

Network Working Group  
Request for Comments: 864

J. Postel  
ISI  
May 1983

## Character Generator Protocol Протокол генерации символов

Этот документ содержит спецификацию стандарта для сообщества ARPA Internet. Предполагается, что хосты сети ARPA Internet, использующие протокол генерации символов будут реализовать его в соответствии с данным стандартом.

Генератор символов является полезным инструментом для отладки и измерений. Служба генерации символов просто передает поток данных независимо от принимаемой информации.

### **Служба генерации символов на базе протокола TCP**

Сервер генерации символов работает на основе соединений TCP. Сервер прослушивает соединения TCP на порту 19. После организации соединения в него передается поток данных, а полученные из соединения данные сервер просто отбрасывает. Передача потока символов продолжается, пока пользователь не разорвет соединение.

Очевидно, что пользователь этого сервиса может внезапно решить, что ему больше не требуется поток символов и разорвать соединение TCP вместо его корректного завершения. Сервер должен быть готов как к нормальному завершению соединений, так и к их неожиданному разрыву.

Поток данных через соединение ограничивается с помощью обычных механизмов управления потоком TCP, следовательно можно не думать о том, что сервер будет передавать данные быстрее, чем пользователь способен обрабатывать.

### **Служба генерации символов на базе протокола UDP**

Другой вариант службы генерации символов работает на основе протокола UDP. Сервер прослушивает дейтаграммы UDP на порту 19. При получении дейтаграммы в ответ передается пакет (дейтаграмма), содержащий случайное (от 0 до 512) количество символов. Принятые в дейтаграмме данные сервер не принимает во внимание.

Поскольку протокол UDP не поддерживает информации о состоянии, этот тип сервиса не будет давать непрерывного потока символов, а будет просто выдавать символами порциями по 1 дейтаграмме.

Сервер передает 1 дейтаграмму в ответ на каждый запрос клиента, следовательно не возникает вопроса о том, что сервер передает поток символов быстрее, чем пользователь может их обрабатывать.

### **Синтаксис данных**

Данные могут быть совершенно произвольными. Рекомендуется использовать в данных распознаваемый набор символов.

Одним из популярных вариантов является генерация строк, содержащих 72 печатаемых символа ASCII. Код ASCII содержит 95 пригодных для печати символов. Можно отсортировать символы в порядке возрастания кодов и пронумеровать их от 0 до 94. такую последовательность можно рассматривать как кольцевой буфер, в котором после символа 94 следует символ 0. первая строка (строка 0) будет содержать символы 0 – 71, следующая (строка 1) – символы от 1 до 72 и т. д. В строке N будут содержаться символы с номерами от  $(0+N \text{ mod } 95)$  до  $(71+N \text{ mod } 95)$ . Каждая строка завершается символами перевода строки и возврата каретки<sup>1</sup>.

### **Пример**

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ [\]^_`abcdefghijklmnopgh  
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ [\]^_`abcdefghijklmnopghi  
#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ [\]^_`abcdefghijklmnopghij  
$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ [\]^_`abcdefghijklmnopghijk  
%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ [\]^_`abcdefghijklmnopghijkl  
&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMN OPQRSTUVWXYZ [\]^_`abcdefghijklmnopghijklm
```

<sup>1</sup> CRLF. Прим. перев.

```
'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnop
()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnop
()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnop
*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopq
+,-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqr
,-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrs
-./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrst
./0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstu
/0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuv
0123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvw
123456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwx
23456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxy
3456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz
456789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{
56789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|
6789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}
789:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
89:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
9:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !
:;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"
;<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"#
<=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $
=>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ %
>?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % &
?@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & '
@ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' (
ABCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( )
BCDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) *
CDEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * +
DEFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + ,
EFGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , -
FGHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - .
GHIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . /
HIJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0
IJKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0 1
JKLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0 1 2
KLMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0 1 2 3
LMNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0 1 2 3 4
MNopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0 1 2 3 4 5
NOPqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}~ !"# $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6
```

## Перевод на русский язык

Николай Малых  
[nmalykh@bilim.com](mailto:nmalykh@bilim.com)