



the globus alliance  
www.globus.org

# GridFTP for Users

John Bresnahan

Michael Link

Raj Kettimuthu

Argonne National Laboratory

University of Chicago



# Obtain Installer Now

## GridFTP Tutorial

- Installing to a remote machine
  - <http://www.gridftp.org/tutorials>
  - Handout on build instructions and exercises available here
- Installing to laptop (linux and mac users)
  - 1 of 2 ways
    - USB Drive
    - <http://www.gridftp.org/tutorials>



# Outline

- Introduction
- Security Options
- GSI Configuration
- Optimizations
- Advanced Configurations
- New Features



# What is GridFTP?

- High-performance, reliable data transfer protocol optimized for high-bandwidth wide-area networks
- Based on FTP protocol - defines extensions for high-performance operation and security
- Standardized through Open Grid Forum (OGF)
- GridFTP is the OGF recommended data movement protocol



# GridFTP

- We (Globus Alliance) provide a reference implementation:
  - Server
  - Client tools (globus-url-copy)
  - Development Libraries
- Multiple independent implementations can interoperate
  - Fermi Lab and U. Virginia have home grown servers that work with ours



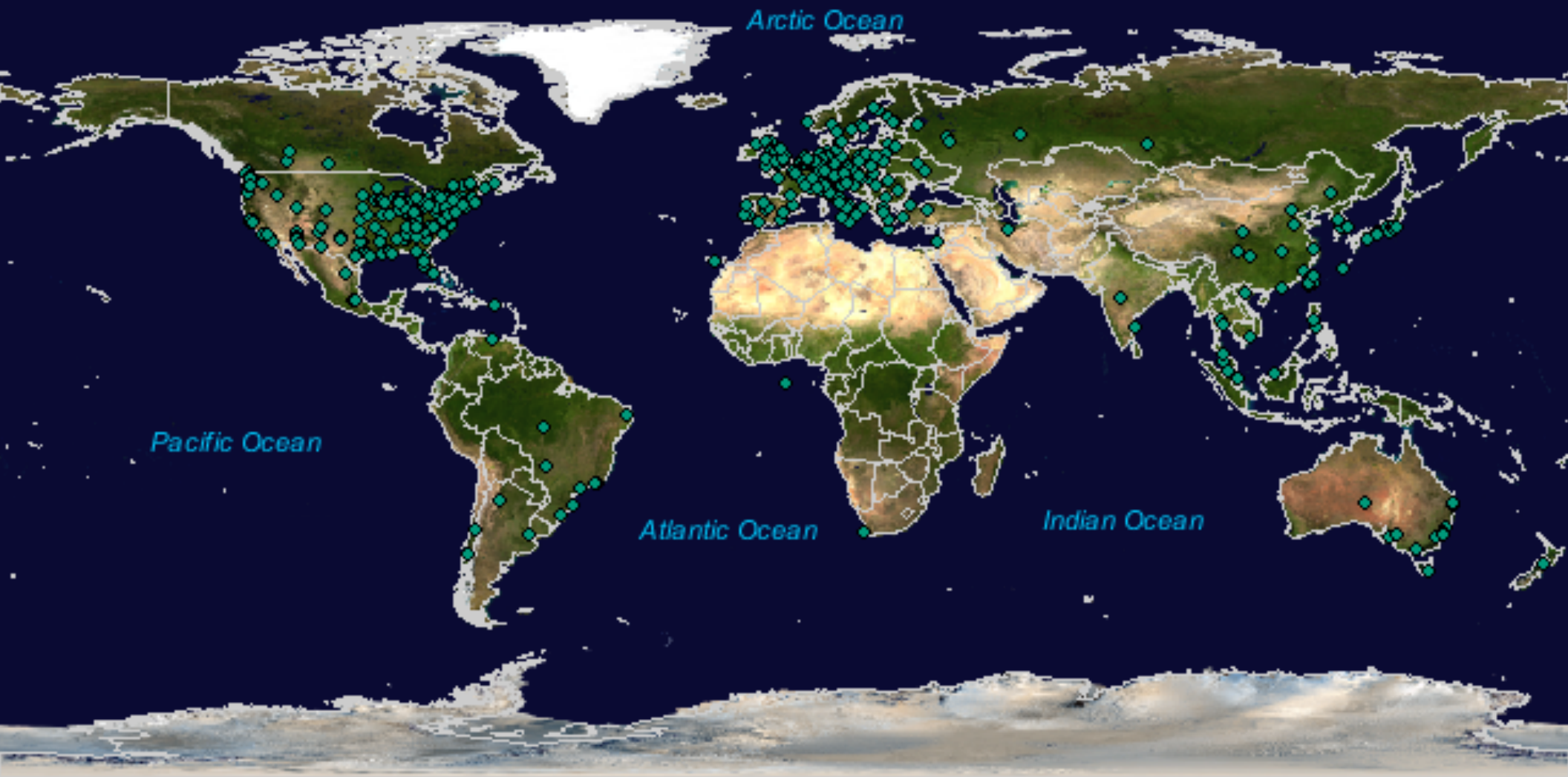
# Globus GridFTP

- Performance
  - Parallel TCP streams, optimal TCP buffer
  - Non TCP protocol such as UDT
- Cluster-to-cluster data movement
- Multicasting, Overlay routing
- Multiple security options
  - Anonymous, password, SSH, GSI
- Support for reliable and restartable transfers



the globus alliance  
www.globus.org

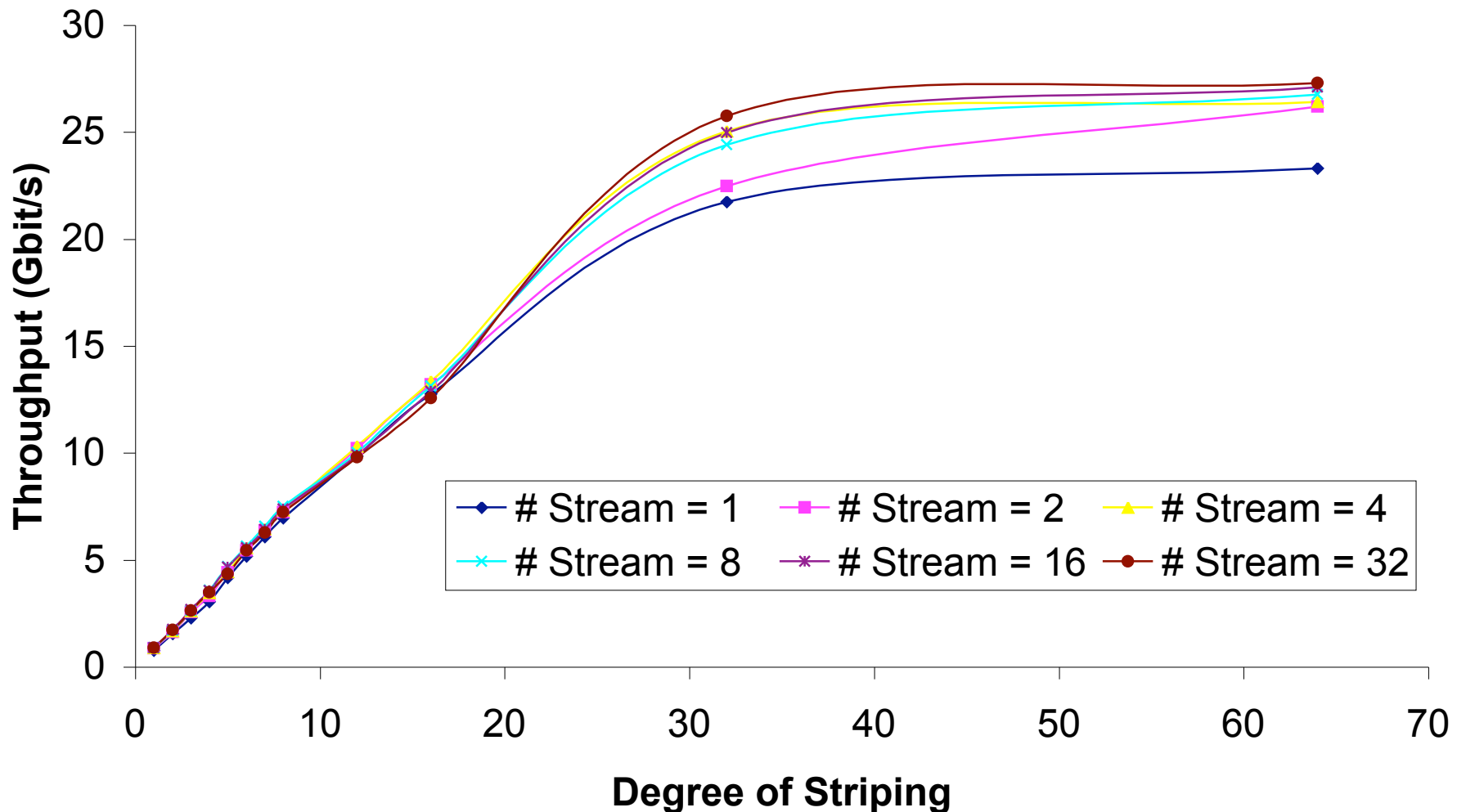
# GridFTP Servers Around the World



Created by Lydia Prieto ; G. Zarrate; Anda Imanitchi (Florida State University) using  
MaxMind's GeoIP technology (<http://www.maxmind.com/app/ip-locate>).



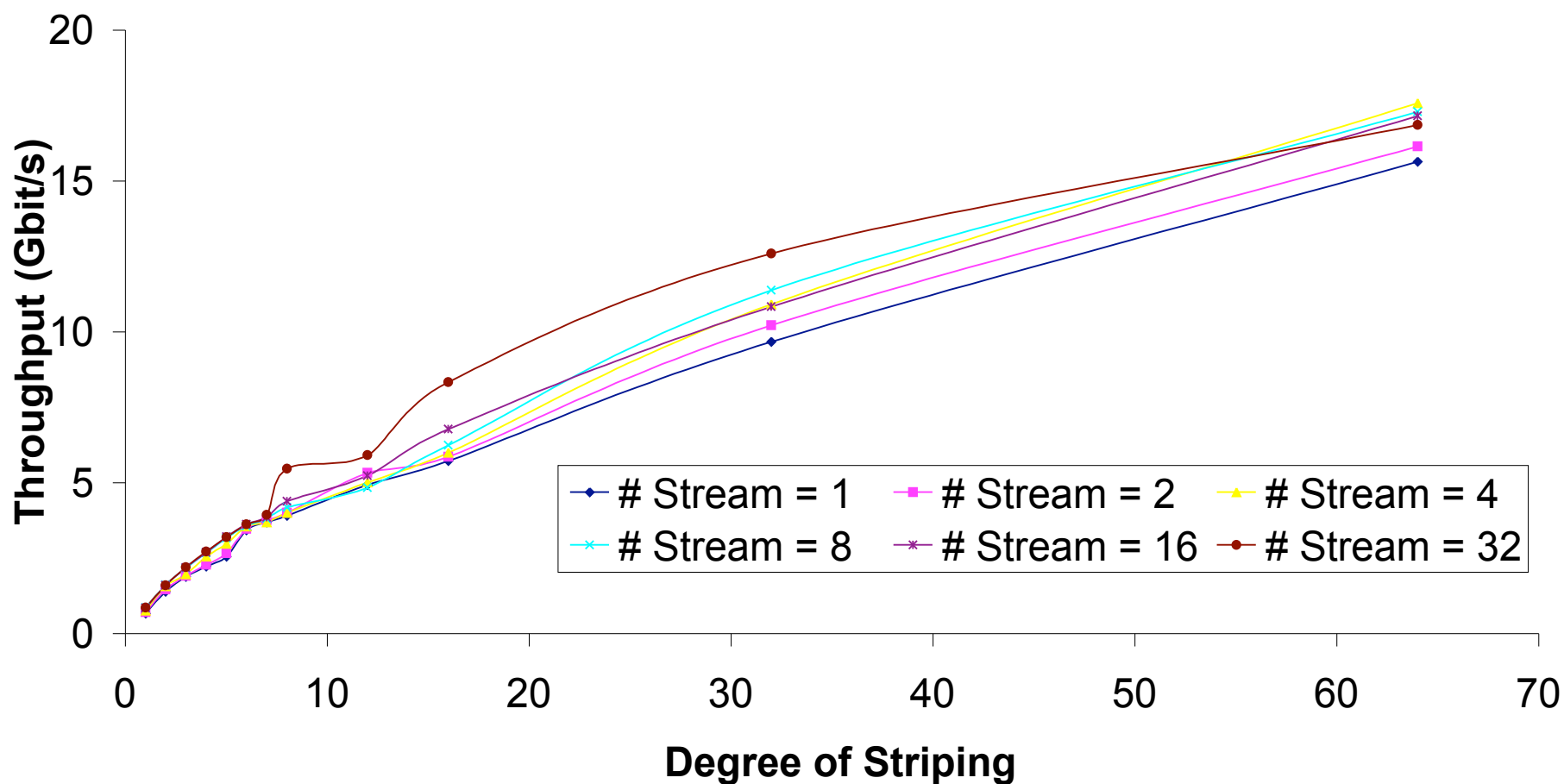
# Memory to Memory over 30 Gigabit/s Network (San Diego — Urbana)







# Disk to Disk over 30 Gigabit/s Network (San Diego — Urbana)



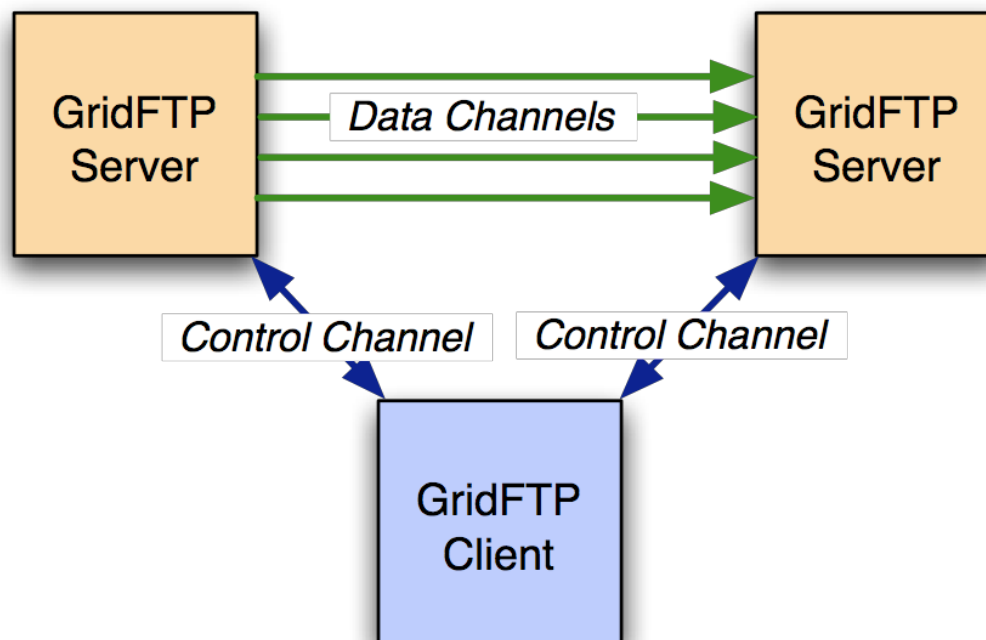
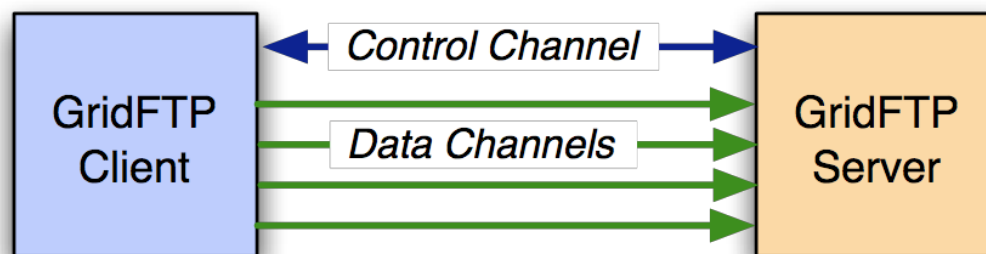


# Understanding GridFTP

- Two channel protocol like FTP
- Control Channel
  - Command/Response
  - Used to establish data channels
  - Basic file system operations eg. mkdir, delete etc
- Data channel
  - Pathway over which *file* is transferred
  - Many different underlying protocols can be used
    - MODE command determines the protocol



# Client/Server and 3rd Party Transfer





# Globus-url-copy

- Command line scriptable client
- Globus does not provide an interactive client
- Most commonly used for GridFTP, however, it supports other protocols
  - gsiftp:// (GridFTP, historical reasons)
  - ftp://
  - http://
  - https://
  - file://



# Syntax Overview

- `globus-url-copy [options] srcURL dstURL`
- `guc gsiftp://localhost/foo file:///bar`
  - Client/server, using FTP stream mode
- `guc -vb -dbg -tcp-bs 1048576 -p 8 gsiftp://localhost/foo  
gsiftp://localhost/bar`
  - 3<sup>rd</sup> party transfer, MODE E



# URL Rules

- protocol://[user:pass@[host]/path
- For guc supported protocols are:
  - gsiftp:, ftp:, file:, http:, https:
- host can be anything resolvable
  - IP address, localhost, DNS name



# Security Options

- Clear text (RFC 959)
  - Username/password
  - Anonymous mode (anonymous/<email addr>)
  - Password file
- SSHFTP
  - Use ssh/sshd to form the control connection
- GSIFTP
  - Authenticate control and data channels with GSI
- User is mapped to a local account and file permissions are handled by the OS



# Exercise 1

## Anonymous Transfer

- **Install the GridFTP Server**
  - `http://www.gridftp.org/tutorials/`
  - `tar xvfz gt-gridftp*.tar.gz`
  - `cd gt-gridftp-installer`
  - `./configure -prefix /path/to/install`
    - *ignore any java/ant warnings*
  - `make gridftp install`
- **Setup the environment (repeat for all globus sessions)**
  - `export GLOBUS_LOCATION=/path/to/install`
  - `source $GLOBUS_LOCATION/etc/globus-user-env.sh`





# Exercise 1

- Run a two party transfer
  - globus-url-copy -v <file:///etc/group> <ftp://localhost:5000/tmp/group>
- Run 3<sup>rd</sup> party transfer
  - globus-url-copy -v <ftp://localhost:<port>/etc/group> <ftp://localhost:<port>/tmp/group2>
- Experiment with -dbg, -vb options
  - globus-url-copy -dbg <file:///etc/group> <ftp://localhost:5000/tmp/group>
  - globus-url-copy -vb <file:///dev/zero> <ftp://localhost:5000/dev/null>



# Exercise 2

## Password file

- Connect with standard ftp program
  - ftp localhost 5000
  - ls, pwd, cd, etc...
- Transfer with globus-url-copy
  - globus-url-copy <file:///etc/group> <ftp://username:pw@localhost:5000/tmp/group>
  - globus-url-copy -list <ftp://username:pw@localhost:5000/>



# Exercise 3

## sshftp

- Configure SSHFTP
  - `$GLOBUS_LOCATION/setup/globus/setup-globus-gridftp-sshftp`
    - Enables **client** support for `sshftp://` urls for this `$GLOBUS_LOCATION`
- `globus-url-copy` transfers
  - `globus-url-copy -v file:///etc/group sshftp://localhost/tmp/group`
  - `globus-url-copy -list sshftp://localhost/tmp/`

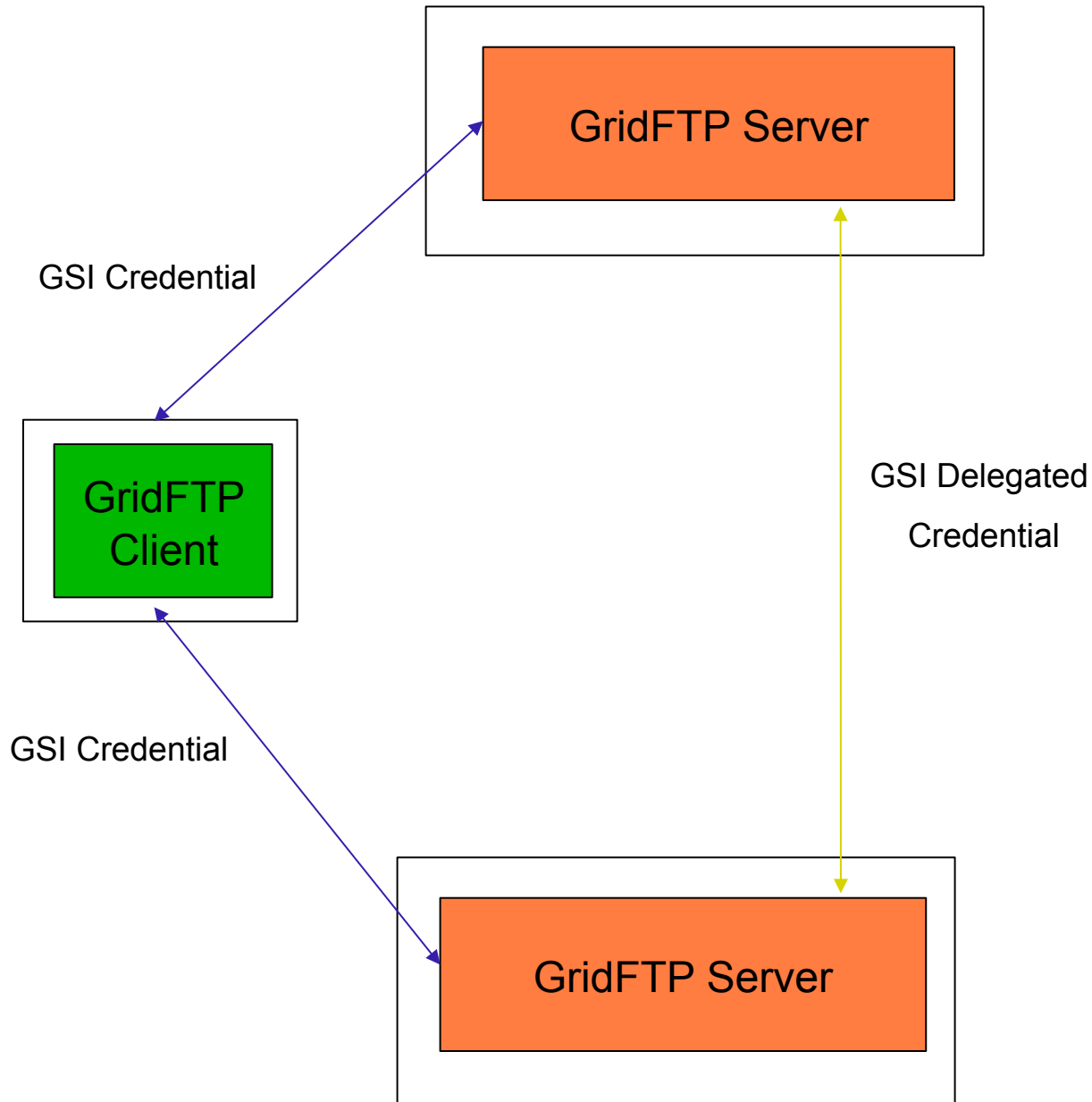


# GSI Authentication

- Strong security on both channels
  - SSH does not give us data channel security
- Delegation
  - Authenticates DC on clients behalf
  - Flexibility for grid services such as RFT
    - Agents can authenticate to GridFTP servers on users behalf
  - Enables encryption, integrity on data channel



# GSI Authentication





# Certificates

- **Central concept in GSI**
  - Information vital to identifying and authenticating user/service
- **Certificate Authority (CA)**
  - Trusted 3<sup>rd</sup> party that confirms identity
- **Host credential**
  - Long term credential
  - Allows a client to verify the host is what they expect
- **User credential**
  - Passphrase protected
  - Used to activate a short term proxy



# Exercise 4

## GSI Security

- Obtain a user credential
- Create proxy
  - grid-proxy-init
- Get your DN added to gridmap file
- Perform a GSI authenticated transfer
- Evaluate results



# Troubleshooting

- Can I get connected?
  - telnet to the port: telnet hostname port
  - 2811 is the default port
- You should get something like this:
  - <add GridFTP banner>
- If not, you have firewall problems, or server config problems.





# Troubleshooting

- no proxy
  - grid-proxy-destroy
  - guc gsiftp://localhost/dev/zero file:///dev/null
  - add –dbg
  - grid-proxy-init
  - guc gsiftp://localhost/dev/zero file:///dev/null
  - add –dbg



# Troubleshooting

- Bad source file
  - grid-proxy-init
  - guc gsiftp://localhost:2811/tmp/junk file:///tmp/empty
    - junk does not exist



# Optimizations

- TCP buffer size
- Parallel streams



# TCP Buffer Size

- Most important tuning parameter for TCP
  - Memory the kernel *allocates* for retransmits/reordering
  - Affects the maximum window size
  - Amount of data that can be sent before receiving an acknowledgment (ACK)
- Bandwidth Delay Product (BWDP)
  - $BWDP = \text{latency} * \text{bandwidth}$
  - The optimal number of bytes that is needed to keep the network pipe full



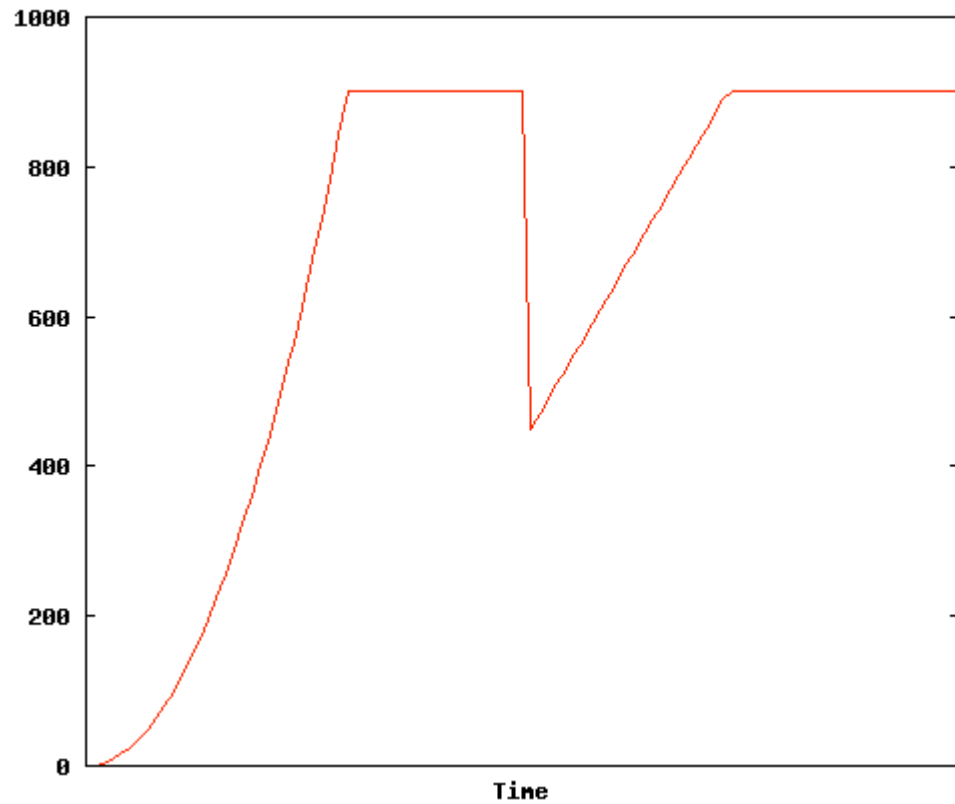
# Why Parallel TCP?

- Taking advantage of loopholes in the system
  - *Cheat* TCP out of intended fair backoff
- Reduces the severity of a congestion event
  - Only effects  $1/p$  of the overall transfer
- Faster recovery
  - Smaller size to recover
- Work around for low TCP buffer limit

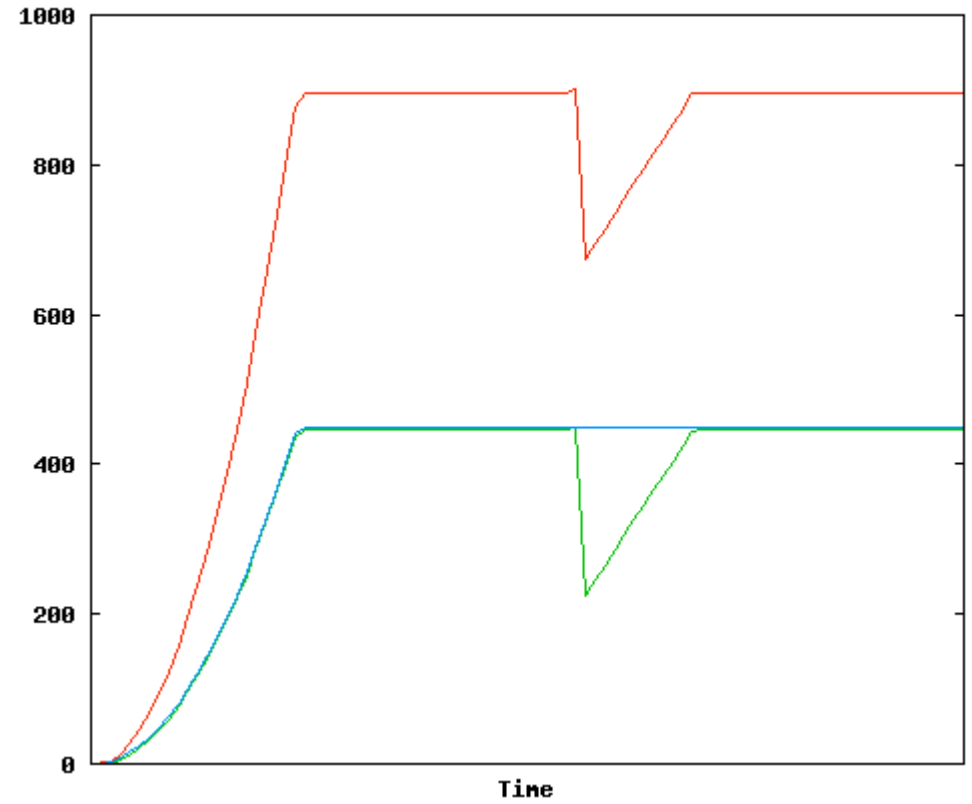


# Lost Packets

One Stream



Two Streams





# Optimization options

- -p (parallelism or number of streams)
  - rule of thumb 4-8, start with 4
- -tcp-bs (TCP buffer size)
  - use either ping or traceroute to determine the RTT between hosts
  - buffer size =  $BW \text{ (Mbs)} * RTT \text{ (ms)} * 1000 / 8 / <(\text{parallelism value} - 1)>$
  - If that is still too complicated use 2MB
- -vb if you want performance feedback
- -dbg if you have trouble



# Demonstration 1

## Performance

- Transfer on a real network
  - Show performance markers
  - Show transfer rate
- Calculate the BWDP
- Vary -tcp-bs
- Vary -p





# Partial File Transfer

- **Large file transfer fails**
  - We don't want to start completely over
  - Ideally we start where we left off
- **Restart markers sent periodically**
  - Contain blocks written to disk
  - Sent every 5s by default
  - In worst case recovery sends 5s of redundant data



# Advanced configuration - UDT

- Provides a reliable layer on top of UDP
- Designed to provide optimal performance on high bandwidth high latency networks
- GridFTP supports this as an alternative protocol for TCP



# Demonstration 2

## GridFTP over UDT

- Show a transfer with UDT as the transport protocol
  - Show dynamic data channel protocol stack selection
  - Show the performance increases
- Requirements
  - Threaded build of the Globus GridFTP server
  - Threaded build of globus-url-copy (for client-server transfers)
- Transferring a file
  - globus-url-copy -udt <file:///etc/group> <ftp://localhost:5000/tmp/group>



# Transferring lots of small files

- Use concurrency optimization for transferring lots of small files
- What is a small file?
  - Depends on the network bandwidth and latency
  - Files of size  $\leq$  100 MB
- Transfer multiple files concurrently
  - `globus-url-copy -cc`



# Demonstration 3

## Concurrency

- Show lots of small files transfers with and without concurrency
  - Show performance increases with concurrency



# Feedback

- Comments welcome
- If you need any specific functionality requirement, please let us know



# Thank you

- More Information:
  - <http://www.gridftp.org>
  - <http://www.globus.org/toolkit>
  - `gridftp-user@globus.org`