕФІТ КЕЙ"

16. Використання ключових даних наданих АЦСК

На відміну від дворівневої класичної моделі роботи з центром генерації ключів (ЦГК), модель роботи з акредитованим центром сертифікації ключів (АЦСК) є трирівнева:

- сертифікат центрального засвідчуваного органу ЦЗО (root certificate),
- сертифікат АЦСК (subroot certificate)
- сертифікат серверу (local certificate).

Для реалізації трирівневої моделі необхідно створити нову директорію «intermediatecacerts»

```
mkdir /etc/ipsec.d/intermediatecacerts
```

```
chmod 755 /etc/ipsec.d/intermediatecacerts
```

```
chown root /etc/ipsec.d/intermediatecacerts
```

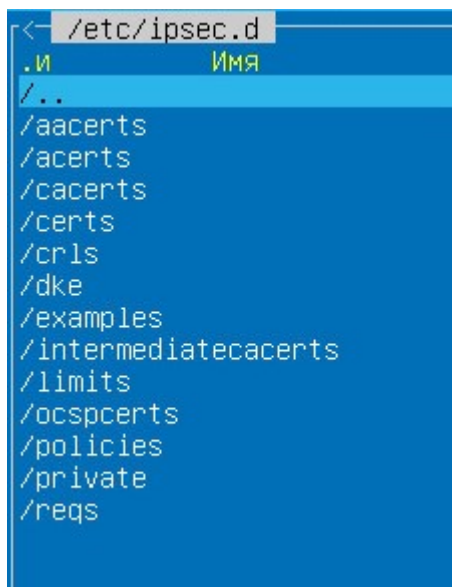


Схема розподілення ключових даних наступна:

/etc/ipsec.d/cacerts – містить сертифікат центрального засвідчуваного органу (ЦЗО);

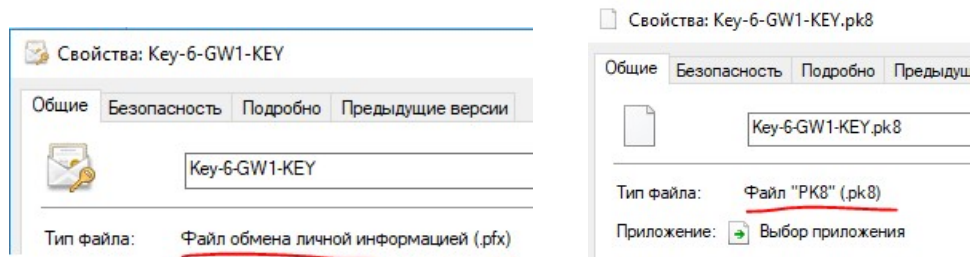
/etc/ipsec.d/intermediatecacerts - містить сертифікат акредитованого центра сертифікації ключів (АЦСК);

/etc/ipsec.d/certs – містить сертифікати локального серверу і серверів клієнтів.

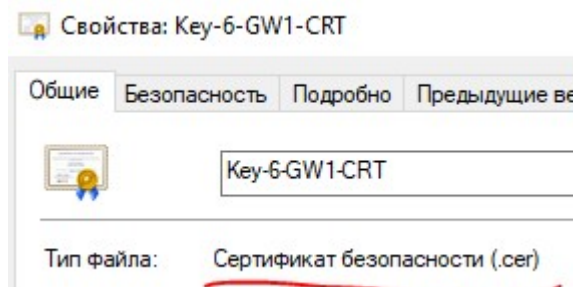
Пор. № зміни	Підпис відпов. особи	Дата внесення

17. Використання ключових даних сформованих іншими стандартами криптографії (PFX та PKCS8)

Зовнішні відмінності ключових даних цих стандартів тільки у типі (файлових розширеннях) файлів закритого ключа:



Сертифікати відкритого ключа обох стандартів, як правило мають тип (файлові розширення) *.cer



Під час конфігурування серверної компоненти МШ файли відкритих ключів (*.crt, *.cer) копіюються у підкаталог /etc/ipsec.d/certs , а файли закритого ключа (*.cnt, *.pk8, *.pfx) - /etc/ipsec.d/private

18. Протокол подій роботи МШ

Протокол подій роботи МШ відображається у файлі /var/log/pluto.log



Пор. № зміни	Підпис відпов. особи	Дата внесення

Для відображення подій роботи МШ у режимі реально часу, необхідно в командному рядку виконати команду:

```
tail -f /var/log/pluto.log
```

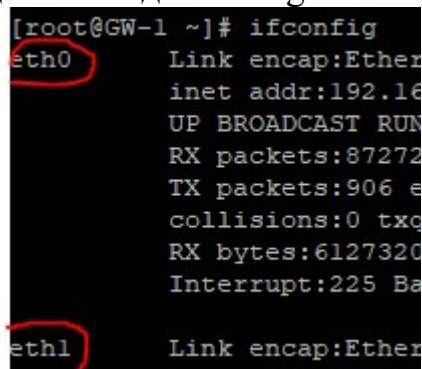
19. Приклад конфігураційних файлів

Для прикладу візьмемо два сервери (GW1, GW2) та налаштуємо захищений канал зв'язку між LAN мережами серверів.

На GW1 мережеві інтерфейси налаштовано на IP-адреси WAN 192.168.100.1/24 та LAN 192.168.1.1/24.

На GW2 мережеві інтерфейси налаштовано на IP-адреси WAN 192.168.100.2/24 та LAN 10.100.1.1/24. GW2 є ініціатором створення захищеного каналу зв'язку ().

На обох серверах захищений канал зв'язку створюється між WAN інтерфейсами, які в операційній системі мають логічне ім'я "eth1". На серверах різних виробників логічні ім'я можуть мати різний вигляд. Використовувати необхідно такий запис, який видає команда ifconfig



```
[root@GW-1 ~]# ifconfig
eth0      Link encap:Ether
          inet addr:192.16
          UP BROADCAST RUN
          RX packets:87272
          TX packets:906 e
          collisions:0 txq
          RX bytes:6127320
          Interrupt:225 Ba
eth1      Link encap:Ether
```

GW1. Конфігураційний файл ipsec.conf

```
version 2.0
config setup
    dumpdir=/var/run/pluto/
    nat_traversal=yes
    virtual_private=%v4:10.100.1.0/24,%v4:192.168.1.0/24
    protostack=klips
    interfaces="ipsec0=eth1"
    gostbox_path=/etc/ipsec.d/dke/1001.00000002.dke
    gostbox_pass=7EoZ31s6
    plutostderrlog=/var/log/pluto.log
    plutodebug ="none"
    klipsdebug ="none"
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

```

conn gate2
    authby=dstusig
    left=192.168.100.1
    leftsubnet=192.168.1.0/24
    leftCert=1001.00000002.crt
    leftid=%fromcert
    right=192.168.100.2
    rightsubnet=10.100.1.0/24
    rightCert=1002.00000003.crt
    rightca=%same
    forceencaps=no
    pfs=no
    auto=add
    salifetime=24h
    ikelifetime=24h
    dpddelay=20
    dpdtimeout=30
    dpdaction=clear
    rightid=%fromcert

```

GW1. Конфігураційний файл ipsec.secrets

```
: DSTU      1001.00000002.cnt "7EoZ31s6"
```

GW2. Конфігураційний файл ipsec.conf

```

version 2.0
config setup
    dumpdir=/var/run/pluto/
    nat_traversal=yes
    virtual_private=%v4:10.100.1.0/24,%v4:192.168.1.0/24
    protostack=klips
    interfaces="ipsec0=eth1"
    gostbox_path=/etc/ipsec.d/dke/1002.00000003.dke
    gostbox_pass=435gjjH7
    plutostderrlog=/var/log/pluto.log
    plutodebug ="none"
    klipsdebug ="none"

conn gate1
    authby=dstusig
    left=192.168.100.2
    leftsubnet=10.100.1.0/24
    leftCert=1002.00000003.crt
    leftid=%fromcert
    right=192.168.100.1
    rightsubnet=192.168.1.0/24
    rightCert=1001.00000002.crt
    rightca=%same
    forceencaps=no
    pfs=no
    auto=start
    salifetime=24h

```

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
ikelifetime=24h  
dpddelay=20  
dpdtimeout=30  
dpdaction=clear  
rightid=%fromcert
```

GW2. Конфігураційний файл ipsec.secrets

```
: DSTU      1002.000000003.cnt "435gjjH7"
```

АДМІНСТРУВАННЯ КОНТЕНТУ

З метою забезпечення роботи Серверу необхідно періодично завантажувати на Сервер наступний контент:

1. Посилені сертифікати відкритих ключів користувачів – періодично у разі виникнення такої необхідності.
2. Планова або позапланова заміна особистого ключа ЕЦП та/або шифрування Серверу – у разі виникнення такої необхідності.
3. Перегляд файлів-протоколів роботи Сервера – періодично або у разі виникнення такої необхідності.

ПРИМІТКИ АДМІНІСТРАТОРУ

При здійсненні експлуатації та адміністрування Серверу необхідно дотримуватися загальноприйнятих правил, щодо забезпечення безпеки адміністрування інформаційних систем.

Пор. № зміни	Підпис відпов. особи	Дата внесення