Usługi wysokopoziomowe - interakcyjna praca na odległość

- sieci komputerowe pozwalają na dostęp do wielu maszyn; niektóre są mniej obciążone od innych, na niektórych jest dostępne szczególne oprogramowanie — stąd pomysł pracy na odległość;
- można by w zasadzie dla każdej konkretnej czynności budować osobną parę klient–serwer ze specyficznym protokołem, ale jest to niesłychanie niepraktyczne;
- z tego powodu zaprojektowano ogólne protokoły do pracy na odległość: Telnet, rlogin, ssh.

Protokół TELNET

- jest standardowym protokołem zestawu TCP/IP; umożliwia użytkownikowi jednego węzła nawiązanie sesji terminalowej z innym węzłem za pośrednictwem protokołu opartego na TCP;
- klient odczytuje klawisze naciskane na maszynie lokalnej i przekazuje je maszynie docelowej, a sam odbiera wypisywane przez nią znaki i wyświetla je lokalnie; daje to złudzenie bezpośrednio przyłączonego terminala;
- usługi definiowane przez protokół TELNET:
  - definiowanie **sieciowego terminala wirtualnego**;
  - mechanizm negocjacji opcji między klientem i serwerem;
  - symetryczne traktowanie obu końców połączenia.

Współpraca implementacji TELNET–u z systemem operacyjnym

- dla swojego działania wymaga mechanizmu pseudoterminali (na poziomie systemu lub emulowanych);
- klient i serwer działają na poziomie programów użytkownika (nie są częścią implementacji TCP/IP w jądrze);
- zalety:
  - błędy implementacji nie szkodzą działaniu całego systemu;
  - w razie potrzeby można łatwo wymienić całą implementację;
- wadą jest ogromna nieefektywność (wielokrotne przełączanie kontekstu przy przesyłaniu każdego znaku).

Wirtualny terminal sieciowy (VNT)

Różne systemy operacyjne używają różnych konwencji dla reprezentacji końca wiersza, znaku przerwania i tym podobnych. Dla zapewnienia niezależności od systemu TELNET definiuje sposób przesyłania tego rodzaju informacji sterujących przez sieć; definicja ta nosi nazwę *wirtualnego terminala sieciowego (NVT)*:

- komunikacja odbywa się za pomocą ośmiobitowych bajtów;
- o ile negocjacje tego nie zmienią, NVT używa 7–bitowego zbioru znaków USASCII (95 znaków „drukowalnych” i 33 znaki sterujące);

- interpretacja znaków sterujących:

Nazwa	Kod	Znaczenie
`NUL`	0	nic nie rób
`BEL`	7	sygnał (dźwiękowy lub wizualny)
`BS`	8	presunięcie w lewo o jeden znak
`HT`	9	przesunięcie w prawo do tabulatora
`LF`	10	przesunięcie do następnego wiersza
`VT`	11	przesunięcie w dół do pionowego tabulatora
`FF`	12	przesunięcie na górę następnej strony
`CR`	13	przejście do lewego marginesu
inne	—	nic

Polecenia sterujące

- system operacyjny definiuje pewne znaki sterujące; które pozwalają wpływać na działanie programów (np. znak przerwania — w Uniksie standardowo `Ctrl-C`;
- klient protokołu TELNET obsługuje funkcje sterujące; protokół NVT definiuje sposób przekazywania drugiej stronie (serwerowi TELNET-u) informacji na ich temat;
- polecenia są oddzielone od znaków ASCII; ma to następujące zalety:
  - można przekazywać komplet znaków ASCII;
  - nie ma wątpliwości co jest daną, a co kodem sterującym;

- rozpoznawane przez NVT funkcje sterujące (sygnalizowane klientowi przez system):

Funkcja	**Znaczenie**
`IP`	Interrupt Process (przerwij proces)
`AO`	Abort Output (oczyść bufor wyjściowy)
`AYT`	Are You There (test serwera)
`EC`	Erase Character (skasuj poprzedni znak)
`EL`	Erase Line (skasuj bieżący wiersz)
`SYNCH`	Synchronize (opróżnij strumień do pilnych danych, interpretując polecenia)
`BRK`	Break (przerwanie)

- sekwencje sterujące są przesyłane w postaci dwóch oktetów; pierwszy z nich to **IAC** (Interpret As Command); drugi może być jednym z następujących:

Polecenie	**Kod**	**Znaczenie**
`IAC`	255	Interpret As Command (dublowany, gdy wystąpi w danych)
`DON'T`	254	Odmowa zgody na ustawienie opcji
`DO`	253	Zgoda na ustawienie opcji
`WON'T`	252	Odmowa ustawienia opcji
`WILL`	251	Ustawienie opcji
`SB`	250	Dodatkowe negocjacje
`GA`	249	Go Ahead
`EL`	248	Erase Line

- oktety sterujące IAC c.d.

Polecenie	**Kod**	**Znaczenie**
`EC`	247	Erase Character
`AYT`	246	Are You There
`AO`	245	Abort Output
`IP`	244	Interrupt Process
`BRK`	243	Break
`DMARK`	242	Data Mark (dla `SYNCH`)
`NOP`	241	Nic nie rób
`SE`	240	Koniec dodatkowych negocjacji
`EOR`	239	Koniec zapisu

- przesyłanie sekwencji sterujących jest oparte na mechanizmie pilnych danych TCP; po sekwencji sterującej serwer wysyła funkcję sterującą `SYNCH`, a po niej polecenie `DMARK` (znacznik początku normalnych danych), ustawiając segmentowi TCP znacznik pilnych danych.

Opcje TELNETu

- protokół TELNET jest wyposażony w mechanizm negocjacji opcji;
- niektóre opcje proponuje klient, inne serwer, dlatego protokół negocjacji jest *symetryczny*;
- prośbą jest `WILL` *X*, na którą odpowiedziami są `DO` *X* oraz `DON'T` *X*; jednocześnie prośbą jest `DO` *X*, a odpowiedziami `WILL` *X* oraz `WON'T` *X*; dla uniknięcia pętli (z powodu wzięcia odpowiedzi za prośbę) ustalono, że nie potwierdza się prośby o opcję, której się już używa;
- prośba, która nie jest zrozumiała, może zostać po prostu odrzucona; pozwala to na poprawną współpracę programów używających różnych wersji protokołu TELNET;

- najczęściej spotykane opcje TELNET-u:

Nazwa	**Kod**	**RFC**	**Znaczenie**
Transmit Binary	0	856	Przesyłanie ośmiobitowe
Echo	1	857	Pozwolenie na echo odbieranych danych
Suppress-GA	3	858	Nie wysyłaj `GA` po danych
Status	5	859	Pytanie o stan opcji drugiej strony
Timing–Mark	6	860	Prośba o znacznik czasu w strumieniu powrotnym

- najczęściej spotykane opcje TELNET-u c.d.:

Nazwa	**Kod**	**RFC**	**Znaczenie**
Terminal–Type	24	884	Informacja o typie terminala
End–of–Record	25	885	Koniec danych sygnalizowany przez `EOR`
Linemode	34	1116	Edycja lokalna, przesyłanie znaków wierszami