СОФИЙСКИ УНИВЕРСИТЕТ
“ СВ. Климент Охридски”
гр. София

Факултет Матиматика и Информатика

КУРСОВА РАБОТА

Тема: ***Сигурен интернет доставчик***

Изготвили: Росен Енчев Чолаков, fN 42882,
Радостина Николова Маркова, fN 42924,
Величко Гинев Сарев, fN 43293

Специалност: Информатика 4 курс

гр. София
2004 год.

URL

Унифициран локатор на ресурси (URL):

В настоящия момент съществуват множество протоколи и системи за търсене и извличане на документи по Internet. Тези системи имат за цел да осигурят глобално търсене на документи при различни компютърни платформи и произволна дълбочина на вмъкване в компютърните мрежи.

Универсалният идентификатор на ресурси (Universal Resource Identifier, URI) е елемент от това универсално множество от имена. Унифицираният локатор на ресурси (Uniform Resource Locator, URL), е форма на URI, изразяваща адрес, който съответствува на алгоритъм за достъп при използуване на мрежовите протоколи. .

Съществуват два типа URL. Първият представлява абсолютен URL, който съдържа пълния адрес. Не е нужно нищо друго, за да се намери желаната информация. Вторият тип представлява относителен URL. Относителният *URL* съдържа само необходимото допълнение към базовия адрес за намиране на желаната информация от текущо установената позиция (директория, виртуална машина и др.).

HTTP

Протоколът HTTP:

Една от целите на проекта WWW е лесното получаване на информация, независимо от това къде се намира по Internet мрежата. Като стандартен формат за представяне на WWW документи се използва хипертекст. Протоколът HTTP (Hypertext Transport Protocol) е прост комуникационен протокол за извличане на документи на базата на TCP/IP протоколи от Web сървъри по Internet с цел тяхното изобразяване от Web клиенти. Той се базира на факта, че документите, които се извличат, съдържат информация за бъдещите връзки, които потребителят може евентуално да заяви (за разлика от FTP или Gopher, при които информацията за възможните бъдещи връзки трябва да бъде предадена чрез протокола).

Схема на взаимодействие по модела на ISO:

Протоколът за пренасяне на хипертекст HTTP притежава необходимата простота и скорост за разпределена корпоративна хипермедийна информационна система (Intranet). В такава система е необходимо бързо проследяване на връзки между информационни единици, които се намират върху отдалечени станции.

Протоколът HTTP е обектно-ориентиран и може да се използува в разпределени обектно-ориентирани системи. Важно негово свойство е универсалност на представяне на данните, което позволява системите да са независими от нови, по-усъвършенствани методи за представяне на данни. Протоколът HTTP позволява използването на отворено множество от методи. Той се базира на достъпа, осъществяван от URI (Universal Resource Identifier) към който се включват име (URN Uniform Resource Names) (виж [RFC 1737]) или адрес (URL Uniform Resource Locators) (виж [RFC 1738]).

Компонентите на протокола HTTP са:

- изпращане на заявка до сървъра;
- метод, приложен върху обекта, специфициран чрез URL;
- заглавна информация (list of headers) в съобщението-заявка;

• получаване на отговор;
• кодове на състоянието;
• заглавна информация (list of headers) за всеки предаван обект;
• съдържание на всеки предаван обект;
• формати.

За работа с WWW не са необходими знания върху HTTP. Протоколът се обслужва от съответните приложения, както от страна на сървърите, така и от страна на клиентите. Трябва да се отбележи, че този протокол обуславя подръжката на информационни услуги в Internet и не може да се разглежда като конкурент на останалите видове основни протоколи и свързаните с тях услуги. Независимостта на HTTP протокола го превръща в платформа при комуникацията между потребители/услуги и proxies/gateways и други Internet протоколи като SMTP, NNTP, FTP, Gopher, WAIS с което осигурява хипермедиен достъп до ресурси на различни приложения и с това улеснява интегрирането на услугите. Един от основните му недостатаци е, че при преноса на данните те не са криптирани. Затова в големи организации, в които е нужно някакъв вид сигурност на данните се използва HTTPS.

В общи линии, HTTP не е несигурен. Обаче, има някои неща, които трябва да имаме в предвид. Най-големият проблем с HTTP не е HTTP, а системния администратор, осигуряващ обслужването. Не трябва да изпълняваме `httpd` като root. Дори малката слабост в CGI програма може да означава компромис на системата, ако вие изпълнявате httpd като `root`. Това означава, че remote потребителите могат, да изпълняват процеси като root.

Също така, да разгледаме възможността httpd, да се изпълни като `chrooted` процес. Първо, това предоставя само много минимална полза за защита и сериозно ограничава способността да използува се CGI. Например, при нормални обстоятелства, потребители могат да изпълнят CGI програми от подструктурата на своята собствена директория . Ако httpd е в `chrooted` enviroment, потребителите няма да бъдат способни, да изпълняват тези скриптове, освен ако те не са под `chrooted enviroment`. Освен това, защитната полза е лъжлива в най-добрият случай. Това прави достъпа твърде ограничен за ефективна работа на CGI.

Една ценна програма, която може да ни помогне при тестването на CGI приложения, е CGIWRAP.

HTTP сигурността е претърпявала много промени, особено в миналите двете години - например развитието на сигурни httpd сървъри. Secure Hypertext Transfer Protocol (S-HTTP) включва RSA и Kerberos-основано криптиране и установяване на истината.

Domain Name System

Domain Name System (DNS):

Ранните Internet конфигурации изисквали потребителите да използват само IP адреси в цифров вид. Това се променя бързо и започват да се използват символни имена на хостове. Например, вместо да се пише TELNET 128.12.7.14, може да се напише TELNET eduvm9, и това се превежда до IP адрес 128.12.7.14. Това представя проблема за организирането на на връзки между IP адресите и имената във високотехнологичните машини по координиран и централизиран начин.

Първоначално организирането на тези връзки се осъществява от Мрежов Информационен Център (Network Information Center - NIC) в един файл наречен HOSTS.TXT, до който се осъществява връзка от хостовете по FTP. Този метод на организация се нарича flat namespace.

Поради много бързото нарастване на броя на хостовете, този механизъм става прекалено тромав и се заменя от нова концепция: DNS. Хостовете може да продължат да ползват локалния flat namespace (HOSTS.LOCAL файла) вместо или в добавка към DNS, но извън малки мрежи DNS-ът е много съществен. DNS позволява на програма, работеща на хост, да извърши свързването на символно име с IP адрес за всеки друг хост, без нуждата друг хост да има цялостна база данни от имена на хостове.

DNS spoofing атаки

Съществуват DNS spoofing атаки. Те се могат да се появят, когато DNS машината бъде изложена на риск от кракер (cracker). Вероятността за това събитие е незначителна, но ако се случи, това води до опасно излагане на системата. Рядкостта на тези атаки не би трябвало да е успокоение. Възможно е атакуващият да опитат spoof bind и да променят данните. Някои системи и програми зависят от тази информация за автентициране, така атакуващият ще пробие тези системи и ще получи неправомерен достъп.

Във всеки случай, в DNS spoofing кракерът явно изменя hostname-IP адресни таблици. Тези промени се записват в таблични на DNS сървъра. По такъв начин, когато има клиентски заявки за търсене, сървърът дава лъжлив адрес. Този адрес ще бъде IP адрес на машина напълно под управлението на кракера.

DNS spoofing се открива лесно. Ако има съмнения за spoof на някой DNS сървър, проверяваме другите DNS сървъри на мрежата. Сървърът е бил spoof-нат за известно време, доказателството незабавно ще изплува. Другите сървъри ще съобщят резултати, които се различават от тези на кракнатия сървър. Допитването до другите сървъри може да не даде разултат, ако DNS сървърът, който проверяваме е с променени таблици от по-дълъг период от време. Тогава той е препратил лъжлива информация и на другите сървъри. Ако има нередности в разрешаването на имената (naming resolution), може да се прибегне до използване на DOC (domain obscenity control) - програма, която поставя диагноза на сървъри със съмнително поведение, като изпраща заявки до подбрани сървъри и извършва анализ на получените отговори.

E-mail

Електронна поща (e-mail):

Това е Internet услугата, без която не може. Това е личният достъп до мрежата на всеки потребител в Internet.

Защитата на кореспонденцията се осигурява на няколко нива. Първо, достъпът до всяка пощенска кутия е санкциониран чрез потребителското име на собственика й и парола за достъп известна само на него. Второ, самите съобщения могат да бъдат

кодирани посредством ключ, известен само на двамата кореспонденти със средствата на специализиран софтуер.

Функциониране на електронната поща:

Схемата на функциониране на електронната поща предполага наличието на съответно приложение за оформяне на съдържанието на едно съобщение и мрежов протокол за неговото транспортиране по глобалната мрежа.

За мрежа, чийто базов протокол е TCP/IP, независимо от това дали става дума за Internet или Intranet схемата е една и съща. За изпращане на едно съобщение от локална машина се използва протоколът Simple Mail Transfer Protocol (SMTP) с помощта на инсталираното Mail приложение (Mail клиент). Пътят на съобщението преминава през SMTP сървър и оттам по същия протокол в глобалната мрежа. Пристигащите съобщения по мрежа се приемат от POP сървър (Post Office Protocol), където е разкрита пощенска кутия за съответния кореспондент. Достъп до пристигащите съобщения осъществява съответното Mail приложение (Mail клиент) от локалната машина на кореспондента чрез извличането им от POP сървъра.

Електронни адреси:

Електронният адрес на всеки кореспондент в Internet мрежата е част от едно подмножество на Internet адресите. Всяка машина с достъп в мрежата има свой Internet адрес, който се състои от четири десетични цифри разделени с точки (194.141.3.180). Поради явните неудобства при боравенето с цифри реалните адреси имат свой символичен еквивалент от имена на машина.подмрежа.област, определен като FQDN (Fully Qualified Domain Name). Съответствието се поддържа от специализиран сървър във всяка област или подобласт наречен DNS (Domain Name Service). Електронният адрес на една пощенска кутия (Mail Box) се формира на базата на Internet адреса на машината с добавена информация за личностната идентификация на кореспондента.

Протоколът SMTP е доста примитивен. Значителен недостатък е фактът, че чрез него се обменят данни в ASCII 7 бита, което означава, че едно съобщението може да съдържа само символи от англо-саксонската азбука. Това ограничение се преодолява от разширеният вариант на протокола SMTP (Extented SMTP, описан в документа RFC 1425), където се задава схема на обмен на съобщения в 8 бита. Допълнението позволява обмен на символи и от "горната" част на ASCII таблицата (букви с ударения, кирилица), но в зависимост от възможностите на кореспондентските MTA (Message Transfert Agent) агенти, т.е в зависимост от функционалните възможности на съответната Mail-сървър-клиент система.

Независимо от подобренията SMTP теоретично е ограничен в обмена само на текст и не предвижда средства за описание на конвертиращи методи. Нуждите на потребителите са наложили разработването на допълнителни програми, които преобразуват произволни файлови формати в 7 или 8 бита (uuencode, binhex). Така първоначално в Unix системите се появяват програмите uuencode и uudecode за конвертиране на файлове от и към 7 битов код. Версии на тези програми са достъпни и за MSDOS и Windows. Полученият преобразуван текст се транспортира като SMTP съобщение и подлежи на повторно преобразуване при получателя. Тези неудобства са наложили разработването на MIME(Multipurpose Internet Mail Extensions) като формат на мултимедийно съобщение съвместимо с SMTP, с което се преодоляват два основни проблема - указва се методът на кодиране, структурата и типа на данните в съдържанието на съобщението. За различните платформи вече има значителен брой Mail-клиент приложения, които поддържат MIME (Elm - Unix, Eudora - Windows и MacOS, Netscape - Un ix, Windows и MacOS). Удобството тук е, че конвертирането на

съдържанието на съобщението се извършва автоматично както при подателя, така и при получателя. С това обменът на документни файлове от типа Word например, престава да бъде проблем. Сходни възможности са реализирани и за метода BinHex.

Използване на електронната поща:

Изборът на програмен продукт за електронна поща зависи от конкретната платформа. За системи, работещи под Linux изборът е вече направен, защото Mail приложението е част от операционната система. Съществува голямо разнообразие от програми за Windows и MacOS. Сред най-популярните и достъпни са Eudora и Mail приложението на Netscape. И двата продукта имат версии за IBM PC и Macintosh.

Усвояване на работата с електронната поща предполага изучаване на основните й функции в конкретна програмна среда. За различните платформи приложенията са различни, но изпълняват еднакви функции. Една програма за обработка на електронна поща обикновено позволява изпълнението на следните действия:

• санкциониран достъп и четене на пристигащите съобщения;

• съхранение ранжиране на пристигащите съобщения по тематика, имена, адреси и др.;

• отговор и препращане на постъпило съобщение;

• редактиране и изпращане на собствени съобщения;

• присъединяване на външни обекти към изпращаните съобщения.

Електронна поща в Unix:

За среда Unix или Linux съществува, вградено в операционната система приложение за обработка на електронна поща. Mail програмата е итерактивна и се управлява от командния ред посредством команди за получаване, изпращане, четене, запис, печат, изтриване и препращане на съобщения.

За получаване и изпращане на съобщения в Unix система е необходимо предварително потребителят да има собствена регистрация (account) , защото идентификацията на всеки се извършва с потребителско име и парола за достъп при първоначално влизане в системата.

FTP

Обмен на файлове по Internet - File Transfer Protocol (FTP):

FTP е Internet услуга за прехвърляне на файлове в ASCII или BINARY формат между два компютъра, свързани в мрежата. Услугата се обслужва от свой протокол като надстройка на TCP/IP протоколите. FTP е много полезно средство особено за пренасяне на файлове или програмни китове с голям обем.

Осъществяване на FTP връзка между два компютъра в Internet мрежата и провеждане на сесия изисква изпълнението на определена последователност от действия от страна на клиента (FTP Client) в процеса на установяване на връзка и извличане на информация от файловия сървър (FTP Server) на кореспондентския компютър. Тези действия предполагат познаването на едно не голямо множество от команди за този протокол. Познаването на основните команди е препоръчително дори и в случаите, в които се използват програмни надстройки от меню или диалогов тип. Изпълняваните действия по време на една FTP сесия е добре да се познават, с което се

гарантира правилното и преди всичко колегиалното използване на възможностите на тази услуга.

FTP услугата се основава на взаимодействието по Internet на един FTP клиент, който влиза в контакт по FTP протокол с FTP сървър. Клиенски приложения се предлагат за всички компютърни платформи във вид на freeware, shareware или comercialeware. Приложенията реализиращи FTP Server са относително по-малко на брой, покриват всички компютърни платформи и се предлагат във вид на freeware, shareware или comercialeware.

Докато услугата WWW поддържа четири нива на автентикация (anonymous, basic, digest, and integrated), плюс опция за SSL криптирано предаване, FTP поддържа само anonymous достъп и basic методи за автентикация. FTP е по-несигурен от HTTP, защото FTP предлага само anonymous достъп и basic автентикация. Например, ако работите в един вътрешен FTP сайт в рамките на една организация и използвате basic автентикация всеки потребител от мрежата с използването на снифер потенциално може да получи връзка към една FTP сесия и да разкрие потребителската парола.

Формат на файловете:

Файловете в FTP сървърите по принцип са общодостъпни и съставляват част от т.н. Public Domain. Това означава, че те могат свободно да бъдат копирани. Файловете са записани в различни формати, но все пак на базата на три основни правила:

• архивиране - множество файлове, съставляващи обикновено един цялостен продукт или тема, се записват в един архив;

• компресия - с цел намаляване обема на архивите и намаляване времето за прехвърляне на файловете по мрежата;

• кодиране - осигуряване на възможност за прехвърляне на файлове по електронна поща (e-mail) - Mail или News протоколи;

Комбинирането на тези правила или техники е възможно и зависи от произхода и предназначението на файла. Така архивите могат да бъдат с компресия, като отделно при необходимост да се и кодират;

При FTP услугата са възможни два вида трансфер на файлове:

• ASCII - за файлове, чието съдържание е обикновен текст, включващи и кодовете за нов ред (CR, LF) и нова страница (FF);

• BINARY (IMAGE) - за файлове, записани в двоичен формат. Това са изпълнимите файлове, архивните и компресираните.

Разширенията на файловете могат да се приемат като идентификатори на техния формат.

FTP сесия:

Потребителско име е необходимо, за да се открие процес за този потребител в машината. Официално регистрираните потребители използват своите потребителски имена за достъп до тази услуга в тази машина. За останалите е предвидено едно обобщено име, което е anonymous.

Анонимен или неизвестен съвсем не означава, че потребителя, търсещ достъп до файловата система на сървъра, остава такъв за него. Следващият въпрос е паролата за достъп (Password). Това предопределя същинските права на потребителя. Регистриран потребител добива предписаните му права, а външен потребител (guest) добива достъп до общите части на файловия сървър. В един FTP сървър като общи и достъпни за публиката се заделят определени части от файловото му пространство.

Един анонимен потребител може да се регистрира като гост в отговор на въпроса за парола на достъп (Password). Това отговаря на общите правила, дефинирани

в самия FTP протокол. Голяма част от публичните или обществени FTP сървъри обаче изискват от потребителя идентификация с неговият e-mail адрес. Общите правила за работа в Internet и колегиалното отношение в тази мрежова среда предполагат точната регистрация на всеки потребител. Така вместо guest за парола на достъп е препоръчително да се въвежда електронният адрес на потребителя.

Всяка изпълнена от страна на потребителя операция се регистрира в FTP сървъра и представлява своеобразен протокол за неговите действия. Така всеки опит за неправомерни действия се регистрира веднага.

Преобладаващата част от FTP сървърите са базирани на Unix машини и файловата им структура обикновено е сходна. Различията се отнасят предимно до предмета и обема на съхранявана информация.

Важно е да се провери дали дължината на файла в байтове в сървъра отговаря на броя прехвърлени байтове в машината на потребителя. Данните за средната скорост на трансфер са информация за моментната пропускателна възможност на мрежата. При по-ниски скорости и най-вече под 1 Kb/s прехвърлянето на големи файлове става почти невъзможно, а и не е желателно в такъв момент допълнително да се натоварва мрежата. Като общо правило, трябва да се избягва FTP услугата от отдалечени сървъри в работните часове на деня.

FTP клиенти:

Операционни системи като Windows и Linux съдържат стандартно FTP клиент. Те поддържат по-голямо или по-малко множество от ftp-команди. Използването им става от команден ред със задаване на последователност от управляващи атрибути. Диалогът между клиента и сървъра се извършва конзола. Фирми, производителки на софтуерни продукти, предоставят freeware или shareware програми за управление на FTP-клиенти. Разработени в графична среда, те предоставят един удобен за използване потребителски интерфейс общ за операционната среда. Интергрираната среда на един FTP-клиент включва и някои допълнителни фунции за улесняване на работата. Потребителят има възможност:

• последователно да изгражда тематични каталози на посещаваните от него FTP-сървъри;

• да стартира връзка към повече от един FTP-сървър;

• бързо да възстановява прекъсната връзка;

• да възстановява извличането на файл от сървър след прекъсване - resuming, reget, а това зависи от възможностите на сървъра;

• да получава диагностична информация за обема на тегления файл, приблизително време за изпълнение на операцията, моментна скорост на обмен по мрежата.

Анонимен FTP (anonymous FTP):

Има няколко причини, които определят дали, да се позволява напълно неограничен и анонимен FTP. Обичайно, това се прави, когато се предлага софтуер дистрибуция свободно или, когато се поддържа архив от информация, към която има многостранен интерес. В който и да е случай, се разпределя машина изрично с такава цел, която не изпълнява други услуги и държи само информация, която вече е била записана.

Някои протоколи трудно извършват сигурно филтриране (например RPC и UDP услуги) и по този начин се осигурява повече отваряния на вътрешната мрежа. Услугите поддържани от тази машина могат да взаимодействуват по катастрофален начин. Например, позволявайки анонимен FTP на същата машина, където има WWW сървър,

може да позволи да се постави файл в анонимната FTP област и в последствие HTTP сървър да го изпълни .

Напълно анонимният FTP с writable директория го прави лесна мишена, за практукуващите FTP bounce атака.Накратко, FTP bounce атаката включва използване един FTP сървър за прикритие, за да се получи достъп до друг FTP сървър, който е отказал връзка. Типичната ситуация е, когато целевата машина е конфигурирана, да откаже връзки от определена IP адресна йерархия. Така атакуващия използва другата машина посредник (intermediary) за достъп до целта. Bounce атаката завършва с писане в директорията на FTP посредника файл, който съдържа команда, да се свърже с целта, и да извлече някой файл там. Когато посредникът се свързва с целта, заявката идва от неговия собствен адрес, а не от адреса на атакуващия. Целта уважава заявката за връзката и препраща изискания файл.

FTP bounce атаки не са били проблем с висок приоритет за защитени кръгове, главно тъй като те са редки и обикновено не включват опити за crack-ване.

TELNET

Отдалечен терминал (TELNET):

TELNET (Networking Terminal Protocol) е Internet протокол (услуга), надстройка на TCP/IP, за осъществяване на връзка (диалог) с друга машина в режим на отдалечен терминал. Преимуществата на тази услуга са в това, че потребител от своя компютър може, да използва ресурсите на друга машина, отдалечена от него на значително разстояние. Дистанционното включване може да осигури достъп до собствена пощенска кутия на друга машина в друг град, в друга държава, достъп до бази данни с интересна за клиента информация, консултиране на библиотечни каталози за книги и статии, получаване на валутна, борсова и друга бизнес информация. Това е едно подходящо средство за използване на изчислителните възможности на по-мощни компютърни системи, отдалечени на хиляди километри от собствената машина. В случаите на връзка по модемна линия услугата се оказва особено ценна, поради минималния обмен на информация с “отдалечения терминал”. Ако е налице модем, чиято скорост на обмен е по-малка от 9600 bps или качествата на телефонната линия не позволяват обмен при по-висока скорост, то за предпочитане е използването на Telnet услуга за свързване с Internet сървър вместо ползване на пълно Internet ядро по SLIP или PPP.

Работата с Telnet услугата предполага наличието на програма, която поддържа Telnet клиент за осъществяване на връзка с Telnet сървър (Telnet deamon). Управлението на услугата може да се осъществи от команден ред или чрез избор на операция от меню. И в двата случая става дума за изпълнение на определен набор от команди, с които се реализира непосредственото й управление. Достъпният набор от команди зависи от конкретната програмна реализация и операционната система. В общия случай програмата Telnet-клиент използва два параметъра за осъществяване на връзка с друга машина:

Telnet <address> <port>

• address е IP адрес или име на машина, псевдоним;

• port е номер на порт за обслужване на услугата, ако не е указано се приема 23.

Номерът на порта не е директно свързан с хардуерната част на компютъра, а указва на отдалечената машина коя услуга ще бъде използвана.

При различните платформи и операционни системи Telnet услугата работи по еднообразен начин, но има различия в набора от команди поддържани от конкретната програмна реализация. Telnet приложения има реализирани за UNIX, LINUX, MS-DOS, WINDOWS, MacOS.

Telnet сесия:

Една Telnet сесия е времето на взаимодействие - диалог с отдалечената машина. Провеждането на успешна Telnet сесия се предхожда от определена последователност от стъпки. Open с IP адрес или име на Telnet сървър отваря връзка на подразбиращия се порт. При нормална връзка първо протоколът прави опит да открие път към сървъра. Стартирането на процес клиент в отсрещната машина изисква идентификацията на потребителя с име и парола за достъп.

За някои обществено достъпни сървъри паролата за достъп може и да не е необходима, а потребителското име, ако се изисква, е точно определено и предварително известно.

С появата на символа за команден ред на отсрещната машина завършва установяването на контакт с нея. От тук нататък потребителят разполага с възможностите, които отсрещната система му предлага.

При работата в режим на отдалечен терминал трябва да бъдат установени параметрите на терминалната емулация в локалната машина. В редки случаи, след установяване на връзка, на екрана на клиентската машина се появяват странни символи. Това означава, че не е била подбрана подходящата терминална емулация. При подобни случаи, връзката трябва да се прекрати и да бъде възобновена след установяване на коректни параметри за отдалечения терминал.

За Unix операционна система Telnet услугата е част от системните програми и допълнителна инсталация не се налага. За Windows реализираните Telnet приложения са значителен брой. Различията между тях не са съществени, понеже реализират една услуга. Изборът на потребителя е насочен повече към удобствата, които му предлага съответната програма и начина на осъществяване на връзка - чрез Ethernet мрежа или по модем. Различията отразяват още множеството от емулирани терминали и удобствата при използване на приложението. Отделните програмни реализации могат да бъдат класифицирани по различен признак, като един от тях е начинът на разпространението им. Една част за freeware, т.е. за свободен достъп, друга shareware, т.е. с поделен достъп, третата група принадлежи на комерсиалните продукти.

Firewalls

Една от основните идеи, която се крие зад защитни стени (firewalls) е, че мрежата остава теоретично невидима (или недостъпна) за неоторизиран достъп. Този процес работи в резултат на схемата на изключване, която потребителят може да приложи използвайки защитни стени.

Теоретично защитните стени са най-силно ограничителната мярка, която може да бъде приложена за защита (с изключение разбира се на изключването на вашата система от Internet). Независимо от това спорните въпроси касаещи тази силно ограничителна среда остават. Един от тези въпроси е, че гарантирането на сигурността със защитна стена може, да бъде така строго конфигурирано, че на практика сериозно, да ограничи работата на мрежата. Например някои изследвания предполагат, че

използването на firewall е непрактично в среди, където потребителите зависят в критична степен от разпределени приложения. Тъй като защитните стени прилагат толкова стриктна политика за гарантиране на сигурността, тези работни среди се оказват в задънена улица. Това, което печелят от гледна точка на сигурността, го губят като функционалност. Университетите са перфектен пример на този тип работна среда. Изследователските звена в университетите много често се управляват като два или повече департамента (често разположени в мрежови сегменти отдалечени един от друг) са включени за събиране на данни (и потвърждение на резултатите от изследването). В такава среда е много трудно да се работи при толкова строги ограничения за сигурност.

Другият проблем при разглеждането на защитни стени е това, че те водят към струпване на много неща на едно място. Тъй като защитната стена е представя мрежата пред света едно нарушение може да предизвика срив на цялата вътрешна мрежа. Това означава, че защитните стени създават една обстановка, в която те представляват единствения контрол над реалния достъп до мрежата. Защитните стени почти винаги се описват като тясното гърло на мрежата, където се извършват всички автентикации. Това изглежда удобно дотогава докато защитните стени са непробиваеми. В случай, че firewalls бъдат разбити, мрежите, които разчитат на защитни стени, ще бъдат напълно прозрачни и шансовете за оцеляване ще бъдат слаби.

Преди да се конструира една защитна стена необходимо е да се предприемат сериозни изследвания. Когато конструирате една защитна стена, трябва да се познава добре мрежата. Различните мрежови сегменти (в една и съща или в различни мрежи) се нуждаят да комуникират помежду си. Тези мрежи могат да комуникират посредством автоматизиран процес или чрез операторски намеси. Автоматичните процеси на комуникация би могло да бъдат лесно адаптируеми. Операторските намеси обаче могат съществено да се различават.

За някои организации защитните стени са просто непрактични. Доставчиците на интернет услуги (ISPs) са в този клас. Един такъв доставчик може много бързо да загуби всичките си клиенти, установявайки такава строга политика за сигурност. Наистина някои твърдят, че защитните стени не са необходими. Тези хора смятат, че наличието на солидна практика на системните администратори, ще предостави същите предимства както защитната стена без намаляване бързината на мрежата и без затрудняване процеса на свързване. Освен това има и други проблеми свързани с изграждането на една защитна стена. Ако FTP, Telnet, Gopher, HTTP, RPC, rlogin и NFS са единствените протоколи, които интернет ще използва, защитната стена ще създаде само определен брой проблеми при достъпа. Все пак proxies са написани за всички тези приложения. Проблема е, че това не са единствените услуги, всеки месец се появяват нови. Следователно, за да се осигури на вътрешните потребители ефикасен интернет достъп, трябва да сме в крак с нововъзникващите приложения. Proxy за такива услуги ще бъдат достъпни, но след като новите услуги или протоколи са били на пазара известно време.

Password security

Предполага се, че на една машина ще работят повече от един потребител (вероятно ще бъдат дузина). Системният администратор (или човекът отговорен за политиката), ще трябва да въведе определени стандарти за ползването на пароли.

Първо трябва да се има предвид, че всяка система на пароли има вградени вътрешни слабости. Това е опасно, защото паролите са в самата сърцевина на схемите за сигурност на Unix системите. Всеки компромис със сигурността чрез пароли води до сериозен проблем. Обикновено единственото решение е всичките потребители да си сменят паролите. Съвременните схеми за защита с пароли са много добри, предлагайки едновременно криптирани пароли и в някои случаи скрити пароли.
Скриването на паролите е, когато файла /etc/passwd съдържа само указатели или символи, които служат като абстрактна представа за истинската криптирана парола на потребителя. Тази истинска парола се съхранява някъде върху диска на място недостъпно за кракерите.

Някои реализации не разполагат със свойството да скриват паролите. В такъв случай файла /etc/passwd ще бъде най-малкото видим за потребителите. Наистина паролите са в криптирана форма, но съвсем проста задача е да бъдат разшифровани. Ако могат да бъдат видяни те могат да бъдат разшифровани.
Паролите в техния изходен криптиран вид не трябва да бъдат видими за когото и да било. Модерните технологии предоставят средства за скриване на тези пароли и няма никаква основателна причина това да не се направи.

SSH

Secure Shell (SSH) осигурява безопасна и криптирана комуникация в Интернет. SSH е отлична замяна за Telnet или rlogin. Като SSH има 16-битова версия за Windows, но се изпълнява добре на които и да е TCP/IP реализации. SSH не е обикновено помощно средство (utility). То използва IDEA и RSA криптиране и е безопасно. Тази криптирана комуникация не е бавна. SSH напълно елиминира възможността за трети лица да се намесят във установената комуникационна връзка. SSH сесии не може да бъдат откраднати, нито може да бъдат sniff-нати. Единственото истинско неудобство е това, че другита страна също така трябва да използва SSH.

SSL

Преглед на SSL (Secure Socket Layer) protocol:

SSL е предложен от Netscape за сигурна комуникация по Internet. SSL работи над транспортния слой на ISO/OSI мрежовия слой и може да бъде използван, да подсигурява комуникацията между всеки протокол от приложния слой на TCP/IP пакета като HTTP, telnet, FTP. Протоколът HTTPS (Secure HTTP) е HTTP над SSL. URL на защитен Web сайт започва с https:// вместо с http://. SSL се надгражда върху транспортния слой. Самият протокол е разделен на слоеве. Той просто взема информацията от приложния слой, преформатира я и я предава на транспортния слой. SSL обработва съобщение по следния начин:

- изпращач (sender):
    - взема съобщението от по-горния слой;
    - фрагментира данните на подходящи блокове;
    - дава възможност за компресиране на данните;

• добавя код за автентикация на съобщението (message authentication code MAC);

• криптира данните;

• предава резултата на по-долния слой;

- получетел (receiver):

• взема данните от по-долния слой;

• декриптира ги;

• проверява данните с приетия за сесията MAC ключ;

• декомпресира данните, ако са били компресирани;

• сглобява съобщението от получените блокове;

• предава съобщението на по-горния слой.

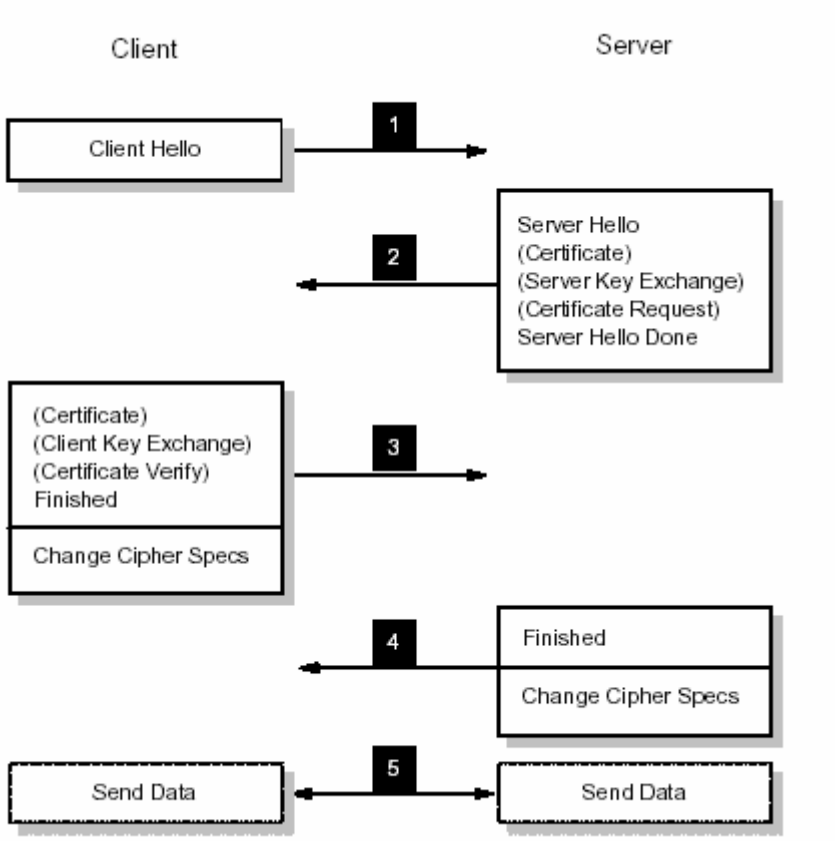
SSL осигурява предаване на криптирани данни през мрежата и позволява автентикация на сървъри. SSL гаранитира:

• поверителност - всички клиентски заявки и отговори от сървъри са криптирани, за да пазят поверителността на данните предавани по мрежата;

• непокътнатост на данните – данните, които се предават между клиент и сървър са предпазени от намесата на трета страна.

Допълнително от клиентите може да бъде поискано, да се автентицират на сървъра чрез техния собствен цифров сертификат.

Когато клиентът се свърже със защитен сървър, процесът познат като SSL ръкостискане (handshake) започва. Сървърът се представя на клиента първо с изпращане на своя дигитален сертификат. Ако клиентът се довери (trust) на сървъра, процесът продължава, включвайки споразумяване между сървървъра и клиента относно ключа на сесията, с който да се криптират всички съобщения, изпращани в двете посоки (от клиента към сървъра и обратно). Това се прави от клиента, който генерира произволен симетричен ключ, който се криптира с публичния ключ на сървъра, и после го изпраща на сървъра. Това би могло да бъде ключът на SSL сесията, ако сървърът се "съгласи". В противен случай сървърът и клиентът "преговарят", докато стигнат до споразумение за симетричен ключ, с който да се криптирата информацията, изпращана по Internet. Да отбележим, че за всеки клиент се поддържа отделна сесия, и сървърът отговаря за планирането на работата с клиента и сесийния ключ. Заявките, изпращани на сървъра, са криптирани. Това би могло да бъде доста значимо, ако използваме GET или POST методи с HTML форма, за да изпращаме поверителна информация, като например парола или номер на кредитна карта, на сървъра.



VPN

Виртуални частни мрежи (VPN), приложение:

Вероятно един от най-неправилните термини в компютърната индустрия днес е VPN. Изглежда, че въпреки, че всеки доставчик има VPN, който да ви продаде, в много от случаите VPN на един доставчик е пълната противоположност на този на друг доставчик. Например, не много отдавна един от основните телекомуникационни доставчици предложи VPN услуга включваща частна релейна мрежа, към която потребителите могат да се свързват, и използвайки тази мрежа, да се свързват към бизнес мрежата. Тя не включваше криптиране и единственото решение, осигуряващо реална защита, би бил какъвто и да е защитен механизъм, използван от клиента на входната точка на бизнес мрежата.

Друг VPN доставчик би се опитал да ви продаде специално конструирана хардуерна платформа предназначена за доставка точно на VPN услугите. Доставчиците на защита от тип Firewall ще се опитат да ви продадат VPN с тяхната Firewall платформа, а доставчиците на маршрутизатори ще се опитат да ви убедят, че VPN услугите в комплект с маршрутизатор (router) са най-подходящото решение за вас. И накрая съществуват решения, работещи с мрежови операционни системи, като например VPN в конфигурацията на Windows 2000.

VPN е механизъм, предоставящ защитена и криптирана комуникация в две конфигурации. Първата конфигурация е *user-to-network*, при която отдалечен

потребител се свързва към Internet и, използвайки VPN, може защитено да стане член на корпоративна мрежа. Това обикновено се обозначава като модел на отдалечен достъп за VPN. Другата конфигурация е, когато една офис мрежа използва VPN в комплект с Internet връзка за защитен достъп към другия край на VPN. Това обикновено се обозначава като VPN *site-to-site*. VPN с отдалечен достъп се използва, за да замести стандартния отдалечен достъп чрез dial-in или автентициран firewall достъп към мрежата. Моделът *site-to-site* се използва, за да се заобиколи необходимостта от WAN. И двете конфигурации позволяват съществено намаляване на разходите в сравнение с традиционните методи на достъп. Един от недостатъците на VPN модела все пак е по време на прекъсвания на Internet линиите. А когато не можем да се свържем към офис мрежата, защото една от междинните мрежи е получила прекъсване в оптичния канал, на кого можем да се обадим за помощ? С WAN или решение с отдалечен достъп винаги има доставчик на когото можем да се обадим за състоянието на връзката или да имаме техник на разположение. Не разполагаме със специален Internet номер, на който да се обадим за техническа поддръжка при проблеми с Internet връзката. Това е основният търговски минус на VPN решението в сравнение с традиционния подход.

Това са две по-често срещани използвания на VPN, но не са единствените причини за използване на VPN технологията. Без защита и обществените, и частните мрежи са достъпни за неоторизирано наблюдение и достъп. Частните мрежи са също толкова застрашени от атаки както и обществените. Системите защитени с парола не могат да контролират потока от данни предаван по мрежата.

Основни проблеми при защитата на информацията:

Следващия списък без съмнение е един изчерпателен списък на възможните проблеми на сигурността на информацията, с които човек може да се сблъска при поддръжка на мрежата. Тук са изброени видовете атаки и уязвими точки, които могат да възникнат при използване на VPN технологиите.

Мрежов “Sniffing”:

От създаването на Telnet, едно от най-сериозните предизвикателства за мрежовата сигурност е мрежовия трафик по канала на некриптирани данни или данни в чист текстов вид. Ако се върнем в дните, когато е бил написан TCP/IP протокола, Internet е бил виртуална частна мрежа използвана от университети и изследователи. Сигурността не е била най-важното, защото понятието подслушване на съобщения не е съществувало. Това е било преди бизнесът да стане водещ; в същност по това време търговското използване на Internet не е било разрешено. Както и да е, когато информацията пътува в чист текстов вид (както повечето SMTP, Telnet, HTTP, FTP видове трафик), един недоброжелател, който е получил физически достъп до мрежата може да слуша и чете всеки некриптиран трафик. Когато мрежата е атакувана от подслушвач, най-вероятно става въпрос за sniffing или snooping. Компания наречена Network General предлага продукт наречен Sniffer. Това устройство става толкова популярно за мрежови атаки, че името Sniffer става нарицателно за всяко приложение, което може да чете пакетите по мрежата. Способността на подслушвача да следи мрежата, обикновено е най-сериозния проблем със сигурността, с който се сблъскват администраторите на мрежи в предприятията. Но с висока степен на криптиране други не могат да четат информацията ни, пътуваща по мрежата. Това е най-доброто постижение във вътрешните и външни (Internet) VPN мрежи.

Виртуални частни мрежи VPN:

Логически VPN е просто едно разширение на частната мрежа. В действителност, частните мрежи са географски изолирани от отделните потребители и от други частни

мрежи чрез използването на несигурни комуникационни линии като Internet. Използването на сигурни протоколи от мрежово ниво, какъвто е IPSec, позволява да се създаде частна връзка между две отделни мрежи. Всъщност опаковането на данните преди да бъдат изпратени по несигурните extranet или intranet линии, се възприема и от заявителя, и автентикатора като частна защитена линия. Тъй като пакетите с данни се защитават с криптиране, всяко прекъсване на преноса води до нечетимост на данните. За разлика от използването на IPSec за пренос peer-to-peer, VPN мрежите използват специално предназначен за целта сървър, който е свързан към частната мрежа. Чрез допускането на връзки само към VPN-оторизирани потребители, VPN сървърът позволява частен трафик във физически обособената мрежа да протича нормално без изисквания за допълнителна вътрешна сигурност.

Преди да установим VPN връзка, се изисква VPN оторизация за клиента. Оторизацията се основава на правила за отдалечен достъп, контролирани от мрежовия администратор и dial-in характеристиките на потребителското искане за връзка. След като един потребител е получил право на достъп VPN връзката може да бъде осъществена. Windows 2000 включва два протокола, които се използват за пакетиране на данните в рамките на VPN:

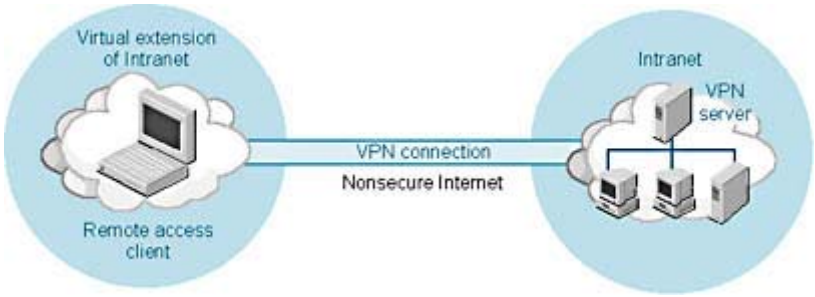
• Point-to-Point Tunneling Protocol (PPTP) - Осигурява криптиране на данните използвайки Point-to-Point криптирането на Microsoft;

• Layer Two Tunneling Protocol (L2TP) - Осигурява криптиране на данните, автентикация, и интегритет използвайки IPSec.

VPN с отдалечен достъп:

Изолиран, но имащ право на достъп потребител може да достигне защитените ресурси на частна или скрита мрежа като удостовери самоличността си на VPN мрежовия сървър и установи VPN връзка. Отдалечения потребител може да бъде вкъщи или на път и да се нуждае от връзка чрез един ISP (Internet Service Provider) или отдалечения потребител може да бъде част от същия intranet, отделен от защитената или скрита мрежа. Фактът, че имате VPN връзка към частна мрежа все още не означава, че имате пълен достъп. Все още е необходимо да получите права на достъп към някои специфични ресурси.

След като клиентът е осъществил VPN връзка към VPN сървъра, потребителят получава директен достъп до частната мрежа.

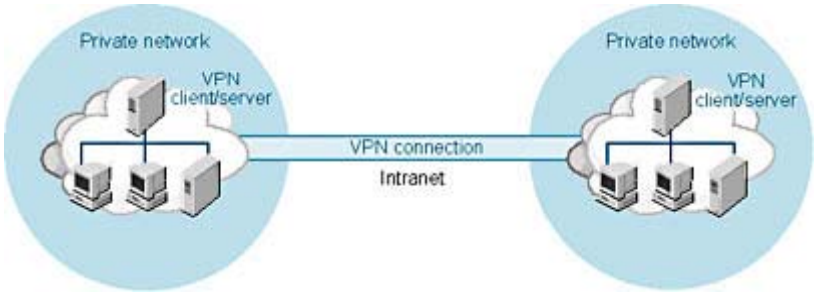


Отдалечен достъп за Internet-организирана VPN.

Организация Router-to-Router на VPN:

VPN връзката може да бъде осъществена и между две обособени частни мрежи. В този случай и VPN клиента, и VPN сървъра са маршрутизатори. Както и при VPN мрежите с отдалечен достъп, router-to-router VPN мрежите могат да включват анклави,

които са част от една и съща intranet структура или може да включват частни мрежи, които изискват Internet комуникационна инфраструктура. Фугурата отдолу показва router-to-router връзка към VPN мрежа с intranet структура.



Тук VPN клиента удостоверява самоличността си на VPN сървъра и се постига сигурна VPN връзка. И отново на логическо ниво и в двата случая, частните мрежи изглеждат физически свързани.

Защита срещу IP Spoofing атаки

IP spoofing атаките може да бъдат осуетени чрез конфигуриране на мрежата, така че да отхвърля пакети от мрежа, с локален адрес (т.е. отхвърля пакети, които имат адрес работна станция на вътрешната мрежа). Това най-често се прави с рутер (router).Router-те работят чрез прилагане на филтри на постъпващите пакети. Например, те могат да спрат специални типове на пакети да достигнат мрежата, която защитават.

Макар че router-те са решение на spoofing проблема, те също работят чрез изпитване на изходния адрес. По този начин те могат само да защитят срещу постъпващи пакети, с адрес от защитаваната вътрешна мрежа. Ако мрежата се доверява (trust) на външни хостове, router-ите не защитават срещу spoofing атака, идваща от тези хостове.

Ако съществува firewall, това не защитава автоматично spoofing атаки. Ако се позволява вътрешни адреси, да имат достъп извън firewall-а, то мрежата е уязввима. Предпазването на мрежата можа да се осъществи, чрез наблюлдение (monitoring). Стартира се индификация на пакетите, които твърдят , че излизат от вътрешната мрежа, но се стремят да получат достъп през firewall-а или до първата мрежа, до която се натъкнат

Има няколко класа пакети, които би трябвало да се наблюлдават. Всеки TCP пакет с адрес на подателя или получателя, които не е от мрежата. Тези пакети обикновено няма да излязат от изходната мрежа, освен ако има проблем с router-а, или пакетите в действителност, произхождащи от външна мрежа.

Инсталиране на WWW сървър

Един WWW сървър (Web server) се създава на базата на инсталиран HTTPD (daemon) програмен пакет, реализиращ HTTP протокол. Функцията на сървъра е да обработва и изпълнява получените заявки за достъп до съдържащата се в неговата структура информация от WWW клиенти. Достъпът до хипермедийни документи в Internet се осъществява чрез протокол HTTP с помощта на HTTPD сървър (WWW сървър). Съдържанието на извлечения от един WWW сървър документ посредством WWW клиент се визуализира на екрана на локалния компютър на потребителя. Не е възможно в тази схема (клиент-сървър) два WWW клиента да обменят документи директно помежду си.

Първата стъпка при инсталиране на WWW сървър е да се определи хардуерната платформа и операционната система, които ще се използват. Програмно осигуряване за WWW сървъри е достъпно за всички платформи (персонални компютри и техните разновидности, работни станции), така че изборът се базира на съображения за икономия, удобство и съвместимост със съществуващите машини в локалната мрежа. Важно е да се отбележи, че за WWW сървър не е необходимо да се избира твърде мощен компютър. Главното предназначение на сървъра е просто да съхранява и изпраща файлове. Ето защо скоростта на мрежата е много по-важна отколкото мощността на процесора, което трябва да се свързва и с избор на подходящи мрежови контролери. Друго изискване е по отношение обема на дисковата и оперативнтана памет. Трябва да се осигури достатъчно дисково пространство за данните, които ще бъдат съхранявани. Обемът оперативна памет е фактор, който често се пренебрегва за сметка на бърз процесор. Ефективната производителност на един WWW сървър е в пряка зависимост от обема RAM.

Втората стъпка е да се осигури връзка на избрания компютър с Internet. Компютърът трябва да се включи към мрежата чрез интерфейсна мрежова карта (платка) и съответните програмни драйвери. В допълнение към това е необходимо в компютъра да се инсталира софтуерният стек на протокола TCP/IP. Софтуерът за TCP/IP трябва да се конфигурира с IP адреса на компютъра и IP адресите на машините - Gateway (шлюз) и DNS сървъри (сървър за разрешаване на съответствията между IP адрес и символично име на машина). Необходимо е също да се актуализират таблиците на всички рутери в мрежата, така че TCP/IP трафикът от новосъздадения сървър да достига Internet. Това включва също установяване на адреса на подходящия за мрежата proxy сървър. Сървърите DNS (Domain Name Server) за съответния домейн (област) трябва да се актуализират с името, идентификатора и IP адреса на компютъра, който става сървър. Инсталирането на софтуера за TCP/IP е твърде различно за различните платформи.

Следваща стъпка е да се осигури програмен пакет за инсталиране на WWW сървър, предназначен за конкретния компютър и операционна система. Малко на брой са програмните системи лесни за инсталиране и администриране. Всяка операционна среда има свойте особености. Проблемите за защита на информацията в сървъра от разрушаване и неправомерен достъп са от първостепенно значение. Осигуряване на нормална и дългосрочна работа на един Web сървър предполага известна предварителна подготовка по операционни системи, мрежи и запознаване с характеристиките и възможностите на конкретния софтуер.

• Apache - един от най-разпространените, freeware, поддържа версии за Unix/Linux/Windows NT;

• Microsoft Information Server - най-разпространен е за платформа Windows NT, част е от инсталационния кит на Windows NT 4.0 Server, за Windows NT 4.0 Workstation има допълнителна опция Personal Information Server;

• Netscape Server - комерсиални версии (comercialeware) за платформи Unix, Windows NT);

• NCSA - всички платформи;

• CERN - всички платформи (остаряла концепция);

• Roxen - комерсиални версии (comercialeware) за Unix;

• Jigsaw - реализация на Java;

• MacHTTP - shareware версии за Mac;

• WebStar - shareware версии за Windows 95, 98, NT.

И все пак, това може да стане най-лесно чрез самата работна група W3 - адресът на началната страница на CERN е:
http://info.cern.ch/hypertext/WWW/TheProject.html където, може да се избере необходимата за конкретния случай страница с информация за различни програмни среди и програми. Достъпът посредством FTP протокол се осъществява на следния адрес:
ftp://ftp.w3.org

WWW клиенти:

Web браузърите в своето развитие следват повишаващите се изисквания на потребителите на информация във Internet. Голяма част от тях в последните си версии предлагат мултимедийни възможности - интерпретация на аудио, видео и анимирани графични компоненти. Мобилността и повишената ефективност на WWW клиентите се подсигурява от възможност за включване на допълнителни програмни компоненти - Plug-Ins, с които могат да се изпълняват по-сложни задачи. Почти всички браузъри включват вече възможност за разпознаване на Java.

WWW клиентите се различават главно по техните възможности. Съществуват следните типове клиенти:

• Текстоориентирани WWW клиенти - Lynx;

• Ориентирани към графика WWW клиенти - X - версията на Mosaic, Mosaic за PC и Macintosh, Netscape Navogator, Netscape Communicator, Microsoft Internet Explorer, които използуват графичните възможности на съответната система;

• WWW клиенти с възможност за възпроизвеждане на документи, които освен форматиран текст, съдържат друга информация - звук, неподвижни и подвижни изображения. В този случай са необходими апаратни средства и програми-драйвери, инсталирани върху компютъра, използуван като WWW клиент;

• WWW клиенти, които могат да стартират определени приложни програми с цел визуализацията на документи от определен тип (напр. ако WWW документ съдържа хипервръзка към документ с Microsoft Word for Windows формат, автоматично се стартира WinWord и се визуализира извлечения документ);

• WWW клиенти, осигуряващи достъп до други Internet услуги (FTP, Gopher, WAIS и UseNet сървъри, Telnet сесии).

Гореописаните технологии и протоколи трябва да бъдат използвани от организации и фирми, предлагащи интернет услуги. Чрез тези технологии интернет доставчиците могат да грантират максимална сигурност на продаваните данни между техните сървъри и клиенти от неправомерен достъп, стабилна връзка, загуби на данни сведени до минимум.

Използвана литература:

1. TCP – Tutorial by Martin Murhamer
2. Special Edition Using Linux by Jack Tackttt , Steve Burnett , Rob Napiar, Jeff Tranter
3. Internet Security - Bradly Dunsmore, Jeffrey Brown , Michael Cross
4. Публикувани статии в интернет.