

Tunnel Server

- Problem: people need help to get started
- plug and play IPv6 using the current IPv4 Internet as the transport
- ideas:
 - tunnel broker (Alain Durand)
 - web site for info/implementations/... (Orlando Bof)

What we did

- A concept of freenet for IPv6
- Made an implementation of a tunnel server
- Domain name freenet6.net. Used to register the user 's IPv6 address with a nickname under freenet6.net
- Put a web site with pointers to most implementations, and a faq

User interface

- Go to www.freenet6.net (freenet6.viagenie.qc.ca)
- Click to get the IPv6 release you need (depending on the platform)
- Fillout a form
 - with your IPv4 address (prefilled with what we received from the web client)
 - Choose a nickname (which generates an entry in the dns for `nickname.country.freenet6.net`)
 - Choose a country (just for the purpose of naming space: can be anything) (next version: choose a server with a name space)

User interface

- (server creates its tunnel end point)
- receive a script that you execute: this script creates the tunnel on the client side.
- You are connected

- Other service: check « who is on this IPv6 server » lists all active nicknames

User interface



[Get your own IPv6 tunnel](#)

[Host Implementations of IPv6](#)

[Frequently asked questions](#)

User interface



FreeNet6: Create your own tunnel

First, select the Operating System you are using. After, you'll only need to follow the instructions.

- [FreeBSD/INRIA](#)
- [FreeBSD/KAME](#)
- [Windows NT](#)

User interface

IPv6 Tunneling with FreeBSD/KAME

Validate

or correct
your IPv4
address

Enter

your
nickname

Select

your
country

Implementation

- Apache web server
- Perl scripts
- DNS server primary of « country ».freenet6.net
- FreeBSD with kame stack
- Currently supported clients: NT, FreeBSD/Kame, FreeBSD/Inria. Very easy to add new clients (if the tunnel creation can be scripted.)

Future work

- Support for more host implementations
- Support for multiple servers
 - a server register itself as a tunnel server in the registry
 - syntax for the registry defined in a separate document
 - user choose the server in the list presented based on registry entries
- Support for ipv6 routers (net behind tunnel endpoint)
- v6 tunnels over IPv4 nat (ugly!)
- Make the code available
- Test with MobileIPv4
- Currently working with Alain Durand and CSELT