УДК 004.056; 004.057.4:004.772

Скопа О. О., д.т.н. *(Одеський національний економічний університет)*
Пархуць Л. Т., д.т.н. *(Національний університет «Львівська політехніка»)*

КОМПЛЕКСНА ПРОТИДІЯ ВРАЗЛИВОСТЯМ ПРОТОКОЛУ FTP

Скопа О. О., Пархуць Л. Т. Комплексна протидія вразливостям протоколу FTP. Розглядаються аспекти роботи протоколу FTP. Аналізуються та класифікуються віддалені атаки на нього. Пропонуються заходи протидії вразливостям протоколу FTP.
Ключові слова: FTP- ПРОТОКОЛ, КРИПТОСТІЙКІСТЬ, АТАКА, FTP-BOUNCE, ВРАЗЛИВІСТЬ, ФАЙЛ

Скопа А. А., Пархуць Л. Т. Комплексное противодействие уязвимостям протокола FTP. Рассматриваются аспекты работы протокола FTP. Анализируются и классифицируются удаленные атаки на него. Предлагаются меры противодействия уязвимостям протокола FTP.
Ключевые слова: FTP- ПРОТОКОЛ, КРИПТОСТОЙКОСТЬ, АТАКА, FTP-BOUNCE, УЯЗВИМОСТЬ

Skopa O. O., Parkhuts' L. T. Complete resistance to vulnerability of FTP-protocol. The function aspects of FTP-protocol are examined. Analysed and classified remote attacks on him. The measures of counteraction of vulnerabilities FTP-protocol are offered.
Key words: FTP, RELIABILITY, ATTACK, FTP-BOUNCE, VULNERABILITY, PROTOCOL, PORT

Постановка проблеми. Аспекти безпеки протоколу FTP пов'язані з його широким використанням в комп'ютерних мережах. В даний час протокол є одним з двох найбільш популярних засобів обміну файлами. Проблеми його безпеки обумовлені часом ухвалення комунікаційного стандарту на нього та особливостями його роботи. З часу створення протокол жодного разу не доопрацьовувався з урахуванням вимог безпеки, хоча рекомендації з цього питання були випущені. З метою отримання загального уявлення про проблеми даного протоколу приведемо основні відомості про нього, взяті з [1...5].

FTP (*File Transfer Protocol* – Протокол передавання файлів) – це протокол, який створений для передавання файлів у інформаційній (комп'ютерній) мережі [2]. Однією з функцій протоколу FTP є можливість підключення до віддалених серверів, перегляду та зміни вмісту каталогів, відправки файлів на сервер або копіювання їх з сервера. По протоколу FTP можна передавати файли між серверами.

Протокол FTP відноситься до протоколів прикладного рівня [6] і для передавання даних використовує транспортний протокол TCP, тобто він забезпечує взаємодію з мережею додатків, призначених для користувача. Команди та дані, на відміну від більшості інших протоколів, передаються по різних портах (рис 1).

Висхідний порт 20, що відкривається на стороні сервера, використовується для передавання даних, а порт 21 – для передавання команд. Порт для прийому даних клієнтом визначається в процесі діалогу узгодження. У випадку, якщо передавання файлу було перерване з будь-яких причин, протокол передбачає засоби для докачки файлу, що є досить актуальною проблемою при передаванні великих файлів [4].

Основною проблемою безпеки протоколу є той факт, що в протоколі не передбачене шифрування інформації і, відповідно, при аутентифікації логін і пароль передаються відкритим текстом. У разі побудови мережі з використанням хабів, зловмисник може перехоплювати логіни і паролі користувачів FTP, що знаходяться в тому ж сегменті мережі, або, за наявності спеціального програмного забезпечення, отримувати передаванні по FTP файли без авторизації. При побудові мережі на світчах завдання зловмисника ускладнюється, але злом все одно можливий. Щоб запобігти перехопленню трафіку, необхідно використовувати протокол шифрування даних SSL, який підтримується багатьма сучасними FTP-серверами і деякими FTP-клієнтами. Отже, вразливостям піддаються всі основні додатки прикладного рівня, включаючи можливість використання мережевих служб, призначених для користувача (віддалений доступ до файлів і баз даних; пересилка електронної пошти та інших додатків), передавання службової інформації, доступ до інформації про збої та помилки, а також можливість формування запитів до рівня представлення.

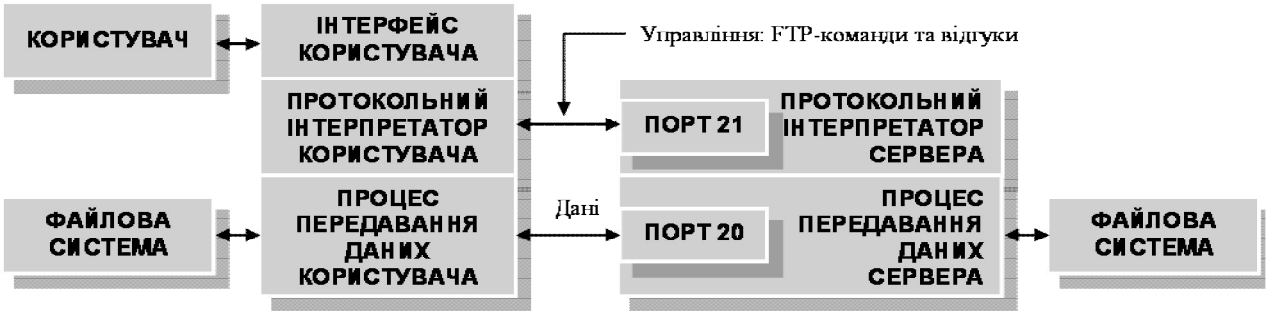


Рис.1. Схема процесів, які забезпечуються протоколом FTP

Як показано в [1], на фоні мережі Інтернет, яка досить швидко розвивається, протокол FTP виглядає не просто старим, а взагалі архаїчним. Ранні чорнові специфікації протоколу датуються 1971 роком, а поточна специфікація існує з 1985 року [2…5]. В ті часи мережею Інтернет користувалися, в основному, університети і дослідницькі центри. Співтовариство користувачів було невеликим, більшість з них знали один одного і всі працювали спільно. Інтернет був доброзичливою мережею, а проблеми безпеки, як такої, не стояло. Ті часи пройшли і багато що змінилося. Технологічний прогрес йшов швидше, ніж хто-небудь міг собі представити, одночасно виросло нове покоління користувачів. На сьогоднішній день Інтернет – повсюдне явище, яким люди спілкуються один з одним безліччю різних способів.

Головна відчутна зміна: Інтернет став ворожим. Доступність та відвертість мережі привернула користувачів-зловмисників, що активно використовують помилки і недосвідченість інших. Як відмічено в [1], побічним ефектом такого розвитку подій стали, наступні явища:

– *Nat-роутери.* Більшість мереж використовує протокол IPv4, який має обмежений адресний простір. Завдяки NAT-роутерам системи з великою кількістю пристроїв можуть користуватися однією і тією ж IP-адресою.

– *Використання персональних файрволов* для захисту користувачів від недоробок в операційних системах та прикладних додатках.

В більшості випадків ці явища конфліктують з роботою FTP-протоколу. Ситуацію погіршують недоробки в самих роутерах і файрволах. Проте, не дивлячись на те, що при правильній настройці FTP-протокол пропонує надійний та випробуваний спосіб передавання файлів, в даний час зловмисники часто використовують його як один з об’єктів для віддалених атак. При цьому протокол використовують як знаряддя, на якому або відбувається збір інформації про систему і мережу в якій знаходиться об’єкт, що атакується, або на вузол через FTP записується шкідливий код з метою його локального виконання.

Виходячи з цього, *метою статті* є огляд та короткий аналіз протоколу FTP; опис його роботи і особливостей, які впливають на його інформаційну стійкість; класифікація віддалених атак; а також аналіз атак, що відносяться безпосередньо до протоколу, і способи захисту від них. На закінчення приводиться опис комплексного підходу до протидії вразливостям протоколу FTP, що є *частиною раніше невирішеної загальної проблеми*.

Викладення основного матеріалу та обґрунтуванням результатів. Як було сказано вище, протокол FTP – це один із старих протоколів Internet, що входить в його стандарти, які відомі як RFC (*Request for Comments* – дослівно: Запит на коментарі). Його основа була створена в 1971 році в стандарті RFC 114, який описував перші механізми передавання файлів між двома вузлами мережі. Після цього він зазнав безліч змін, а в 1985 році був прийнятий стандарт RFC 959, який сьогодні вважається основною версією. Цей стандарт визначає 4 основних функції FTP-протоколу: *надання файлів* для загального доступу; *спрощення обміну* даними та непряме використання видалених комп’ютерів за допомогою програмних засобів; *усунення відмінностей* в представленні даних між вузлами мережі різної архітектури; *надійне* та ефективне передавання даних.

У загальному випадку будь-який мережевий протокол припускає участь у процесі передавання двох вузлів мережі, між якими відбувається обмін інформацією, і наявність одного каналу зв'язку, по якому цей обмін здійснюється. В разі застосовування протоколу FTP використовуються два канали зв'язку: для управління обміном і для передавання даних.

Узагальнена схема роботи протоколу FTP показана на рис.1 [4]. Суть роботи схеми полягає в тому, що спочатку по запиту клієнта формується канал управління, який надалі використовується для передавання команд від клієнта і відгуків від сервера. Канал передавання даних формується сервером по команді клієнта. Цей канал не повинен існувати постійно впродовж всієї FTP-сесії і може формуватися та ліквідовуватися в разі необхідності. Канал управління може бути закритий тільки після завершення інформаційного обміну. Канал передавання даних може працювати в двох режимах: активному і пасивному.

При використанні активного режиму, за допомогою FTP-команди PORT клієнт указує адресу і порт вузла, з яким сервер створює з'єднання з 20-го порту для обміну даними. Пасивний режим включається FTP-командою PASV. Відповідь на цю команду містить адресу і порт сервера, до якого повинен підключитися клієнт для початку обміну даними. Порт вибирається довільно з верхнього діапазону номерів (більше 1024).

Використання роздільних каналів управління і обміну даними дозволяє ввести третього учасника обміну даними. Така схема взаємодії приведена на рис. 2.

Відповідно до рис. 2, користувач організовує канал управління з двома серверами і прямий канал даних між ними. Команди управління йдуть через користувача, а дані – безпосередньо між серверами. Один з серверів працює в пасивному режимі, другий – в активному. Клієнт, перевівши який-небудь сервер в пасивний режим, отримує адресу і порт, який потім використовується для вказівки адреси вузла, з яким повинен з'єднатися той сервер, що працює в активному режимі. З цього випливають вразливості протоколу FTP, тобто ті найбільш істотні недоліки, які вже згадувалися: передавання всієї інформації, а також імен та паролів користувачів у відкритому форматі. Цей факт обумовлює неможливість використання FTP-протоколу для передавання конфіденційної інформації без використання додаткових криптографічних програмних та апаратних засобів. Очевидно, якщо зловмисник має доступ до інформаційного каналу, через який проходять дані, то необхідно використовувати шифрування. Це характерний випадок пасивної дії атаки: стан сервера не змінюється, політика безпеки не порушується, але існує доступ до інформації.

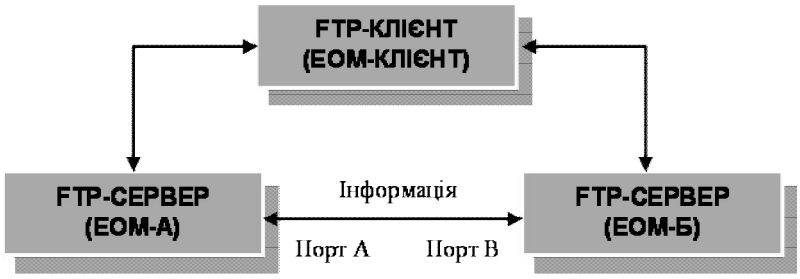


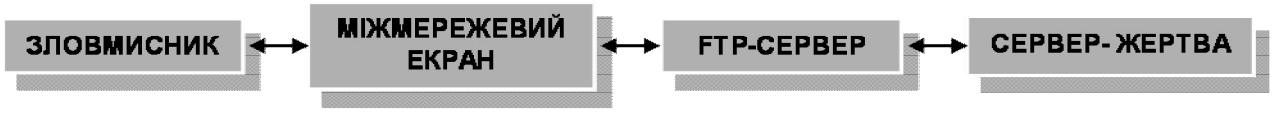
Рис. 2. Процес з'єднання з двома різними серверами та передавання даних між ними

У протоколі не визначені дії, протидіючі підбору паролів. Після неправильного пароля клієнтові надається можливість ввести його повторно, але з'єднання при цьому не розривається. Також не існує обмежень на кількість повторів. В результаті атака, направлена на підбір паролів, може продовжуватися скільки завгодно довго, а відсутність затримок при відповідях сервера підвищує її ефективність. Стандарт протоколу FTP визначає код відповіді 530 на команду USER коли ім'я користувача не є коректним. Якщо ім'я користувача вірне і необхідно ввести пароль, то повертається код відповіді 331. Це є уразливістю з погляду захисту імен користувачів. По відповідях сервера можна судити про існування певного імені в базі користувачів і продовжити підбір пароля.

Наступні дві вразливості пов'язані з пасивним режимом протоколу та можливістю участі у з'єднанні третього вузла. При використанні пасивного режиму передавання даних, при якому сервер вказує клієнту, до якого порту необхідно під'єднатися, на початку передачі, можлива установка з'єднання з іншого комп'ютера. Якщо реальний клієнт вже вибрав

необхідний для скачування файл і має необхідний доступ, то можлива крадіжка від його імені. Зловмисник, знаючи особливості вибору портів FTP-сервером для організації пасивного режиму, підвищує вірогідність успіху атаки. Для цього йому необхідно намагатися постійно встановлювати з'єднання з портами і у разі успіху файл буде вкрадений. Точно так же можна записати файл на сервер від імені зареєстрованого користувача, встановивши з'єднання з портом сервера, який чекає початку файлу.

При атаці «FTP-bounce» використовується команда PORT для з'єднання з третім комп'ютером. Схема атаки зображена на рис. 3. У команді вказується адреса і порт вузла-жертви з яким сервер встановлює з'єднання у відповідь на запит.



Примітка: Пряме з'єднання з сервером-жертвою заборонене правилами Міжнародного союзу електрозв'язку

Рис. 3. Схема атаки «FTP-Bounce»

Перед тим як почати аналіз атаки «FTP-bounce», розглянемо утиліту з відкритим кодом, яка служить для дослідження мережі та перевірки безпеки – NMAP (англ. *Network Mapper* – Мережевий сканер) [7]. Вона була розроблена для швидкого сканування великих мереж, хоча чудово справляється і з одиничними цілями. NMAP використовує IP-пакети такими способами, що з'являється можливість визначення тих хостів, які доступні в мережі. Крім того є можливість точного встановлення інформації про ті служби, які ними пропонуються та які операційні системи і їх версії вони використовують. NMAP також має засоби для визначення типів пакетних фільтрів/брандмауерів, які використовуються в конкретній мережі та ще десятки способів для визначення інших характеристик. В той же час, як NMAP зазвичай використовується для перевірки безпеки, багато мережевих і системних адміністраторів знаходять цю утиліту корисною для повсякденних завдань, таких як контроль структури мережі, управління розкладами запуску служб чи облік часу роботи хоста або служби.

Тепер, маючи загальні відомості про NMAP, розглянемо атаку «FTP-bounce». По своєму внутрішньому змісту «FTP-bounce» не є атакою як такою. Принцип її організації полягає в постійному скануванні портів. Такий спосіб дає можливість обходити фільтри портів, які відфільтровують запити файерволом і, т.ч., сканувати внутрішні ресурси. З цього слідує, що вся справа в команді PORT, яка передається серверу для підготовки з'єднання з клієнтом. Особливість команди полягає в тому, що передається не тільки адреса (номер) порту, але й IP-адреса, що в сукупності з NMAP і надає можливості сканування, тобто NMAP створює FTP-з'єднання і передає PORT команди (хост, що атакується, порт і т.д.).

Приклад: Сканування 666.666.66.66:8080 через хост target.ru [7].

Код:

```
nmap -b anonym@target.ru -p 8080 666.666.66.66
```

З'єднання з FTP-сервером відкрите. Далі NMAP намагається відкрити наступне з'єднання: 666.666.66.66:8080

Код:

```
Server: 220 LAME_HOST FTP server version 4 ready
Client: USER anonym
Server: 331 Guest login OK, send e-mail as password
.......
Server: 230 Login successful
Client: PORT 666,666,66,66,31,144
Server: 200 PORT command successful
Client: LIST
```

Код:

```
Client: PORT 666,666,66,66,31,144
```

Останні дві цифри і є потрібним портом, тобто 31×256+144=8080. При успішно виконаній команді PORT з'являється можливість виконання команди LIST, а це, власне, сигналізує про те, що порт 8080 відкритий. За повідомленням «425 Can't build data connection» або «425 Can't open data connection» є можливість визначити, що сканований порт закритий. Але не варто забувати про те, що FTP-сервери люблять фіксувати з'єднання в журналах і, т.ч., є можливість апостеріорного отримання інформації про несанкціоноване втручання. Як видно, дана атака повністю залежить від того, як реалізований FTP-сервер: присутність передавання PORT тільки до оригінального хосту відсікає цю можливість. Насправді, всі FTP-сервери, розроблені з відповідності вище згаданими RFC, повинні реалізовувати цю можливість. При цьому варто враховувати, що у такому разі немає можливості сканування портів нижче 1024.

Дану уразливість FTP-протоколу можна використовувати для таких цілей [4, 5]:

– прихованого сканування іншого вузла, оскільки сканування проходитиме від імені FTP-сервера, а не того, що атакує. У деяких умовах це дозволяє обійти списки доступів, оскільки FTP-сервер може знаходитися в підмережі жертви або мати з ним довірчі відношення;

– обходу міжмережевих екранів. Якщо якийсь порт внутрішнього сервера закритий міжмережевим екраном, а доступ до FTP-серверу є, то можна від його імені встановити з'єднання з портом, який недоступний. Далі відбувається посилка спеціально сформованого файлу, який містить інструкції для протоколу служби з якою було встановлено з'єднання. В результаті імітується пряме з'єднання з введенням необхідної послідовності команд.

На відміну від атак на FTP-протокол, які використовують його особливості або недоробки, приведені вище, існують також атаки на додатки – сервери, що реалізовують цей протокол. В основному для цього використовуються некоректно сформовані команди. У загальному вигляді метою даних атак є: *порушення працездатності* системи, яке включає аварійний останов або зациклення програми-сервера, а також надмірне використання системних ресурсів (наприклад, оперативної пам'яті або процесорного часу); *підвищення привілеїв* доступу до системи; *вихід за межу* FTP-каталога; *виконання команд* операційної системи з привілеями FTP-сервера.

Розглянемо протидію вразливостям FTP-протоколу. Його вразливості здебільшого обумовлені його особливостями і відсутністю способів захисту передаваної інформації. Для підвищення безпеки необхідно використовувати сторонні засоби, а також ретельно продумувати взаємодію вузлів мережі, які функціонують на основі протоколу FTP [4, 5].

Проблема передавання всієї інформації у відкритому вигляді вирішується або з використанням засобів шифрування, де це можливо, або захистом каналів зв'язку від несанкціонованого доступу. Решту проблем можна вирішити за допомогою фільтрації.

Для захисту імен користувачів від підбору, фільтр повинен підміняти відповідь 530 FTP-сервера, який формується при отриманні команди USER з неіснуючим ім'ям, на відповідь 331, який підтверджує існування імені користувача.

Для захисту паролів від перебору необхідно налаштувати FTP-сервер так, щоб з'єднання закривалися після деякої кількості невдалих спроб введення паролю. У фільтр можна також заносити кількість вказаних команд USER-PASS з негативною відповіддю сервера і при їх неприпустимому значенні, закривати порт для даної адреси на певний проміжок часу. Для цього необхідна взаємодія з міжмережевим екраном. Також необхідно забезпечити паузу перед відповіддю на кожен неправильний пароль, що дозволить істотно загальмувати їх перебір. Проте при цьому залишається можливість перебору з використанням паралельних з'єднань з FTP-сервером. Для протидії необхідно встановлювати обмеження на кількість одночасно обслуговуваних підключень, що, у свою чергу, робить можливою атаку типу «Відмова в обслуговуванні».

Для запобігання крадіжці файлів при пасивному режимі необхідна фільтрація по IP-адресі. Адреса клієнта, який ініціював пасивний режим, буде адресою призначення пакету з відповіддю на команду PASV. При використанні такої фільтрації стає неможливим обмін

між двома серверами, що ініціюється клієнтом, оскільки сервер, переведений в активний режим, матиме адресу, відмінну від адреси клієнта, а пакети від нього фільтруватимуться. З цієї причини необхідно визначити, чи існуватиме такий обмін і чи можна його уникнути, і лише після цього здійснювати фільтрацію.

Можливість атаки «FTP-bounce» також обумовлюється особливостями FTP-протоколу. Спосіб рішення – такий же, як і для проблеми крадіжки файлів. Також можлива фільтрація пакетів або така конфігурація FTP-сервера, щоб при отриманні команди PORT з номером порту менше 1024, видавалася помилка. Це потрібно для того, щоб запобігти атаці на інші стандартні сервіси від імені FTP-сервера. Протокол FTP не повинен використовувати номера портів менше 1024 для передавання даних.

Найбільш значна група вразливостей пов'язана з атаками на додатки. Це пов'язано з великою різноманітністю програмних засобів, що реалізовують FTP-сервери, і їх версій. Існує 2 способи обробки некоректних запитів.

Спосіб 1:

1) Вести базу некоректних запитів і фільтрувати пакети відповідно до неї. Цей варіант поганий тим, що база вразливостей росте, а швидкодія фільтру падає. Також необхідно відстежувати нові вразливості та нові версії програмного забезпечення для оновлення цієї бази.

2) Не фільтрувати некоректні запити взагалі, а відстежувати стан FTP-сервера. В цьому випадку повинна бути забезпечена відмовостійкість операційної системи до збоїв FTP-сервера і можливість коректного перезапуску додатків в ній.

Спосіб 2: Комплексний: включає, крім фільтрації, наступне:

1) Настройку FTP-сервера для роботи під окремим обліковим записом з обмеженими привілеями. В цьому випадку отримання прав самого сервера в результаті використання його уразливості нічого не дасть;

2) Відстежування стану FTP-сервера, а у разі аномалій в його поведінці – коректний перезапуск. Це повинно здійснюватися засобами операційної системи (контроль за виконанням процесу і використовуваних ресурсах) та сторонніми засобами (контроль за можливістю з'єднання з FTP-сервером по мережі);

3) Заборона на виконання програм, записаних клієнтами на FTP-сервер. Це необхідно в разі частого використання FTP-протоколу як засобу для запису зловмисником шкідливого коду на сервер-жертву, який згодом запускається як локальний.

Висновки. На підставі матеріалів, викладених в [1…7], розглянуті аспекти безпеки FTP-протоколу, який є одним з головних засобів передавання файлів по мережі. Встановлено, що FTP-протокол не є безпечним і, отже, необхідне використання комплексних засобів захисту. Цими засобами є:

- правильна побудова моделі файлового обміну;
- коректне налаштування програмного забезпечення;
- використання алгоритмів шифрування скрізь, де це можливо;
- застосування фільтрації при взаємодії з міжмережевими екранами;
- використання сторонніх засобів для контролю за станом програмного забезпечення FTP-сервера.

Застосування цих засобів дозволить зменшити ризик атак через FTP-протокол.

Останнім часом до FTP-протоколу розроблено ряд доповнень, які можуть істотно підвищити його криптостійкість. Так, з виходом доповнення IIS7.5 з'явилася можливість повноцінно використовувати сервіси FTP версії v7.5, а саме – повноцінно працювати по протоколу IPv6 і бути упевненим в захисті передаваних даних по протоколу FTPS на FTP-сайт. Крім того, з'явилися нові розширені методи аутентифікації з великими можливостями гнучкого налаштування за рахунок інтеграції та уніфікованого інтерфейсу адміністратора. Також з'явилася можливість скорочення часу обслуговування і, як результат, квотування за

часом, що допомагає раціонально використовувати ресурси сервера, а розширена настройка брандмауера значно полегшує налаштування FTP-сервісів в пасивному режимі. Ці корисні нововведення роблять сервіс FTP у складі сервера IIS7.5 достойним вибором при вирішенні питань передавання, зберігання та забезпечення безперервного захисту інформації при передаванні даних. Проте, слід зазначити, що відомостей про їх реальну криптостійкість у доступній літературі знайдено не було, що є перспективами ***подальших досліджень.***

Додаток. Приклад сканування за допомогою утиліти NMAP [7].

```
# nmap -A -T4 scanme.nmap.org playground

Starting Nmap ( http://nmap.org )
Interesting ports on scanme.nmap.org (64.13.134.52):
(The 1663 ports scanned but not shown below are in state: filtered)
PORT    STATE  SERVICE VERSION
22/tcp  open   ssh     OpenSSH 3.9p1 (protocol 1.99)
53/tcp  open   domain
70/tcp  closed gopher
80/tcp  open   http    Apache httpd 2.0.52 ((Fedora))
113/tcp closed auth
Device type: general purpose
Running: Linux 2.4.X|2.5.X|2.6.X
OS details: Linux 2.4.7 - 2.6.11, Linux 2.6.0 - 2.6.11

Interesting ports on playground.nmap.org (192.168.0.40):
(The 1659 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn
389/tcp  open  ldap?
445/tcp  open  microsoft-ds  Microsoft Windows XP microsoft-ds
1002/tcp open  windows-icfw?
1025/tcp open  msrpc         Microsoft Windows RPC
1720/tcp open  H.323/Q.931   CompTek AquaGateKeeper
5800/tcp open  vnc-http      RealVNC 4.0 (Resolution 400x250; VNC port: 5900)
5900/tcp open  vnc           VNC (protocol 3.8)
MAC Address: 00:A0:CC:63:85:4B (Lite-on Communications)
Device type: general purpose
Running: Microsoft Windows NT/2K/XP
OS details: Microsoft Windows XP Pro RC1+ through final release
Service Info: OSs: Windows, Windows XP

Nmap finished: 2 IP addresses (2 hosts up) scanned in 88.392 seconds
```

Література

1. Описание особенностей FTP протокола [Електронний ресурс] // – Режим доступу : http://tagievara.narod. ru/pages/articles/ftp_osn_prot.html .

2. FTP. Материал из Википедии – свободной энциклопедии [Електронний ресурс] // – Режим доступу : http://ru.wikipedia.org/wiki/FTP .

3. Протокол FTP [Електронний ресурс] // – Режим доступу : http://www.cyberguru.ru/networks/protocols/ftp. html .

4. FTP [Електронний ресурс] // – Режим доступу : http://elibrary.ru/item.asp?id=11654912&.

5. Иванов А. В. Исследование безопасности протокола FTP / А. В. Иванов // Научно-технический вестник СПбГУ ИТМО. – Выпуск 19. Программирование, управление и информационные технологии. – СПб: СПбГУ ИТМО, 2005. – С. 167-172.

6. Модель OSI [Електронний ресурс] // – Режим доступу : http://citforum.ru/nets/switche/osi.shtml .

7. Справочное руководство NMAP [Електронний ресурс] // – Режим доступу : http://nmap.org/man/ru/index. html#man-ex-repscan .