Proftpd - informacje ogólne

Położenie: (nie dotyczy)

Filozofia FTP

Jest to metoda (*File Transfer Protocol*), za pomocą której możemy połączyć się ze zdalnym dyskiem (zasobami), wysłać tam swoje pliki lub pobrać je na swój dysk. Zdalny dysk ma następującą strukturę:

/home/ftp/pub (zawiera pliki do pobrania)

/home/ftp/uploads (jest to folder, do którego możesz wysłać swoje pliki; po wysłaniu będą one niewidoczne, tylko osoba znająca ich nazwę będzie mogła je pobrać, np. w przeglądarce internetowej za pomocą adresu *ftp://192.168.0.9/uploads/jakisPlik.pdf*) lub

/home/ftp/incoming (alternatywna nazwa folderu, do którego możemy wysłać swoje pliki)

Uwaga: Użytkownik zalogowany przez FTP nie ma dostępu do folderu */home*, lecz jest od razu „chrootowany" do folderu */home/ftp*, który staje się dla niego katalogiem głównym poza który nie może wyjść.

Sposób łączenia się

Uwaga: Tryb tekstowy tylko w *Linux* (klient windowsowy nie obsługuje trybu pasywnego).

$ **ftp ftp.helion.pl**

name: **anonymous**

Password: **nasz@adresEmail.com**

ftp> **dir** (sprawdzamy zawartość folderów)

Uwaga:

1. Jeśli pojawi się pytanie zapory o pozwolenie na połączenie - należy na nie pozwolić. Jeśli nie pojawi się takie pytanie (a firewall wciąż będzie je blokował) - można firewalla zresetować: *Ustawienia / Aktualizacje i zabezpieczenia / Zabezpieczenia Windows / Zapora i ochrona sieci / Przywróć domyślne ustawienia zapór*.
2. Jeśli wykonywanie polecenia zatrzyma się, należy wyłączyć firewall (to nie zadziała jednak na maszynie wirtualnej z kartą sieciową NAT).
3. Jeśli pojawi się błąd „*500 Illegal port command*", należy włączyć tryb pasywny i ponowić polecenie.

ftp> **passive** (ustawiamy połączenie na tryb pasywny; tylko *Linux*)

Uwaga: W przypadku *Windows*, będzie to polecenie „**quote pasv**", które jednak przełącza jedynie serwer w tryb pasywny, a nie klienta (*ftp.exe*) w *„Wierszu poleceń"* (to odwieczny błąd Windowsa). Aby uzyskać tryb pasywny na kliencie, należ korzystać z innego oprogramowania, np. *FileZilla*, *WinSCP*, *Windows Explorer* (używa trybu pasywnego domyślnie, zob. *HKEY_CURRENT_USER\Software\Microsoft\FTP\Use PASV: Yes*).

ftp> **cd pub**

ftp> **dir** (przeglądaj kolejne foldery i wchodź w nie za pomocą polecenia „cd")

ftp> **get jakisPlik.zip** (pobieramy jakiś plik)

ftp> **help** (warto wydać to polecenie, jeśli nie wiesz, co robić)

ftp> **cd ..** (wychodzimy z „*pub*")

ftp> **dir** (patrzymy, czy mamy folder *incoming* lub *uploads*, które służą do wysyłania plików... jeśli któryś z nich jest, próbujemy wysłać do niego jakiś nasz plik)

ftp> **cd incoming**

ftp> **put /home/***nazwaUżytkownika***/plik.jpg** (wysyłamy na serwer FTP jakiś plik)

ftp> **dir** (wysłanych plików najczęściej nie można wyświetlić... zastanów się dlaczego)

ftp> **bye**

Typy połączeń

Port 21 - to port poleceń (*command port / control port*);

Port 20 - to port do przesyłania danych (*data port*).

Przesyłanie plików może działać w dwóch trybach:

- **Tryb aktywny** (*PORT mode*) – klient otwiera po swojej stronie dwa losowe porty o wysokim numerze (powyżej 1023): jeden kontrolny do przywitania serwera (np. 1026) i drugi do wymiany danych (n+1, czyli 1027) i wysyła o tym informację (komunikat PORT 1027) do serwera na jego port 21 (klient tworzy tzw. „*command channel*”); serwer odpowiada (potwierdza: ACK) z portu 21, ale wysyła jednocześnie (ze swojego portu 20) na podany przez klienta port 1027 swoje dane (serwer tworzy tzw. „*data channel*”). Uwaga: Nie będzie działać ten tryb zza maskarady, gdyż serwer nie będzie w stanie nawiązać połączenia z klientem ukrytym za routerem; często występuje także problem z firewall’em po stronie klienta, który nie pozwala serwerowi na utworzenie „*data channel*” z jego portu 20 na swój losowy port o wysokim numerze.
- **Tryb pasywny** (*passive mode*) – klient otwiera dwa porty (*command & data channel*) i wysyła z portu „*command*” do serwera na jego port 21 polecenie PASV (klient mówi serwerowi, że jest pasywny i czeka aż serwer otworzy jakiś wysoki numer portu); gdy klient otrzyma od serwera ten numer portu (losowy wysoki numer), nawiązuje do niego połączenie z portu *data* (klient tworzy więc „*data channel*”). Wada: Klient musi mieć pozwolenie na wykonywanie dowolnych połączeń na zewnątrz (inaczej mogą, choć nie muszą, być problemy); rozwiązanie: reguła iptables *RELATED* zapamiętuje żądanie otwarcia portu przez sesję FTP i wypuszcza połączenie związane z tą sesją. Występuje tutaj jeszcze jeden problem: serwer jest w mniejszym stopniu bezpieczny, gdyż musi pozwalać klientom na łączenie się z całą gamą swoich portów (jest to natomiast zaleta dla klienta). Na niektórych serwerach można jednak ustawić zakres tych portów. Uwaga: W trybie pasywnym, port 20 nie bierze udziału w komunikacji ani na serwerze, ani na kliencie.

Ustawienia iptables

:: Ustawienia wspólne dla aktywnego i pasywnego transferu na serwerze:

Akceptujemy połączenia przychodzące kontrolne (*control / command connections*) zainicjowane przez klienta z dowolnego portu (powyżej 1023) do portu 21 serwera:

iptables -A INPUT -p tcp -m tcp --dport 21 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT -m comment --comment "*Allow ftp connections on port 21*"

Akceptujemy potwierdzenia serwera (ACK) wysyłane z portu 21 do dowolnego portu klienta (powyżej 1023):

iptables -A OUTPUT -p tcp -m tcp --sport 21 -m conntrack --ctstate ESTABLISHED -j ACCEPT -m comment --comment "*Allow ftp connections on port 21*"

:: Ustawienia aktywnego połączenia na serwerze:

Akceptujemy połączenia z portu 20 serwera (*data connection*) do wysokiego portu klienta (powyżej 1024):

iptables -A OUTPUT -p tcp -m tcp --sport 20 -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT -m comment --comment "*Allow ftp connections on port 20*"

Akceptujemy potwierdzenia klienta (ACK) wysyłane z wysokiego portu klienta (powyżej 1024) do serwera na port 20:

iptables -A INPUT -p tcp -m tcp --dport 20 -m conntrack --ctstate ESTABLISHED -j ACCEPT -m comment --comment "*Allow ftp connections on port 20*"

:: Ustawienia pasywnego połączenia na serwerze:

Akceptujemy wszelkie połączenia przychodzące z wysokich portów klienta (powyżej 1023) na wysokie porty serwera (powyżej 1023):

iptables -A INPUT -p tcp -m tcp --sport 1024: --dport 1024: -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT -m comment --comment "*Allow passive inbound connections*"

Akceptujemy wysyłanie potwierdzeń serwera (ACK) wysyłane z wysokich portów (powyżej 1014) do wysokich portów klienta (powyżej 1024):

iptables -A OUTPUT -p tcp -m tcp --sport 1024: --dport 1024: -m conntrack --ctstate ESTABLISHED -j ACCEPT -m comment --comment "*Allow passive inbound connections*"

Uwaga: połączenia na porty 113 i 1080 związane z protokołami IDENT i SOCKS muszą być w iptables odrzucane w sposób jawny, bo inaczej jest kilkudziesięciosekundowe opóźnienie (tzw. „przekroczenie czasu oczekiwania na połączenie”). Czyli powinno być:

iptables -A INPUT -p tcp --dport 113 -j REJECT --reject-with icmp-port-unreachable

iptables -A INPUT -p tcp --dport 1080 -j REJECT --reject-with icmp-port-unreachable

Sprawdzamy na koniec, czy jest otwarty port 21:

netstat -atnp | grep -w 21

W razie konieczności sprawdzamy ustawienia iptables:

cat /etc/sysconfig/iptables

lub czyścimy wszystkie reguły:

iptables -F

Uwaga: W dystrybucji *SuSE,* iptables nie funkcjonuje jako usługa i nie jest widoczne w wykazie procesów; jest tylko skrótem (*/usr/sbin/iptables*) do */usr/sbin/xtables-multi.* Należy także wyłączyć lub skonfigurować (w centrum sterowania YaST):

- *AppArmor Configuration*
- *Firewall*
- *Services Manager / SuSEfirewall2**

Katalogi FTP

Prawa systemowe do katalogów są ważniejsze i mają większą moc niż prawa ustawione w plikach konfiguracyjnych *proftpd*. Kłopoty z logowaniem jako anonymous mogą być spowodowane właśnie ustawieniem niewłaściwych praw do folderu */home/ftp* lub niewłaściwym właścicielem. Struktura katalogów (takie prawa należy nadać w głąb katalogów, opcja *chmod -R*):

/home/ftp

drwxr-xr-x	*ftp*	*ftp*	*pub*
drwxr-xr-x	*ftp*	*ftp*	*uploads*
-rw-r--r--	*ftp*	*ftp*	*.readme.txt*
-rw-r--r--	*ftp*	*ftp*	*.welcome.msg*

Należy pamiętać, iż użytkownik systemowy ftp powinien mieć shell */bin/false*, a jego katalog domowy powinien prowadzić do */home/ftp*. W celu zaoszczędzenia miejsca na dysku można współdzielić te same zasoby zarówno w proftpd jak i w sambie.

Aby włączyć wybrane foldery spoza katalogu FTP do tego katalogu, należy użyć polecenia:

 # mkdir /home/ftp/distfiles && mount --bind /usr/portage/distfiles /home/ftp/distfiles

oraz włączyć na stałe do montowania w */etc/fstab*:

/usr/portage/distfiles /home/ftp/distfiles none bind 0 0

Autoryzacja PAM

Niestety, występują problemy z autoryzacją proftpd 1.2.10-r1 poprzez pam 0.77-r6. Niektóre błędy zostały już zgłoszone:

Bugzilla Bug 63196

Bugzilla Bug 83312

Polegają one między innymi na tym, że podczas kompilacji *proftpd* w Gentoo, *emerge* wykrywa w systemie *pam-pwdb*, chociaż go nie ma (nie jest domyślnie instalowane). Należy więc jeszcze raz skompilować pam z flagą "pwdb". Niestety, to nie rozwiązuje w całości problemu. Jeśli w proftpd.conf istnieje opcja

AuthOrder mod_auth_pam.c

autoryzacja systemowego użytkownika kończy się błędem. Jeśli doda się mod_auth_unix.c to wszystko jest OK.

Zawartość pliku /etc/pam.d/ftp:

auth required /lib/security/pam_listfile.so item=user sense=deny file=/etc/ftpusers onerr=succeed
auth required /lib/security/pam_pwdb.so shadow nullok
account required /lib/security/pam_pwdb.so
session required /lib/security/pam_pwdb.so

Problemy

Podczas logowania: Unable to set anonymous privileges

Rozwiązanie: Należy skompilować *proftpd* z flagą *USE="-acl".* ACL (*Access Control Lists*) nie jest wykorzystywane, jeśli w jądrze nie zaznaczono opcji *ReiserFS Extended / ACL* i nie zamontowano partycji z tą opcją. ACL służy do przyznawania plikom rozszerzonych niestandardowych praw.

Publiczne anonimowe serwery FTP

- ftp.man.lodz.pl
- ftp.gnu.org
- ftp.pureftpd.org
- ftp.vim.org
- cdaweb.gsfc.nasa.gov*
- heasarc.gsfc.nasa.gov*

* Połączenie do tych serwerów wymaga szyfrowania (SSL / TLS). W *Gentoo* bez problemu radzi sobie z tym klient FTP z pakietu **net-ftp**. Jednak domyślny klient w *openSuSE* (*tnftp*) nie obsługuje szyfrowania, dlatego należy go zamienić na pakiet **lftp**. Procedura:

\# **zypper rm tnftp**

\# **zypper in lftp**

user@localhost:~> **lftp -u anonymous** *cdaweb.gsfc.nasa.gov*

(lub: **lftp anonymous@cdaweb.gsfc.nasa.gov**)

Alternatywnie:

user@localhost:~> **lftp**

lftp :~> **set ftp:ssl-force true**

(lub wręcz przeciwnie, gdyby były problemy: **set ssl:verify-certificate no**)

lftp :~> **connect** *cdaweb.gsfc.nasa.gov*

lftp cdaweb.gsfc.nasa.gov:~> **login anonymous**

Hasło: *twoj@adres-email.com*

lftp anonymous@cdaweb.gsfc.nasa.gov:~> **dir**

Dokumentacja

Pełna dokumentacja znajduje się tu: www.proftpd.org
Błędy w pliku konfiguracyjnym: **/usr/sbin/proftpd -t**

Ostatnia aktualizacja: 5 stycznia 2024.