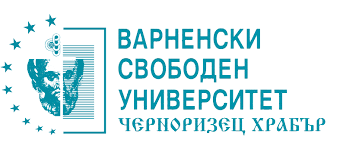
****

|  |
| --- |
| **Курсова работа** |
|
| **Компютърни мрежи и комуникации** |

*Изготвила:  
Мария Николаева Станчева – 163011014  
Информатика и Компютърни науки*

# 

# 

# 

# 

# 

# 

# A picture containing computer, drawing, table Description automatically generated

# Описание на задачата

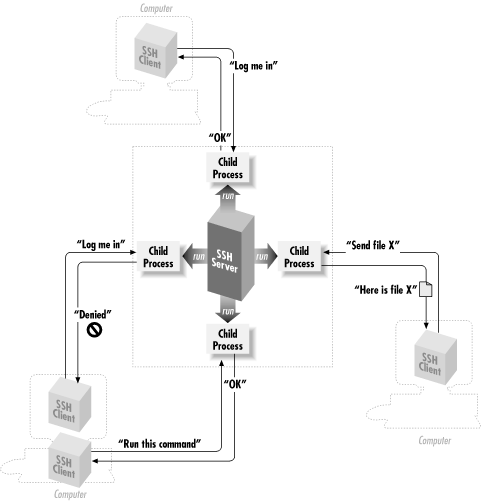
Подготовка на проект за изграждане на мрежа за връзка на минимум 3 обекта с описано хардуерно и софтуерно обезпечаване.

Целта на задачата е да се изгради компютърна мрежа в която от крайно устройство, като например лаптоп или персонален компютър, да може да се осъществи достъп до комутатор (switch) в мрежата посредством SSH протокол.

# Описание на SSH (Secure Shell)

Мрежов протокол, който позволява криптирано предаване на данни. За разлика от по-ранни протоколи, като например Telnet (Teletype Network), предаваната информация е изцяло криптирана, тоест не се изпраща нищо под формата на plain-text. Това прави протокола сигурен и е причината да е широко използван за отдалечен достъп до устройства свързани в незащитени мрежи.

Протоколът използва така нареченото „прозрачно криптиране“ (transparent encryption). Данните се криптират автоматично преди изпращането им от компютърното устройство към мрежата и се декриптират когато пристигнат при получателя. Криптирането и декриптирането си извършват автоматично от самия SSH протокол, което прави работата му „прозрачна“ от гледна точка на потребителите.



*Фиг. 1 Схематичен модел на SSH архитектура*

SSH използва клиент-сървър архитектура. Сървърната програма се инсталира на избран хост, и отговаря за обработката ( отхвърлянето или приемането ) на дадени заявки, изпращани от клиентските SSH програми, инсталирани при крайните потребители. Комуникацията помежду им е изцяло криптирана и не подлежи на модификация.

Съставен е от три основни компонента (протокола):

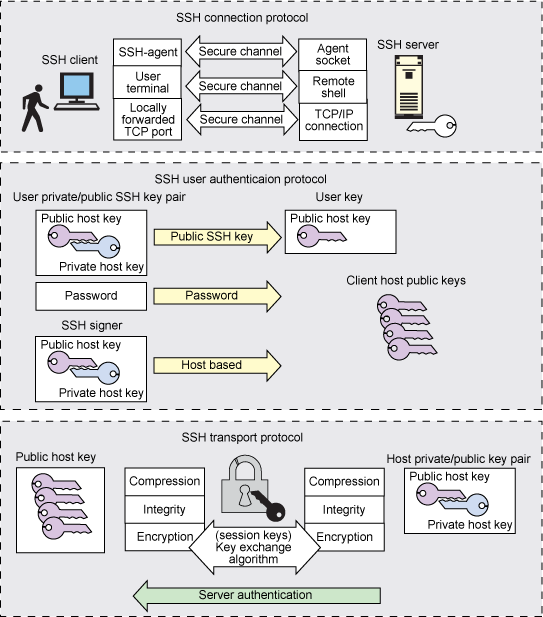
* ***Транспортен протокол*** – отговаря за първоначалната автентикация на ниво сървър, чрез ключ, съдържащ две части - публична и частна. Също така криптира и компресира данните.
* ***Потребителски идентификационен* *протокол*** – отговаря за оторизацията на потребители. Оторизацията е „задвижвана от клиента“, тоест клиентската SSH програма изисква оторизиращата информация, а сървъра единствено отговаря на изпратените заявки. Най-честите методи за оторизация са парола или публичен ключ.
* ***Мрежов протокол*** – организира логическите връзки между различните комуникационни канали, които се използват от дадена SSH връзка.

## Ключове

Криптирането на данните се извършва от шифроващи алгоритми, които използват малки части информация, наречени *ключове*, като криптиращи параметри. Чрез тях алгоритъма гарантира, че за всяка част криптирана информация има точно един ключ, който може да я декриптира. Ключовете се делят на два вида – асиметрични (публичен) и симетрични (секретен). Всеки асиметричен ключ е съставен от два компонента, публичен и частен.

SSH използва следните типове ключове:

* **Потребителски ключ** – постоянен, асиметричен ключ използван за потребителска идентификация. Потребителите може да имат повече от един ключ.
* **Хост ключ** – постоянен асиметричен ключ използван за идентифициране на сървъра. В случай, че дадена машина управлява точно един SSH сървър, то хост ключа играе и ролята на уникален идентификатор на машината. Ако сървърите са повече от един, то всеки от тях може да има различен ключ, или да използват споделен.
* **Сесиен ключ** – произволно-генериран, временен симетричен ключ използван за криптиране на връзката между SSH клиент и сървър. Генерира се по време на установяване на връзка и след това се споделя между двете страни, което гарантира, че само те ще могат да разшифроват предаваната информация. След прекратяване на връзката, ключът се унищожава.



*Фиг. 2 Видове ключове*

Ключовете се съдържат в хост база данни, която се използва от клиента и сървъра по време на идентификация. Генерацията на постоянни ключове се извършва от специализирани за тази цел програми, като най-често срещаната е ssh-keygen.

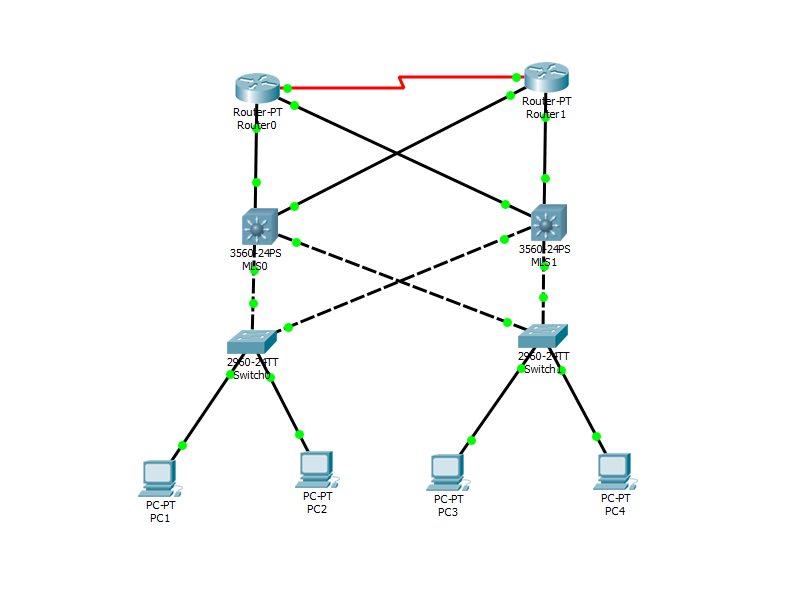
# Описание на мрежата

Мрежата е изградена съгласно йерархичния модел на Cisco. Устройствата са разделение на три слоя:

• **Слой за достъп (Access layer):** Чрез него се осъществява достъпът на потребителите до мрежовите устройства. В примерната мрежа този слой е представен от комутаторите Switch0 и Switch1.

• **Разпределителен слой (Distribution layer):** Агрегира данните събрани от комутаторите в слоя за достъп и ги изпраща към слоя на ядрото. Използва се за разделяне на потребители на работни групи. При възникване на проблем в даден комутатор, работата на мрежата не се нарушава. Слоя е представен от многослойните комутатори MLS0 и MLS1 в примерната мрежа.

• **Слой на ядрото (Core layer):** Устройствата в този слой предоставят услуги, които оптимизират транспортните потоци в мрежата. В допълнение от тях се очаква да осигурят максимална наличност и надеждност с минимална обработка на пакетите. Устройствата от ядрото трябва да могат да поддържат връзката, дори когато някои от свързващите ги вериги са прекъснати. В примерната мрежа това са маршрутизаторите Router0 и Router1.



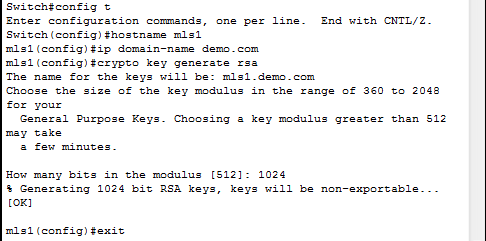
*Фиг. 3 Архитектура на примерната мрежа*

# Конфигуриране на мрежата

За да бъде изпълнена поставената задача, върху избраното устройство трябва да бъдат извършени няколко настройки. В примерната мрежа представена в горния раздел, тази конфигурация може да бъде приложена както върху комутаторите от слоевете за достъп (Switch0, Switch1) и разпределение (MLS0, MLS1), така и върху маршрутизаторите от слоя на ядрото (Router0, Router1) .

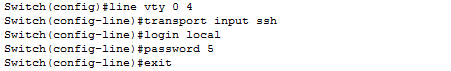
Необходим е директен достъп до даденото устройство, като за тази цел се използва конзолен кабел. Преди да се конфигурира отдалечения достъп е необходимо устройството да има правилно зададени мрежови настройки. След което се изпълняват следните последователни стъпки:

1. Задаване на хост име (hostname) и домейн. Хост името служи за идентификация на устройството, а домейнът е необходим за да е възможно генерирането на ключове. Задават се с команда ***hostname***и ***domain-name****.*
2. Генериране на RSA ключове. Необходими са за криптирането и декриптирането на информацията предавана през SSH. Генерират се с команда ***crypto*** ***key generate rsa***.



*Фиг. 4 Стъпка 1 и 2*

1. Настройване на VTY (Virtual Teletype Terminal) линиите. Задава са SSH като желан метод за комуникация. По този начин ще бъдат изключени другите възможности за свързване, като например Telnet. Не е задължително да се задава само един метод за свързване, но това подобрява сигурността на мрежата.



*Фиг. 5 Стъпка 3*

С командата ***line vty 0 4*** избираме VTY линии от 0 до 4 (това са всички зададени фабрично на повечето устройства).

***transport input ssh*** указва SSH протокола като метод на комуникация.

Командата ***login local*** означава, че ще използваме потребителско име и парола, запазени в локалната база данни. Необходимо е да се използва, защото SSH ги изисква, за да работи. С ***password 5*** задаваме типа алгоритъм за криптиране на пароли (стандартния е 7, с 5 използваме по-силния md5) .

1. Задаване на потребителско име и парола. Ще бъдат изискани с цел оторизация по време на свързване с устройството.



*Фиг. 6 Стъпка 4*

Изключително важно е да се подсигурим, че настройката за криптиране на пароли, **password-encryption**, е включена. По този начин гарантираме, че всички пароли, които за записани в конфигурационни файлове ще бъдат криптирани и така повишаваме сигурността на мрежата.

1. Проверяваме дали SSH е правилно настроен, чрез командата ***show ip ssh***.

**

*Фиг. 7 Стъпка 5*

Това са базовите настройки, които са необходими да се осъществи SSH достъп до желан комутатор или маршрутизатор от мрежата. Посредством командата ***ip ssh*** могат да бъдат приложени и допълнителни настройки, като например разрешен брой опити за автентикация, timeout и др.

След това, устройството ще бъде достъпно от всеки краен потребител, намиращ се в същата мрежа, при представяне на валидно потребителско име и парола.