

Grupy dyskusyjne w systemie Linux.

Autorzy: Mariusz Łęcznar , Stanisław Medygrał 4FDS

STRESZCZENIE

Celem pracy jest przedstawienie podstawowych zagadnień związanych z grupami dyskusyjnymi pod systemem Linux.

Pierwszy rozdział poświęciliśmy teorii grup dyskusyjnych usenetu. Zdefiniowaliśmy czym są te grupy i jak one działają. Następnie dokonaliśmy krótkiego przeglądu najczęściej stosowanych serwerów grup dyskusyjnych. Wyróżniliśmy tutaj trzy serwery : C NEWS, NNTP i demon NNTPD, Internet News.

Dokładnej analizie poddaliśmy tylko serwer INN , gdyż obecnie jest on zdecydowanie najbardziej popularny. Przedstawiliśmy jego budowę i opisaliśmy sposób działania. Opisaliśmy jak go zainstalować i jak go później skonfigurować. Zwróciliśmy szczególną uwagę na pliki konfiguracyjne, ich przeznaczenie i zawartość, celem pokazania możliwości konfiguracyjnych jakie w nich drzemią.

Kolejna część pracy została przeznaczona na omówienie sposobów zarządzania INN'em. Przedstawiliśmy sposób użycia polecenia *ctlinnd* i jego składnie. Pokazaliśmy jak za jego pomocą ,na przykład: tworzyć nowe grupy dyskusyjne, zmieniać je i usuwać.

W części praktycznej zawarliśmy szczegółowy opis konkretnych czynności jakie musi wykonać użytkownik, aby uruchomić na swoim komputerze serwer grup dyskusyjnych. Przedstawiona przez nas konfiguracja była realizowana na systemie operacyjnym Linux Debian.

SPIS TREŚCI

Grupy dyskusyjne w systemie Linuks.....	0
Streszczenie.....	1
1. Czym są grupy dyskusyjne usenetu i jak są obsługiwane.....	3
2. Przegląd wybranych serwerów grup dyskusyjnych.....	4
2.1. C news.....	4
2.2. nntp i demon nntpd.....	4
2.3. Internet news.....	5
3. Omówienie serwera inn.....	5
3.1. Budowa i działanie inn.....	5
3.2. Instalacja i podstawowa konfiguracja <i>inn</i>	6
3.3. Pliki konfiguracyjne <i>inn</i>	7
3.3.1. Plik <i>inn.conf</i> - parametry globalne.....	7
3.3.2. Pliki <i>active</i> i <i>newsgroups</i> – konfigurowanie grup dyskusyjnych.....	7
3.3.3. Plik <i>newsfeeds</i> – konfigurowanie dostarczania grup do innych serwerów.....	8
3.3.4. Plik <i>nntpsend.ctl</i> – konfiguracja programu <i>nntpsend</i>	10
3.3.5. Plik <i>incoming.conf</i> – kontrolowanie dostępu przeglądarki.....	10
3.3.6. Plik <i>nntp.access</i> – konfigurowanie programu <i>nntpd</i>	11
3.3.7. Plik <i>expire.ctl</i> – wygasanie artykułów w grupach.....	12
3.3.8. Plik <i>control.ctl</i> – obsługiwanie wiadomości kontrolnych.....	13
4. Zarządzanie <i>inn</i> – polecenie <i>ctlinnd</i>	15
4.1. Dodawanie nowej grupy.....	16
4.2. Zmiana grupy.....	16
4.3. Usuwanie grupy.....	16
4.4. Przenumerowanie grupy.....	16
4.5. Używanie serwera przez przeglądarki grup – pozwalanie i zabranianie.....	17
4.6. Odmowa połączenia od innego serwera.....	17
4.7. Pozwolenie na połączenia od innego serwera.....	17
4.8. Zamknięcie serwera grup.....	17
4.9. Restart serwera grup.....	18
4.10. Wyświetlanie statusu pobierania plików z innego serwera.....	18
4.11. Odłączenie dostarczania plików z innego serwera.....	18
4.12. Rozpoczynanie dostarczania plików z innego serwera.....	18
4.13. Anulowanie artykułu.....	19
5. Instalacja i konfiguracja <i>inn</i> na linux debian.....	19
5.1. Informacje wstępne.....	19
5.2. Instalacja.....	19
5.3. Konfiguracja.....	19
5.3.1. Zmiana ustawień globalnych- plik <i>inn.conf</i>	19
5.3.2. Modyfikacja pliku <i>newsfeeds</i>	20
5.3.3. Konfiguracja pliku <i>readers.conf</i>	20
5.3.4. Dodanie nowej grupy.....	21
5.4. Przykładowe pliki konfiguracyjne z naszego serwera.....	21
5.4.1. Plik <i>inn.conf</i>	21
5.4.2. Plik <i>newsfeeds</i>	24
5.4.3. Plik <i>readers.conf</i>	25
6. Literatura.....	26

1. CZYM SĄ GRUPY DISKUSYJNE USENETU I JAK DZIAŁAJĄ.

Grupy dyskusyjne Usenetu są jedną z najważniejszych usług oferowanych przez dzisiejsze sieci komputerowe. Stanowią niezastąpione źródło informacji.

Usenet nie jest częścią organizacji, ani nie ma żadnej centralnej władzy zarządzającej. W zasadzie taki już jest Usenet, że poza opisem technicznym nie da się go praktycznie zdefiniować. Gdyby jednak podjąć taką próbę to można by było zdefiniować Usenet jako współpracę ośrodków wymieniających wiadomości grup dyskusyjnych Usenetu. Aby stać się ośrodkiem Usenetu, wystarczy znaleźć inny ośrodek Usenetu i uzgodnić z jego właścicielami sposób i prawa wymiany wiadomości grup dyskusyjnych. Udostępnianie wiadomości grup dyskusyjnych innym ośrodkom nosi w języku angielskim nazwę *feeding* (dosłownie: *karmienie*) czasem używane jest określenie *dostarczanie*. [1]

Podstawową jednostką grup dyskusyjnych Usenetu jest *artykuł*. Jest to wiadomość, napisana przez użytkownika i „wysłana” do sieci. Artykuły są wysyłane do jednej lub kilku *grup dyskusyjnych*. Można powiedzieć, że grupa dyskusyjna to forum artykułów związanych z jakimś tematem. Wszystkie grupy dyskusyjne są uporządkowane w pewnej hierarchii, a nazwa grupy wskazuje miejsce w tej hierarchii. Często na tej podstawie łatwo jest stwierdzić, czego dotyczy dana grupa. Artykuły te są następnie wymieniane pomiędzy wszystkimi ośrodkami Usenetu, które chcą udostępnić daną grupę. Gdy dwa ośrodki ustalą, że będą wymieniać wiadomości, mogą przysyłać sobie dowolne grupy, a nawet dodawać swoje lokalne hierarchie grupy.

Każdy użytkownik pisze artykuły w specjalnej aplikacji, tak zwanej przeglądarce grup dyskusyjnych (ang. *newsreader*), która odpowiednio je formatuje w celu przesłania do lokalnego serwera grup dyskusyjnych. W środowiskach uniksowych przeglądarka grup zwykle używa polecenia *inews* do przesłania artykułów do serwera za pomocą protokołu TCP/IP. Możliwe jest jednak również zapisanie artykułu bezpośrednio do pliku w specjalnym katalogu nazywanym buforem grup. Gdy tak przygotowana wiadomość zostanie dostarczona do lokalnego serwera grup dyskusyjnych, bierze on odpowiedzialność za dostarczenie artykułu do innych użytkowników grupy. Grupy są rozpowszechniane w sieci za pomocą różnych protokołów transportowych. Kiedyś najczęściej korzystano z UUCP, ale obecnie główny ruch jest generowany przez ośrodki internetowe. Ośrodki w Internecie zwykle opierają się na oprogramowaniu TCP/IP, które wykorzystuje protokół NNTP (*Network News Transfer Protocol*). NNTP jest opisany w RFC-977. Jest on odpowiedzialny za przysyłanie grup między serwerami i zapewnia pojedynczym użytkownikom dostęp do zdalnych hostów. NNTP oferuje trzy różne sposoby przysyłania grup. Jeden to wersja *ihave/sendme* działająca w czasie rzeczywistym, nazywana także *wciskaniem* (ang. *pushing*) grup. Druga technika nosi nazwę *ściągnięcia* (ang. *pulling*) grup i w niej klient prosi o listę artykułów w danej grupie lub hierarchii, które dotarły do serwera po określonym czasie, i wybiera te, których nie ma w pliku historii. Trzecia technika służy do interaktywnego czytania grup i pozwala użytkownikowi lub jego przeglądarce grup na pobieranie artykułów z zadanych grup oraz wysyłanie artykułów z niepełną informacją w nagłówku. W każdym ośrodku grupy dyskusyjne są przechowywane w hierarchii katalogów poniżej */var/spool/news*, gdzie każdy artykuł znajduje się w oddzielnym pliku, a każda grupa w oddzielnym katalogu. Nazwa katalogu jest budowana na podstawie nazwy grupy, z tym że poszczególne człony są kolejnymi podkatalogami. I tak, artykuły z *comp.os.linux.misc* są przechowywane w */var/spool/news/comp/os/linux/misc*. Artykułom w grupie są przypisywane numery w kolejności, w jakiej artykuły nadchodzą. Tymi numerami nazywane są kolejne pliki. Zakres numerów aktualnie dostępnych artykułów jest przechowywany w pliku *active*, który służy jednocześnie jako lista artykułów znanych danemu ośrodkowi.

Ponieważ miejsca na dysku stopniowo ubywa, trzeba więc po jakimś czasie wyrzucać artykuły. Zwykle artykuły z pewnych grup i hierarchii wygasają po określonej liczbie dni od ich przybycia. Może to zmienić autor, określając datę wygaśnięcia w polu Expires: nagłówka artykułu. [1]

2. PRZEGLĄD WYBRANYCH SERWERÓW GRUP DYSKUSYJNYCH

2.1. C NEWS

Jednym z popularniejszych pakietów oprogramowania grup dyskusyjnych jest *C News*. Został zaprojektowany dla ośrodków obsługujących grupy dyskusyjne przez łącza UUCP.

C News przechowuje swoją konfigurację w plikach w katalogu */etc/news*, a większość jego plików binarnych znajduje się w katalogu */usr/lib/news*. Artykuły są przechowywane w katalogu */var/spool/news*. Wymagane jest, aby praktycznie wszystkie pliki w tych katalogach były własnością użytkownika **news** lub grupy **news**.

Szczegółowych informacji na temat instalacji i konfiguracji tego serwera nie przedstawiamy, gdyż obecnie zdecydowanie najpopularniejszym serwerem grup dyskusyjnych jest **inn**, i to właśnie na jego przykładzie pokażemy jak skonfigurować odpowiednio taki serwer.

2.2. NNTP I DEMON NNTPD

Protokół przesyłania wiadomości w sieci Usenet, NNTP (*Network News Transfer Protocol*), reprezentuje zupełnie odmienne podejście do wymiany grup dyskusyjnych, niż *C News* i inne serwery grup bez wbudowanej obsługi NNTP. Do przesyłania artykułów pomiędzy maszynami nie korzysta z technologii wsadowej charakterystycznej dla UUCP, ale pozwala wymieniać artykuły przez interaktywne połączenie sieciowe. NNTP nie jest pakietem oprogramowania, ale standardem internetowym opisanym w RFC-977. Korzysta z połączeń strumieniowych, zwykle działających w oparciu o TCP pomiędzy klientem w sieci a serwerem, który przechowuje grupy na swoim dysku lokalnym. Połączenie strumieniowe pozwala klientowi i serwerowi na interaktywne negocjowanie przesyłania artykułów prawie bez opóźnień, za czym idzie mały stopień ich dublowania. Jeśli uwzględnimy jeszcze wysoką przepustowość Internetu, otrzymujemy rozwiązanie znacznie przewyższające możliwości dotychczasowego UUCP. NNTP jest idealnym narzędziem, które daje dostęp do grup wielu klientom w sieci lokalnej. NNTP zapewnia także czynny i bierny sposób przesyłania grup, potocznie zwany „wciskaniem” i „ściągnięciem”.

Niedoskonałością NNTP jest to, że istnieje możliwość fałszowania informacji o nadawcy (ang. *spoofing*). Rozszerzenie NNTP pozwala na uwierzytelnianie użytkowników przy pewnych poleceniach, co jakoś zabezpiecza przed nadużywaniem twojego serwera grup dyskusyjnych. Istnieje szereg pakietów NNTP. Jednym z bardziej popularnych jest demon NNTP, znany także jako *implementacja wzorcowa* (ang. *reference implementation*). Został napisany przez Stana Barbera i Phila Lapsleya jako ilustracja RFC-977. Podobnie jak większość dobrego oprogramowania, tak i ten pakiet można obecnie znaleźć w większości dystrybucji Linuksa. Pakiet *nntpd* zawiera serwer, dwa klienty ściągające i wciskające wiadomości oraz zamiennik dla *inews*. Działają one w środowisku B News, ale po niewielkich zmianach będą także działać z *C News*. [1]

W przypadku uruchamiania dużych ośrodków grup dyskusyjnych, powinno zainteresować się pakietem *InterNet News*, inaczej INN. Zapewnia on transport grup zarówno przez NNTP, jak i UUCP, co jest zdecydowanie lepsze niż *nntpd*.

2.3. INTERNET NEWS

Demon Internet News (INN) jest chyba najpopularniejszym z obecnie używanych serwerów grup dyskusyjnych. Charakteryzuje się bardzo dużą elastycznością. Jest odpowiedni dla wszystkich ośrodków udostępniających grupy dyskusyjne. INN doskonale się skaluje i jest przystosowany do dużych ośrodków grup dyskusyjnych. Serwer INN składa się z szeregu elementów, z których każdy ma własne pliki konfiguracyjne.

Większość zainstalowanych serwerów to właśnie INN. Jego instalacja jest stosunkowo prosta, w przeciwieństwie do innych tego typu serwerów.

INN może być wykorzystywany:

- jako serwer grup dyskusyjnych
- do komunikacji grupowej
- do replikacji grup dyskusyjnych i list mailowych

3. OMÓWIENIE SERWERA INN

3.1. BUDOWA I DZIAŁANIE INN

Rdzeniem INN-a jest demon *innd*. Jego zadaniem jest obsługa wszystkich przychodzących artykułów, zachowywanie ich lokalnie i dalsze przekazywanie, o ile jest taka potrzeba. Jest uruchamiany w czasie inicjacji systemu i działa jako proces w tle.

Działanie w trybie demona jest wydajniejsze, ponieważ pliki stanu są czytane tylko raz, przy uruchomieniu. W zależności od wielkości obsługiwanych grup, pewne pliki, takie jak *history* (zawierający listę ostatnio przetworzonych artykułów), mogą zajmować od kilku do kilkudziesięciu megabajtów. Inną ważną funkcją INN-a jest to, że zawsze działa tylko jedno jego wcielenie. Ma to także duży wpływ na wydajność, ponieważ demon może przetwarzać wszystkie artykuły bez martwienia się o synchronizację stanów wewnętrznych z innymi kopiami *innd* dostającymi się do bufora grup w tym samym czasie.

Obecnie do przesyłania artykułów najczęściej służy NNTP, a *innd* bezpośrednio obsługuje tylko ten protokół. Oznacza to, że *innd* oczekuje na gnieździe TCP (port 119) na połączenia i przyjmuje artykuły, używając protokołu *ihave*. Artykuły przybywające inną drogą, niż przez NNTP, są obsługiwane pośrednio przez inny proces przyjmujący artykuły i przekazujący je do *innd* przez NNTP. Wsady przychodzące na przykład przez łącze UUCP są tradycyjnie obsługiwane przez program *rnews*. Wersja tego programu zawarta w pakiecie INN w razie potrzeby dekompresuje wsady i dzieli je na pojedyncze artykuły. Następnie po kolei przesyła je do *innd*. Przeglądarki grup mogą dostarczać wiadomości, gdy użytkownik wyśle artykuł. Przyjmując artykuł, *innd* najpierw sprawdza jego ID w pliku *history*. Zduplikowane artykuły są odrzucane, a ich pojawienie się jest (opcjonalnie) odnotowywane. To samo dotyczy artykułów, które są zbyt stare lub brakuje im wymaganych pól nagłówka, takich jak Subject:*. Jeżeli *innd* stwierdzi, że artykuł jest do przyjęcia, sprawdza wiersz nagłówka Newsgroups:, by stwierdzić, do której grupy został wysłany artykuł. Jeżeli w pliku *active* znajdzie jedną lub więcej grup, artykuł jest zapisywany w postaci pliku na dysku. W przeciwnym razie jest przesyłany do specjalnej grupy *junk*. Pojedyncze artykuły są przechowywane w katalogu */var/spool/news*,

zwanym także *buforem grup*. Każda grupa ma oddzielny katalog, w którym artykuł jest zapisywany jako oddzielny plik. Nazwy plików mają postać kolejnych numerów, a więc na przykład artykuł z grupy *comp.risks* może być zapisany jako *comp/risks/217*. Gdy *innd* stwierdzi, że nie istnieje katalog, w którym trzeba zapisać artykuł, automatycznie go tworzy.

Przekazywaniem artykułów dalej, jako dane wychodzące, zarządza plik *newsfeeds*, który opisuje wszelkie ośrodki, do których powinny być wysyłane artykuły z danej grupy. Podobnie jak po stronie odbiorczej *innd*, tak i po stronie wychodzącej, przetwarzanie jest obsługiwane także przez jeden interfejs. Zamiast samodzielnie obsługiwać wszelkie specyficzne sposoby transportu, *innd* opiera się na różnych ukrytych systemach zarządzających przesyłaniem artykułów do innych serwerów grup. Grupy wychodzące są obsługiwane przez *kanały*. W zależności od przeznaczenia kanał może mieć różne atrybuty, określające dokładnie, jakie informacje przekazuje do niego *innd*. W przypadku danych wychodzących przez NNTP, *innd* mógłby przy uruchamianiu wywołać program *innxmit* i przekazywać mu na standardowe wejście ID, rozmiar i nazwę pliku każdego artykułu, który powinien być wysłany dalej. Natomiast w przypadku danych wychodzących przez UUCP, mógłby zapisywać rozmiar artykułu i jego nazwę pliku do specjalnego pliku log, który byłby sprawdzany w regularnych odstępach czasu przez inny proces, który tworzyłby wsady i kolejkował je w podsystemie UUCP. [1]

Poza tymi dwoma przykładami, istnieją inne typy kanałów, które niekoniecznie dotyczą danych wychodzących. Są one używane na przykład przy archiwizowaniu pewnych grup lub przy generowaniu informacji przeglądowych. Informacje takie mają pomagać przeglądarkom efektywniej dzielić artykuły na wątki. Przeglądarki starego typu muszą przeglądać kolejno wszystkie artykuły, by uzyskać z nagłówka informacje wymagane do podziału na wątki. Obciąża to poważnie serwer, szczególnie jeżeli używa się NNTP. Co więcej jest to bardzo wolne. Mechanizm informacji poglądowych łagodzi ten problem, ponieważ zapisuje wstępnie wszystkie istotne nagłówki każdej grupy w oddzielnym pliku (*.overview*). Później przeglądarka może pobrać te informacje albo bezpośrednio ją odczytując z katalogu bufora, albo wykonując polecenie *XOVER* przy połączeniu przez NNTP. Demon *innd* przekazuje wszystkie artykuły poleceniu *overchan*, które jest połączone z demonem przez kanał. [1]

3.2 INSTALACJA I PODSTAWOWA KONFIGURACJA INN

Dystrybucje Linuksa obecnie zawierają skompilowane pliki binarne wersji 2. INN-a (lub nowszych). Kompilacja INN-a wymaga edycji pliku konfiguracyjnego, który przekazuje INN-owi pewne szczegóły na temat systemu operacyjnego i pewnych funkcji, które mogą wymagać niewielkich modyfikacji. Kompilacja samego pakietu jest prosta. Zawiera on bowiem skrypt *BUILD*, który przeprowadzi użytkownika przez cały proces. Kod Źródłowy zawiera także szczegółową dokumentację, mówiącą, jak zainstalować i skonfigurować INN-a. Po zainstalowaniu wszystkich plików binarnych, mogą być potrzebne pewne ręczne poprawki zapewniające kompatybilność INN-a z różnymi innymi aplikacjami, które mogą wymagać dostępu do programów *rnews* lub *inews*. Na przykład UUCP spodziewa się programu *rnews* w katalogu */usr/bin* lub */bin*, natomiast INN instaluje go domyślnie w */usr/lib/bin*. Należy sprawdzić czy */usr/lib/bin/* jest w domyślnej ścieżce przeszukiwań lub czy istnieje dowiązanie symboliczne wskazujące na rzeczywistą lokalizację poleceń *rnews* i *inews*. [3]

Jedną z największych trudności, na jaką może natrafić początkujący, jest to, że INN do poprawnego funkcjonowania wymaga działającej konfiguracji sieciowej, nawet gdy operuje na samodzielnym hoście. Dlatego trzeba dopilnować dwóch spraw. Po pierwsze, jądro Linuksa musi obsługiwać sieć TCP/IP, gdy chcemy uruchamiać INN-a. Po drugie, wymagany jest

skonfigurowany interfejs pętli zwrotnej. Następnie trzeba sprawdzić, czy *inn*d jest uruchamiany w czasie inicjacji komputera.

Domyślna instalacja INN-a zawiera skrypt o nazwie *boot* w katalogu */etc/news/*. Jeżeli dystrybucja używa pakietu *init* typu System V, wystarczy stworzyć dowiązanie symboliczne do pliku */etc/init.d/inn* tak, by wskazywało na */etc/news/boot*. W innych wersjach *init* należy sprawdzić, czy */etc/news/boot* jest uruchamiany z jednego z skryptów *rc*. Ponieważ INN wymaga sieci, skrypt startowy powinien być uruchamiany po skonfigurowaniu interfejsów sieciowych.

3.3 PLIKI KONFIGURACYJNE INN

Wszystkie pliki konfiguracyjne znajdują się w katalogu */etc/news*. W kilku kolejnych podpunktach omówimy kolejno te pliki.

Więcej na temat funkcji poszczególnych plików konfiguracyjnych, można także przeczytać na stronach podręcznika elektronicznego im poświęconych, zawartych w dystrybucji INN-a.

3.3.1 PLIK *INN.CONF*- PARAMETRY GLOBALNE

Głównym plikiem konfiguracyjnym INN-a jest *inn.conf*. Między innymi określa on nazwę, pod jaką maszyna jest znana w sieci Usenet. Wersja 2 INN-a pozwala skonfigurować w tym pliku wiele parametrów. Na szczęście większość ma wartości domyślne, które są sensowne dla prawie wszystkich ośrodków.

Plik *inn.conf* opisuje szczegółowo wszystkie parametry i należy go dokładnie przeczytać, jeżeli napotka się na jakiegokolwiek problemy. Przykładowy plik *inn.conf* przedstawiony jest w części praktycznej pracy.

3.3.2 PLIKI *ACTIVE* I *NEWSGROUPS* – KONFIGUROWANIE GRUP DYSKUSYJNYCH

Pliki *active* i *newsgroups* są używane do przechowywania i opisywania grup dyskusyjnych obsługiwanych przez dany serwer. Zawierają spis grup, które chcemy otrzymywać i do których chcemy wysyłać artykuły, oraz dotyczących ich informacji administracyjnych. Pliki te znajdują się w katalogu */var/lib/news/*.

Plik *active* określa, które grupy obsługuje serwer. Jego składnia jest prosta. Każdy wiersz pliku *active* składa się z czterech pól oddzielonych białymi znakami:

```
nazwa zngór zndol znaczniki
```

Pole *nazwa* to nazwa grupy. Pole *zngór* zawiera najwyższy numer artykułu w grupie. Pole *zndol* zawiera najniższy numer aktywnego artykułu w grupie. Aby pokazać, jak to działa, można rozważyć następujący scenariusz: mamy nowo utworzoną grupę dyskusyjną: i *zngór*, i *zndol* mają wartość 0, ponieważ w grupie nie ma artykułów. Jeśli wyślemy 5 artykułów, zostaną one ponumerowane od 1 do 5. *zngór* będzie teraz miał wartość 5, czyli najwyższy numer artykułu, a *zndol* będzie równy 1 – numerowi pierwszego artykułu. Jeżeli artykuł 5. zostanie anulowany, nie nastąpi zmiana. *zngór* będzie dalej miał wartość 5, gdyż numery

artykułów nie mogą być relokowane, a *zndol* będzie miał dalej wartość 1. Jeżeli teraz anulujemy artykuł 1, *zngór* pozostanie bez zmian, a *zndol* będzie miał wartość 2, ponieważ 1 nie jest już artykułem aktywnym. Jeżeli teraz wyślemy nowy artykuł, zostanie mu przypisany numer 6, a więc *zngór* będzie teraz miał wartość 6. Artykuł 5 był wykorzystywany, a więc nie zmieniamy jego numeru. Wartość *zndol* pozostaje na poziomie 2. Mechanizm ten pozwala nam prosto alokować unikalne numery dla nowych artykułów i szacować liczbę aktywnych artykułów w grupie: *zngór-zndol*. Ostatnie pole może zawierać jedną z następujących wartości:

y – dopuszczalne jest wysyłanie bezpośrednio do serwera,

n – wysyłanie bezpośrednio do serwera nie jest dopuszczalne,

Zapobiega to wysyłaniu wiadomości przez przeglądarki bezpośrednio do serwera grup.

Nowe artykuły mogą być odbierane tylko z innych serwerów grup.

m – grupa jest moderowana,

Wszelkie artykuły wysłane do tej grupy są przekazywane do jej moderatora w celu zatwierdzenia, zanim pojawią się w grupie. Większość grup nie jest moderowana.

j – artykuły z tej grupy nie są przechowywane, ale jedynie przekazywane dalej,

Powoduje to, że serwer grup przyjmuje artykuł, ale wszystko co z nim robi, to przekazanie dalszym serwerom grup. Artykuły nie są dostępne dla przeglądarek podłączających się do tego serwera w celu czytania grup.

x – artykuły nie mogą być wysyłane do tej grup,

Jedynym sposobem na dostarczenie artykułów do tego serwera jest ich przesłanie z innego serwera grup. Przeglądarki nie mogą bezpośrednio zapisywać artykułów na serwerze.

=*foo.bar* - artykuły są zapisywane lokalnie w grupie „foo.bar”.

Plik *newsgroups* jest jeszcze prostszy. Zawiera jednowierszowy opis każdej grupy. Niektóre przeglądarki mogą odczytywać i przedstawiać zawarte w nim informacje użytkownikowi pomagając mu w ten sposób zdecydować do której grupy się zapisać. Format pliku *newsgroups* jest prosty:

```
nazwa opis
```

Pole *nazwa* to nazwa grupy, a *opis* to wiersz z informacją o treści grupy. [1]

3.3.3 PLIK NEWSFEEDS – KONFIGUROWANIE DOSTARCZANIA GRUP DO INNYCH SERWERÓW.

INN zapewnia administratorowi grup możliwość kontroli nad tym, które grupy i w jaki sposób są przekazywane do innych serwerów. Najpopularniejsze metody wykorzystują opisany wcześniej protokół NNTP, ale INN dopuszcza także inne protokoły, na przykład UUCP.

Plik *newsfeeds* określa, gdzie będą kierowane nowe artykuły. Zwykle znajduje się on w katalogu */etc/news/*. Format pliku *newsfeeds* początkowo wydaje się nieco skomplikowany. Opisany tu zostanie jego ogólny wygląd, szczegóły można znaleźć na stronie podręcznika elektronicznego *newsfeeds*.

Format jest następujący:

```
# format pliku newsfeeds
ośrodek:wzorzec:znaczniki:parametry
```

```
ośrodek2:wzorzec2\  
:znaczniki2:parametry2
```

Każde połączenie związane z przesyłaniem grup do ośrodka jest opisane w jednym wierszu lub w kilku (wtedy na końcu kontynuowanego wiersza trzeba umieścić znak kontynuacji \).

Znak: rozdziela pola. Znak # na początku wiersza oznacza komentarz.

Pole *ośrodek* zawiera nazwę ośrodka, dla którego jest przeznaczona dana porcja grup. Nazwy mogą być zapisywane w dowolny sposób i nie musi być to nazwa domenowa. Nazwa ta zostanie użyta później do wskazania wpisu w tablicy z nazwą hosta, która jest potrzebna programowi *innxmit* wysyłającemu artykuły przez NNTP do serwera zdalnego. Można mieć po kilka wpisów dla każdego ośrodka. Każdy wpis będzie rozpatrywany indywidualnie. Pole *wzorzec* określa, które grupy mają być wysyłane do danego ośrodka. Domyślnie wysyłane są wszystkie grupy, a więc jeżeli to właśnie chce się osiągnąć, po prostu należy pozostawić pole puste. Zwykle pole to zawiera oddzielną przecinkami listę wyrażeń wzorców. Do znaku * pasuje zero lub więcej dowolnych znaków, znak (.) nie ma szczególnego znaczenia, znak ! (jeżeli został użyty na początku wyrażenia) realizuje logiczną operację NOT, a znak @ na początku nazwy grupy oznacza „Nie przekazuje żadnych artykułów, które zostały wysłane do tej grupy”. Lista jest odczytywana i analizowana od lewej do prawej strony, a więc należy na początku umieszczać dokładniejsze reguły. Np. wzorzec:

```
sad.drzewo.jablon* , !sad.drzewo.jablon.typ , @sad.drzewo.jablon.ma
```

spowoduje wysłanie wszystkich grup z hierarchii *sad.drzewo.jablon* z wyjątkiem grupy

sad.drzewo.jablon.typ. Nie zostaną przekazane żadne artykuły wysłane do grupy

sad.drzewo.jablon.ma. Zostaną zatrzymane i będą dostępne tylko dla ludzi z tego serwera.

Gdyby zamienić miejscami dwa pierwsze wzorce, pierwszy zostałby nadpisany przez drugi i skończyłoby się to przekazaniem wszystkich artykułów z grupy *sad.drzewo.jablon.typ*. To samo dotyczy pierwszego i ostatniego wzorca. Zawsze musi się umieszczać dokładniejsze wzorce przed mniej dokładnymi, aby spełniały swoje funkcje.

Pole *znaczniki* kontroluje przekazywanie artykułów do danego ośrodka i nakłada ograniczenia.

Pole to jest oddzielną przecinkami listą zawierającą niżej wymienione elementy:

<*rozmiar* – artykuł musi mieć mniej bajtów niż zadany rozmiar,

Aelementy – sprawdzanie artykułów. *Elementy* mogą być mieć wartość jednego lub kilku d (musi mieć nagłówek Distribution) lub p (nie sprawdzaj nagłówka Path dla tego ośrodka),

Bwys/nis – rozmiar bufora wewnętrznego przed zapisaniem na wyjście,

H/liczba/ – artykuł musi mieć mniej niż *liczba* hopów – domyślnie 1,

Irozmiar – rozmiar bufora wewnętrznego (do przesyłania plików),

Mwzorzec – do tego wzorca pasują tylko grupy moderowane,

Nwzorzec – do tego wzorca pasują tylko grupy niemoderowane,

Srozmiar – rozpoczęcie buforowania, jeżeli zostanie zakolejkowane więcej bajtów niż zadany rozmiar

Ttyp – typy przekazywania: f (plik), m (strumień; pole *parametry* musi zawierać nazwy wpisów, do których artykuły będą przekazywane), p (przekazywanie przez potok do programu), c (wysyłanie do kanału stdin podprocesu pola *parametry*) i x (jak c, ale obsługuje polecenia na stdin).

Welementy– co zapisać: b (rozmiar artykułu w bajtach), f (pełna ścieżka), g (pierwsza grupa dyskusyjna), m (ID wiadomości), n (ścieżka względna), s (ośrodek, z którego przyszedł artykuł), t (czas odebrania), * (nazwy strumieni wejściowych lub wszystkich ośrodków, które mają artykuł), N (nagłówek grupy dyskusyjnej), D

(nagłówki dystrybucji), H (wszystkie nagłówki), o (dane poglądowe) i R (dane replikacyjne).

Pole parametry jest specjalnie kodowane w zależności od sposobu dostarczania grup innym serwerom. W większości popularnych konfiguracji podaje się nazwę pliku wynikowego, do którego będą zapisywane wychodzące artykuły. W innych można go nie podawać. W jeszcze innych konfiguracjach ma ono różne znaczenia.

Istnieje szczególna nazwa ośrodka, kodowana jako ME, którą powinien zawierać pierwszy wpis w pliku. Wpis ten jest używany do kontrolowania domyślnych ustawień dostarczania grup. Jeżeli wpis ME posiada związaną z dostarczaniem listę dystrybucji, będzie ona doklejana do każdego wpisu zawierającego ośrodek przed wysłaniem do niego grup. Pozwala to na przykład na automatyczne przekazywanie pewnych grup lub automatyczne blokowanie przekazania innych bez potrzeby powtarzania wzorca w każdym opisie ośrodka. Możliwe jest użycie pewnych połączeń specjalnych do wygenerowania wątku danych, który ułatwia pracę przeglądarki. Wątek danych jest generowany za pomocą polecenia *overchan* będącego częścią dystrybucji INN. Wcześniej jednak trzeba stworzyć specjalną lokalną porcję o nazwie *overview*, przekazującą artykuły do polecenia *overchan* w celu przetworzenia na dane poglądowe. [1]

3.3.4 PLIK *NNTPSEND.CTL* – KONFIGURACJA PROGRAMU *NNTPSEND*.

Program *nntpsend* zarządza przesyłaniem artykułów przez NNTP, wywołując polecenie *innxmit*. Polecenie *nntpsend* spodziewa się plików wsadowych dla ośrodków, do których ma wysłać grupy. Oczekuje, że będą one miały nazwy postaci:

/var/spool/news/out.going/nazwaośrodka.innd tworzy te pliki wsadowe na podstawie wpisu w pliku *newsfeeds*, który był już przedstawiony. W polu *parametry* określiliśmy nazwę ośrodka jako nazwę pliku i tym samym spełniliśmy wymagania co do danych wejściowych dla polecenia *nntpsend*. Polecenie *nntpsend* ma plik konfiguracyjny o nazwie *nntpsend.ctl*, który zwykle znajduje się w katalogu */etc/news/*. Plik *nntpsend.ctl* pozwala nam związać pełną nazwę domenową, ograniczenia rozmiaru przesyłanej porcji artykułów i ograniczenia parametrów transmisji z nazwą ośrodka. Nazwa ta ma unikalnie identyfikować logiczną, wysłaną porcję artykułów. [1] Ogólny format pliku jest następujący:

```
nazwaośrodka:fqdn:max_rozmiar:[argumenty]
```

Poniższa lista opisuje poszczególne elementy tego wiersza:

nazwaośrodka – nazwa ośrodka zgodnie z podaną w pliku *newsfeeds*,
fqdn – pełna nazwa domenowa serwera grup, do którego przesyłamy artykuły,
max_rozmiarm – maksymalna wielkość porcji artykułów wysyłanych za jednym razem,
argumenty – dodatkowe argumenty przekazywane do polecenia *innxmit*.

3.3.5 PLIK *INCOMING.CONF* – KONTROLOWANIE DOSTĘPU PRZEGLĄDARKI.

Obecnie obserwuje się tendencje do likwidowania publicznych serwerów grup dyskusyjnych.. Większość organizacji skrupulatnie nadzoruje dostęp do swoich serwerów, zwłaszcza ogranicza dostęp użytkownikom swojej sieci. INN posiada pliki konfiguracyjne pozwalające na kontrolę dostępu.

W pliku */etc/news/incoming.conf* umieszcza się hosty, które będą dostarczały grupy naszemu ośrodkowi przez protokół NNTP, oraz pewne parametry sterujące sposobem, w jaki host pobiera z nich artykuły. Wszelkie hosty nie wpisane w tym pliku, a łączące się z gniazdem news nie będą obsługiwane przez demona *innd*, ale przez *nnrpd*. Składnia pliku */etc/news/incoming.conf* jest bardzo prosta, ale jej zrozumienie może zająć chwilę. Dopuszczalne są trzy typy wpisów: para klucz/wartość, która określa sposób podawania atrybutów i ich wartości parametry równorzędne, które określają sposób podawania nazw hosta, który może wysyłać do nas artykuły przez NNTP oraz grupy, których dotyczą poprzednie wartości. Pary klucz/wartość mogą mieć trzy różne zakresy. Pary globalne dotyczą każdego elementu zdefiniowanego w pliku, grupy par dotyczą wszystkich elementów zdefiniowanych w danej grupie, a pary równoważne dotyczą tylko danego konkretnego przypadku. Definicje bardziej szczegółowe unieważniają te mniej szczegółowe i dlatego definicje równoważne unieważniają definicje grup, które z kolei unieważniają pary globalne. Nawiasy klamrowe ({}) są używane do oznaczenia początku i końca definicji group i peer. Znak # oznacza, że dalszy ciąg wiersza to komentarz. Pary klucz/wartość są oddzielane dwukropkiem i są wpisywane w wierszu pojedynczo. [1]

Można podać szereg różnych kluczy. Najczęściej używane i najbardziej przydatne z nich to:

- hostname* – ten klucz określa, oddzielaną przecinkami, listę pełnych nazw domenowych lub adresów IP hostów równorzędnych, które mogą wysyłać nam artykuły. Jeżeli ten klucz nie zostanie podany, przyjmowana jest domyślna nazwa hosta partnerskiego.
- streaming* – określa, czy dla danego hosta są dopuszczalne polecenia strumieniowe. Jest to wartość boole'owska, domyślnie – true.
- max-connections* – określa maksymalną liczbę połączeń dopuszczalnych z danej grupy lub z hostów równoważnych. Wartość zero oznacza nieograniczoną ich liczbę (można także podać none).
- password* – ten klucz pozwala ci określić hasło, które musi być używane przez partnera, jeżeli ma on prawo przysyłać wiadomości. Domyślnie hasło nie jest wymagane.
- pattern* – ten klucz określa grupy, które przyjmujemy od partnera. Pole to jest kodowane zgodnie z tymi samymi regułami, których używaliśmy w pliku *newsfeeds*.

3.3.6 PLIK *NNRP.ACCESS* – KONFIGUROWANIE PROGRAMU *NNRPD*.

Program *nnrpd* używa pliku */etc/news/nnrp.access* do określenia, kto ma prawo korzystać z serwera grup i jakie powinien mieć prawa dostępu.

Plik *nnrp.access* ma budowę podobną do innych plików konfiguracyjnych, które omawialiśmy do tej pory. Składa się z zestawu wzorców używanych do dopasowywania nazw lub adresów IP łączących się hostów i pól, które określają, jakie prawa dostępu powinny być im dane. Każdy wpis powinien znajdować się w oddzielnym wierszu, a pola powinny być oddzielone dwukropkami. Jak zwykle używany będzie ostatni wpis w pliku pasujący do podłączającego się hosta, a więc powinieneś umieszczać wzorce ogólne na początku, a następnie wzorce szczegółowe. [1] Pięć pól w każdym wpisie ma następujące znaczenie :

Nazwa hosta lub adres IP

To pole jest zgodne z regułami dopasowania wzorca *wildmat*. Jest to wzorzec opisujący nazwę hosta lub adres IP podłączającego się hosta.

Prawa dostępu

To pole określa, jakie prawa dostępu powinny być nadane pasującemu hostowi. Istnieją dwa rodzaje praw, które można skonfigurować: R daje prawo czytania, a P daje prawo wysyłania.

Nazwa użytkownika

To pole jest opcjonalne i pozwala określić nazwę użytkownika, na którego musi się zalogować klient NNTP, zanim będzie mógł wysyłać artykuły. Pole to można pozostawić puste. Do czytania artykułów nie jest potrzebna żadna autoryzacja.

Hasło

To pole jest opcjonalne i zawiera hasło towarzyszące polu *nazwa użytkownika*. Pozostawienie tego pola pustego oznacza, że do wysyłania artykułów nie jest wymagane hasło.

Grupy

To pole jest wzorcem określającym, do których grup klient ma dostęp. Wzorzec podlega tym samym regułom, jakie były używane w pliku *newsfeeds*. Domyślną wartością tego pola jest brak grup, a więc zwykle podajesz tu jakiś wzorzec.

3.3.7 PLIK *EXPIRE.CTL* – WYGASANIE ARTYKUŁÓW W GRUPACH.

Artykuły odbierane przez serwer grup są zapisywane na dysk. Aby to miało sens, artykuły muszą być dostępne dla użytkowników przez jakiś okres czasu, a więc duży serwer grup może zajmować wiele miejsca na dysku. Aby miejsce to było wykorzystywane efektywnie, można automatycznie usuwać artykuły po zadanym okresie czasu. Nazywa się to *wygaśnięciem artykułu*. Oczywiście INN obsługuje automatyczne wygasanie artykułów.

Do usuwania starych artykułów serwer INN używa programu *expire*. Program ten z kolei wykorzystuje plik */etc/news/expire.ctl*, w którym są skonfigurowane reguły zarządzające wygasaniem artykułów. Składnia pliku */etc/news/expire.ctl* jest dość prosta. Podobnie jak w większości plików konfiguracyjnych, puste wiersze lub wiersze rozpoczynające się znakiem # są ignorowane. Ogólna zasada jest taka, że każdą regułę pisze się w oddzielnym wierszu. Każda reguła definiuje, jak jest realizowane wygasanie artykułu w grupach zgodnych z zadanym wzorcem. [1] Składnia reguły wygląda tak:

```
wzorzec:znmod:trzymanie:domyślnie:czyszczenie
```

Poniższa lista opisuje poszczególne pola reguły:

wzorzec

To pole jest oddzielną przecinkami listą wzorców dopasowujących nazwy grup. Do ich sprawdzania jest używana procedura *wildmat*. Stosowana jest ostatnia z pasujących reguł, a więc gdybyśmy umieszczali reguły zestawień uniwersalnych(*), powinny być w pliku jako pierwsze.

znmod

Ten znacznik opisuje, jak reguła jest stosowana do grup moderowanych. Może on być zapisany jako M, co oznacza, że reguła dotyczy tylko grup moderowanych, albo jako U, co oznacza, że reguła dotyczy tylko grup niemoderowanych. Można też użyć A, aby wskazać, że reguła ignoruje status moderowania i dotyczy wszystkich grup.

trzymanie

To pole pozwala określić minimalny czas, przez jaki będzie przechowywany artykuł z ustawionym polem Expires w nagłówku, zanim wygaśnie. Wartość jest określana w dniach, ale dopuszczalne są liczby zmiennoprzecinkowe, a więc możesz podać wartość 7.5, która oznacza siedem i pół dnia. Można także podać never, aby artykuł pozostał w grupie na zawsze.

domyślnie

To pole jest najważniejsze. Pozwala określić czas, przez jaki będzie przechowywany artykuł bez pola nagłówka Expires. Większość artykułów nie będzie miała takiego pola w nagłówku. Pole *domyślnie* jest kodowane dokładnie w ten sam sposób jak pole *trzymanie*. never oznacza, że artykuł bez nagłówka Expires nigdy nie wygaśnie.

czyszczenie

To pole pozwala zadać maksymalny czas, przez jaki będzie przechowywany artykuł z polem Expires, zanim wygaśnie. Kodowanie tego pola przebiega tak samo jak pola *trzymanie*.

Istnieje jeszcze jeden specjalny rodzaj wpisu, który może się znaleźć w pliku */etc/news/expire.ctl*. Może mieć dokładnie jeden wiersz wyglądający tak:

```
/remember/:dni
```

Pozwala on zadać minimalną liczbę dni, przez jaką artykuł będzie pamiętany w pliku historii, bez względu na to, czy sam artykuł wygasł, czy nie. Może to być przydatne, - jeżeli jeden z ośrodków dostarczających nam artykuły nie robi tego często i ma tendencję do przysyłania starych artykułów. Ustawienie pola */remember/* pomaga zapobiec ponownemu przysyłaniu artykułu, który u nas już wygasł. Jeżeli serwer pamięta, że już otrzymał kiedyś taki artykuł, odrzuci próbę ponownego jego przysłania. Trzeba pamiętać, że to ustawienie nie ma żadnego wpływu na wygasanie artykułu. Dotyczy jedynie czasu przechowywania informacji o artykule w pliku historii.

3.3.8 PLIK CONTROL.CTL – OBSŁUGIWANIE WIADOMOŚCI KONTROLNYCH.

INN ma doskonały mechanizm konfiguracyjny nadzorujący działania, jakie są podejmowane dla każdej z wiadomości kontrolnych, oraz mechanizm kontroli dostępu, który czuwa nad tym, kto zainicjował działanie i wobec jakich grup.

Budowa pliku *control.ctl* jest dosyć prosta. Reguły składniowe są w zasadzie takie same jak w przypadku innych plików konfiguracyjnych INN-a. Wiersze rozpoczynające się od znaku # są ignorowane; wiersze można kontynuować używając znaku /, a pola są oddzielane dwukropkami.

Gdy zostanie odebrana wiadomość kontrolna, jest sprawdzana po kolei z każdą regułą. Stosowana jest jak zwykle ostatnia pasująca reguła w pliku, a więc powinno umieszczać się ogólne reguły na początku pliku, a dokładniejsze pod jego koniec. [1] Ogólna składnia pliku jest następująca:

wiadomość : skąd : grupa : działanie

Znaczenie poszczególnych pól jest następujące:

wiadomość

Jest to nazwa wiadomości kontrolnej. Typowe wiadomości kontrolne opisujemy dalej.

skąd

Jest to wzorzec opisujący adres e-mail osoby wysyłającej wiadomość. Przed porównaniem adres jest konwertowany do małych liter.

grupa

Jeżeli wiadomość kontrolna to *newgroup* lub *rmgroup*, to pole ma postać wzorca pasującego do tworzonej lub usuwanej grupy.

działanie

To pole określa, jakie działanie podjąć, gdy wiadomość pasuje do reguły. Możliwe są różne działania, opisane na następnej liście.

Pole *wiadomość* w każdym wierszu może przyjmować następujące wartości:

checkgroups

Ta wiadomość stanowi żądanie wobec administratora grup, by zsynchronizował swoją bazę aktywnych grup z listą grup dostarczoną w wiadomości kontrolnej.

newgroup

Ta wiadomość stanowi żądanie utworzenia nowej grupy. Treść wiadomości powinna zawierać krótki opis przeznaczenia tworzonej grupy.

rmgroup

Ta wiadomość stanowi żądanie usunięcia grupy.

sendsys

Ta wiadomość stanowi żądanie przesłania pocztą pliku *sys* z serwera grup do nadawcy wiadomości. RFC-1036 mówi, że warunkiem członkostwa w Usenecie jest publiczne udostępnianie tej wiadomości, ponieważ jest ona wykorzystywana przy uaktualnianiu map usenetowych.

version

Ta wiadomość stanowi żądanie zwrotu do nadawcy tej wiadomości nazwy hosta i wersji oprogramowania serwera grup.

all

Jest to specjalny zapis, do którego pasują wszystkie wiadomości kontrolne. Pole *wiadomość* może zawierać następujące działania:

doit

Żądane polecenie jest wykonywane. W wielu przypadkach do administratora jest wysyłana wiadomość e-mail z informacją, że dane działanie miało miejsce.

doit=plik

Jest to działanie identyczne z *doit*, ale do pliku log *plik* zostanie zapisany komunikat. Jeżeli podanym plikiem jest *mail*, wpis log jest wysyłany pocztą. Jeżeli podanym plikiem jest ciąg pusty, wiadomość log jest zapisywana do */dev/null* i jest to równoważne z użyciem czystego polecenia *doit*. Jeżeli nazwa *plik* rozpoczyna się od znaku */*, jest uznawana za bezwzględną ścieżkę do pliku log. W przeciwnym razie zadana nazwa jest zamieniana do postaci */var/log/news/file.log*.

doifarg

Żądane polecenie jest wykonywane, jeżeli ma argument. Jeżeli nie ma argumentu, wiadomość kontrolna jest ignorowana.

drop

Żądane polecenie jest ignorowane.

log

Wiadomość log jest wysyłana na standardowe wyjście błędów *stderr* procesu *innnd*. Normalnie jest przekierowywana do pliku */var/log/news/errlog*.

log=plik

To samo co *log*, ale plik log jest zadawany na tych samych zasadach co w przypadku działania *doit=plik*.

mail

Do administratora jest wysyłana wiadomość e-mail zawierająca żądane szczegóły. Żadne inne działanie nie jest podejmowane.

verify-

Jeżeli działanie rozpoczyna się od ciągu "verify-", wiadomość kontrolna jest uwierzytelniana przez PGP (lub GPG).

4. ZARZĄDZANIE INN – POLECENIE *CTLINND*.

Serwer grup INN zawiera polecenie do zarządzania jego codziennym działaniem. Polecenie *ctlinnd* może być używane do operowania na grupach i porcjach grup wysyłanych do innych serwerów, uzyskiwania stanu serwera oraz do przeładowywania, zatrzymywania i uruchamiania serwera.

Podsumowanie działania polecenia *ctlinnd* można uzyskać, używając polecenia następująco:

```
# ctlinnd -h
```

Poniżej omówione są ważniejsze zastosowania *ctlinnd*.

4.1. DODAWANIE NOWEJ GRUPY

Aby dodać nową grupę, należy użyć polecenia następująco:

```
ctlinnd newgroup grupa znacz twórca
```

Argumenty mają następujące znaczenie:

grupa - nazwa tworzonej grupy,
znacz - ten argument powinien być zapisywany w ten sam sposób jak pole *znaczniki* pliku *active*. Domyślnie jest przyjmowana wartość y, jeżeli nic innego nie zostanie podane,
twórca - nazwa osoby tworzącej grupę. Umieść ją w cudzysłowie, jeżeli chcesz wpisać jakieś spacje.

4.2. ZMIANA GRUPY

Aby zmienić grupę, należy użyć polecenia następująco:

```
ctlinnd changegroup grupa znacz
```

Argumenty mają następujące znaczenie:

grupa - nazwa zmienianej grupy,
znacz - ten argument powinien być zapisywany w ten sam sposób co pole *znaczniki* pliku *active*. To polecenie przydaje się przy zmianie statusu moderowania grupy.

4.3. USUWANIE GRUPY

Aby usunąć grupę, należy użyć polecenia :

```
ctlinnd rmgroup grupa
```

grupa - nazwa usuwanej grupy.

To polecenie usuwa zadaną grupę z pliku *active*. Nie ma wpływu na katalog bufora.

Wszystkie zawarte w nim artykuły wygasną w zwykły sposób, a nowe nie będą przyjmowane.

4.4. PRZENUMEROWANIE GRUPY

Aby przenieść grupę, należy użyć polecenia *ctlinnd* następująco:

```
ctlinnd renumber grupa
```

Argument *grupa* to nazwa przenumerowanej grupy. Jeżeli *grupa* jest pustym ciągiem znaków, przenumerowywane są wszystkie grupy. To polecenie uaktualnia dolny znacznik w zadanej grupie.

4.5. UŻYWANIE SERWERA PRZEZ PRZEGLĄDARKI GRUP- POZWALANIE I ZABRANIANIE

Użycie poniższego polecenia, pozwoli lub zabroni przeglądarkom korzystać z serwera:

```
ctlinnd readers znacz tekst
```

Argumenty mają następujące znaczenie:

znacz

Jeśli jest ustawiony na *n*, zabrania podłączać się przeglądarkom grup. Podanie *y* pozwala na przyjmowanie połączeń od przeglądarek.

tekst

Podany tekst jest przekazywany do przeglądarki, która próbuje się podłączyć i zwykle opisuje powód zabronienia dostępu. Przy ponownym włączaniu dostępu przeglądarkom, pole to musi być pustym ciągiem lub kopią tekstu podanego przy zakazie dostępu. Polecenie to nie wpływa na przychodzące z innych serwerów porcje grup. Kontroluje jedynie dostęp przeglądarek.

4.6. ODMOWA POŁĄCZENIA OD INNEGO SERWERA

Aby odmówić połączenia innemu serwerowi, użyj polecenia następująco:

```
ctlinnd reject powód
```

Argument *powód* ma następujące znaczenie - podany tekst powinien wyjaśniać, dlaczego przychodzące do *innd* połączenia są odrzucane. To polecenie nie ma wpływu na połączenia obsługiwane przez *nnrpd* (tzn. przeglądarki grup). Dotyczy jedynie połączeń, które są obsługiwane bezpośrednio przez *innd*, czyli połączeń z innych serwerów.

4.7. POZWOLENIE NA POŁĄCZENIE OD INNEGO SERWERA

Aby pozwolić na połączenia innemu serwerowi, należy polecenie *ctlinnd* wykonać z następującymi argumentami:

```
ctlinnd allow powód
```

Argument *powód* - podany tekst musi być identyczny z podanym w poleceniu *reject* lub musi być to ciąg pusty. To polecenie odwraca działanie polecenia *reject*.

4.8. ZAMKNIĘCIE SERWERA GRUP

Aby zamknąć serwer grup, należy użyć polecenia następująco:

```
ctlinnd throttle powód
```

Argument *powód* jest to powód zamknięcia serwera. To polecenie jest równoważne z jednoczesnym wydaniem polecenia `newsreaders no i reject`. Jest przydatne przy pracach awaryjnych wykonywanych na bazie serwera grup. Daje pewność, że nikt nie będzie próbował go uaktualnić, gdy przy nim pracujesz.

4.9. RESTART SERWERA GRUP

Aby zrestartować serwer grup, należy użyć polecenia następująco:

```
ctlinnd go powód
```

Argument *powód* podaje powód zatrzymania serwera. Jeżeli pole to jest pustym ciągiem znaków, serwer zostanie bezwarunkowo ponownie włączony. Jeżeli *powód* został podany, to tylko funkcje, które są wyłączone z powodu zgodnego z podanym tekstem, zostaną ponownie uruchomione. To polecenie jest używane do ponownego uruchamiania serwera po wykonaniu poleceń *throttle*, *pause* lub *reject*.

4.10. WYŚWIETLANIE STATUSU POBIERANIA PLIKÓW Z INNEGO SERWERA

Aby wyświetlić status pobierania plików z innego serwera, wydajemy polecenie:

```
ctlinnd feedinfo ośrodek
```

Argument *ośrodek* to nazwa ośrodka (wzięta z pliku *newsfeeds*), dla którego wyświetlamy status.

4.11. ODŁĄCZENIE DOSTARCZANIA PLIKÓW OD INNEGO SERWERA

Aby wyłączyć dostarczanie plików z innego serwera, wykonujemy polecenie:

```
ctlinnd drop ośrodek
```

Argument *ośrodek* to nazwa ośrodka (wzięta z pliku *newsfeeds*), dla którego dostarczanie plików jest wyłączane. Jeżeli pole jest pustym ciągiem, wszystkie aktywne transmisje zostaną przerwane. Wyłączenie dostarczania plików zatrzymuje wszelkie aktywne transmisje do danego ośrodka. Nie jest to zmiana stała. Polecenie jest przydatne, jeżeli modyfikuje się szczegóły związane z przesyłaniem plików z danego ośrodka, a transmisja jest akurat aktywna.

4.12. ROZPOCZYNIANIE DOSTARCZANIA PLIKÓW OD INNEGO SERWERA

Aby rozpocząć dostarczanie plików z innego serwera, użyj polecenia następująco:

```
ctlinnd begin ośrodek
```

Argument ośrodek - nazwa ośrodka wzięta z pliku *newsfeeds*, z którego rozpoczęto przesyłanie plików. Jeżeli przesyłanie z tego ośrodka jest już aktywne, automatycznie jest wykonywane polecenie *drop*. To polecenie powoduje, że serwer czyta ponownie plik *newsfeeds*, znajduje pasujący wpis i rozpoczyna przesyłanie plików do/z danego ośrodka zgodnie ze znalezionymi szczegółami. Można użyć tego polecenia do przetestowania nowych wpisów po ich dodaniu lub modyfikacji w pliku *newsfeeds*.

4.13. ANULOWANIE ARTYKUŁU

Aby anulować artykuł, używa się polecenia *ctlinnd* następująco:

```
ctlinnd cancel ID-artykułu
```

Argument *ID-artykułu* to ID anulowanego artykułu. To polecenie powoduje, że zadany artykuł zostanie usunięty z serwera. Nie generuje wiadomości *cancel*.

5. INSTALACJA I KONFIGURACJA INN NA LINUX DEBIAN.

5.1. INFORMACJE WSTĘPNE.

Serwer INN stawiamy na komputerze o adresie *192.168.40.1* i nazwie *senegal.afryka.pl*

Podłączamy do niego komputer o adresie *192.168.40.4*. Instalacja jest prowadzona na systemie Debian 3.0.

5.2. INSTALACJA.

```
apt-get install inn2
```

Pliki po instalacji posiadają konfigurację domyślną - serwer dostępny jest już lokalnie.

5.3. KONFIGURACJA.

5.3.1. ZMIANA USTAWIEŃ GLOBALNYCH- PLIK *INN.CONF*

```
organization:      Nieznana ---
domain:            afryka.pl
server:            senegal.afryka.pl
```

Pozostałe opcje zostawiamy niezmienione.

5.3.2. MODYFIKACJA PLIKU NEWSFEEDS

```
news.nigeria.pl/news.nigeria.pl\  
:*.,!junk*,!test*,!to*,!control*,!news*\  
:Tf,Wnm:
```

gdzie *news.nigeria.pl* jest nazwą serwera grup dyskusyjnych z którym wymieniane będą wiadomości. Następnie modyfikujemy listę postów akceptowanych przez serwer (dodajemy polskie grupy dyskusyjne)

```
ME:!*//!local,!collabra-internal\  
/pl,world,usa,na,gnu,bionet,pubnet,ddn,k12\  
::
```

5.3.3. KONFIGURACJA PLIKU READERS.CONF

Określamy komputery mogące wysyłać i pobierać wiadomości z naszego serwera.

```
auth "localhost" {  
    hosts: "192.168.40.1"  
    default: "<localhost>"  
}
```

```
access "localhost" {  
    users: "<localhost>"  
    newsgroups: "*"   
    access: RPA  
}
```

```
auth "gosc" {  
    hosts: "192.168.40.4"  
    default: "<gosc>"  
}
```

```
access "gosc" {  
    users: "<gosc>"  
    newsgroups: "gosc.*"  
    access: RPA  
}
```

Osobom pracującym lokalnie na serwerze dajemy pełny dostęp do wszystkich grup, komputerowi *192.168.40.4* dajemy pełny dostęp tylko do grup *gosc.**

Aby udostępnić serwer dla wszystkich w polu *host* wpisujemy : *hosts: ="*"*.

Resetujemy serwer: */etc/init.d/inn2 restart*

5.3.4. DODANIE NOWEJ GRUPY.

Dodajemy grupę *gosc.fajna.grupa*:

```
ctlinnd newgroup gosc.fajna.grupa
```

Teraz możemy pobierać i wysyłać informacje z listy.

5.4. PRZYKŁADOWE PLIKI KONFIGURACYJNE Z NASZEGO SERWERA.

5.4.1. PLIK *INN.CONF*

```
## $Id: inn.conf.in,v 1.41.2.1 2000/07/17 06:46:17 rra Exp $
##
## inn.conf -- INN configuration data

mta:                /usr/sbin/sendmail -oi -oem %s
organization:       Nieznana ---
ovmethod:           tradindexed
pathhost:           unknown
pathnews:           /usr/lib/news

# General Settings

domain:             afryka.pl
innflags:
mailcmd:            /usr/lib/news/bin/innmail
server:             senegal.afryka.pl

# Feed Configuration

artcutoff:          10
bindaddress:
hiscachesize:       0
ignorenewsgroups:  false
immediatecancel:   false
linecountfuzz:      0
maxartsize:         1000000
```

```
maxconnections:      50
pathalias:
pgpverify:          false
port:               119
refusecybercancels: false
remembertrash:     true
sourceaddress:
usecontrolchan:     false
verifycancels:     false
wanttrash:         false
wipcheck:           5
wipexpire:          10
```

Article Storage

```
cnfscheckfudgesize: 0
enableoverview:     true
groupbaseexpiry:   true
mergetogroups:     false
overcachesize:     15
ovgrouppat:
storeonxref:        false
useoverchan:        false
wireformat:         false
xrefslave:          false
```

Reading

```
allownewnews:       true
articlemmap:        false
clienttimeout:      600
nnrpdcheckart:     true
nnrpperlauth:       false
nnrppythonauth:    false
noreader:           false
readerswhenstopped: false
readertrack:        false
```

Reading -- Keyword Support

```
keywords:           false
keyartlimit:        100000
keylimit:           512
keymaxwords:        250
```

Posting

```
addnntppostingdate: true
addnntppostinghost: true
checkincludedtext:  false
complaints:
fromhost:
```

```
localmaxartsize:      1000000
moderatormailer:
nnrpdauthsender:     false
nnrpdposthost:
nnrpdpostport:      119
spoolfirst:         false
strippostcc:        false
```

```
# Posting -- Exponential Backoff
```

```
backoffauth:         false
backofffdb:
backofffk:           1
backoffpostfast:    0
backoffpostslow:    1
backofftrigger:     10000
```

```
# Monitoring
```

```
doinnwatch:         true
innwatchbatchspace: 800
innwatchlibspace:   25000
innwatchloload:     1000
innwatchhiload:     2000
innwatchpauseload:  1500
innwatchesleeptime: 600
innwatchspoolnodes: 200
innwatchspoolspace: 8000
```

```
# Logging
```

```
docnfsstat:         false
logartsize:         true
logcancelcomm:      false
logcycles:          3
logipaddr:          true
logsitename:        true
nnrpdoverstats:     false
nntpactsync:        200
nntplinklog:        false
status:             0
timer:              0
```

```
# System Tuning
```

```
badiocount:         5
blockbackoff:       120
chaninacttime:      600
chanretrytime:      300
icdsynccount:       10
maxforks:           10
```

```

nicekids:                4
nicenewnews:            0
nicennrpd:              0
pauseretrytime:        300
peertimeout:           3600
rlimitnofile:          -1

# Paths

patharchive:            /var/spool/news/archive
patharticles:           /var/spool/news/articles
pathbin:                /usr/lib/news/bin
pathcontrol:            /usr/lib/news/bin/control
pathdb:                 /var/lib/news
pathetc:                /etc/news
pathfilter:             /etc/news/filter
pathhttp:               /var/log/news
pathincoming:           /var/spool/news/incoming
pathlog:                /var/log/news
pathoutgoing:           /var/spool/news/outgoing
pathoverview:           /var/spool/news/overview
pathrun:                /var/run/news
pathspool:              /var/spool/news
pathtmp:                /var/spool/news/incoming/tmp

```

5.4.2. PLIK NEWSFEEDS

```

## $Id: newsfeeds.in,v 1.14.2.2 2001/02/07 00:35:15 rra Exp $
##
## newsfeeds - determine where Usenet articles get sent
##
## Format:
##   site[/exclude,exclude...]\
##       :pattern,pattern...[/distrib,distrib...]\
##       :flag,flag...\
##       :param

ME:!*/*!local,!collabra-internal\
/pl,world,usa,na,gnu,bionet,pubnet,ddn,k12\
::

news.nigeria.pl/news.nigeria.pl\
:*,!junk*,!test*,!to*,!control*,!news*\
:Tf,Wnm:

```

5.4.3.PLIK READERS.CONF

```

## $Id: readers.conf,v 1.4.2.1 2001/01/16 15:36:37 rra Exp $
##
## readers.conf - Access control and configuration for nnrpd
##
## Format:
##   auth "<name>" {
##       hosts: "<hostlist>"
##       auth: "<authprog>"
##       res: "<resprog>"
##       default: "<identity>"
##       default-domain: "<email-domain>"
##   }
##   access "<name>" {
##       users: "<userlist>"
##       newsgroups: "<newsgroups>"
##       read: "<read>"
##       post: "<post>"
##       access: "<perm>"
##   }
##
access "localhost" {
    users: "<localhost>"
    newsgroups: "*"
    access: RPA
}

auth "localhost" {
    hosts: "192.168.40.1"
    default: "<localhost>"
}

auth "gosc" {
    hosts: "192.168.40.4"
    default: "<gosc>"
}

access "gosc" {
    users: "<gosc>"
    newsgroups: "gosc.*"
    access: RPA
}

```

6. LITERATURA

- [1] Olaf Kirch, Terry Dawson „Linux – przewodnik administratora”, wydanie drugie.
- [2] Strona domowa The Internet Software Consortium (ISC)
<http://www.isc.org/products/INN/>
- [3] Dokumentacja do INN 2.3 i wyższych
<http://web.inter.nl.net/users/Elena.Samsonova/unix/INN/v2.3/>
- [4] Podręcznik systemowy Linuxa - man