NetKeys

Kyzer/CSG

COLLABORATORS			
	TITLE : NetKeys		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Kyzer/CSG	April 13, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

NetKeys

1.1 NetKeys v1.1 documentation

NetKeys v1.1

Introduction

General usage

NetKeys options

NetKeys history

Credits

1.2 Introduction

In one sentence: NetKeys is a modern implementation of the SerNet/ParNet 'netkeys' for the ProNET network.

NetKeys allows you, through software alone, to disconnect your keyboard (and optionally the mouse) from one computer, and connect it to another.

It does this by grabbing all the keyboard and mouse input on one computer and sending it down a network connection to a 'server' program running on the target computer, which will inject that input as if it originated from a keyboard or mouse attached.

1.3 Using NetKeys

Requirements:
Both NetKeys and KeyServ require Workbench 2 (v37), and a working ProNET installation.

In addition, NetKeys (the client program) requires the commodities.library to be available. KeyServ (the server program) does not require any disk-based libraries.

Installation:
You need to run two different programs on the two machines.

The client machine: the Amiga which has the keyboard/mouse.
The server machine: the Amiga which does not have the keyboard/mouse.

'NetKeys' runs on the client machine.
'KeyServ' runs on the server machine.

You need to know the settings of ProNET between the client and server. You need to know which machine the client knows the server as, and vice versa.

For most ProNET installations, there is only one pair of machines connected to each other, where the client knows the server as machine 0, and the server knows the client as machine 0.

Server installation
Copy KeyServ into C: on the server machine, and add the following line to the S:startup-sequence (before pronet-server if you wish):

```
Run <>NIL: KeyServ [machine]
```

where [machine] is the number the server knows the client as.

Now reboot the server machine so KeyServ gets started.

Client installation
Copy the NetKeys program to wherever you like, possibly the WBStartup drawer. You can run NetKeys from either Workbench and CLI. Set the
options
as you want.

Now run NetKeys and press the hotkey. Hey presto, you should be typing on the server machine. Press the hotkey again, and you should be typing on the client machine again.

1.4 NetKeys options

You may change how NetKeys operates using the following ↩
options, from
either the command line, or as icon tooltypes:

MOUSE
Setting this option means that NetKeys not only sends keypresses from your keyboard, but it also sends all your mouse movements too.

Commodity options

CX_HOTKEY
The key sequence which, when pressed, will toggle NetKeys on or off.

You provide this in the normal Commodities style, eg 'ralt lamiga r' or such. If you do not specify a hotkey, 'ctrl alt k' will be used.

CX_PRIORITY
This is the priority within the Commodities system where NetKeys goes. As NetKeys is meant to remove input from the client machine and send it to the server machine, NetKeys should be above all other commodities and applications, so it can remove input before anyone else sees it.

Priority is a number between -128 and 127. The default priority is 120.

CX_ACTIVATE
Normally, NetKeys is dormant when you load it, until you press the hotkey to activate it. Setting this option means that NetKeys will be active from the moment it is loaded.

Network settings

MACHINE
This is the 'machine number' of the server, as described in the
usage
section. The default is 0.

1.5 NetKeys history

```
                1.0:  First release

1.1:  Fixed serious bug in KeyServ - was doing
  ReplyMsg() to port, not message.


                 Future development
```

1.6 Future prospects for NetKeys

```
                - Ask Michael Krause to build KeyServ into pronet-server?

Further suggestions are welcome.
                Send them to me.
```

1.7 Credits

```
NetKeys and KeyServ were written by Kyzer/CSG in Amiga E 3.3a.

Best wishes to the following people:
 Michael Krause for ProNET.
 Matt Dillon for the original netkeys (and ParNet, and SerNet).

Contact:
Kyzer/CSG,
49 Fairview Road,
AB22 8ZG, Scotland.

or email: kyzer@4u.net

http://www.abdn.ac.uk/~u13sac/
```