

Presentation of the Buschtrommel RFC - An Ad-hoc Filesharing Protocol

Felix Wilhelm, Sebastian Leidig, Daniel Reinhardt

Institute of Telematics, Prof. Zitterbart



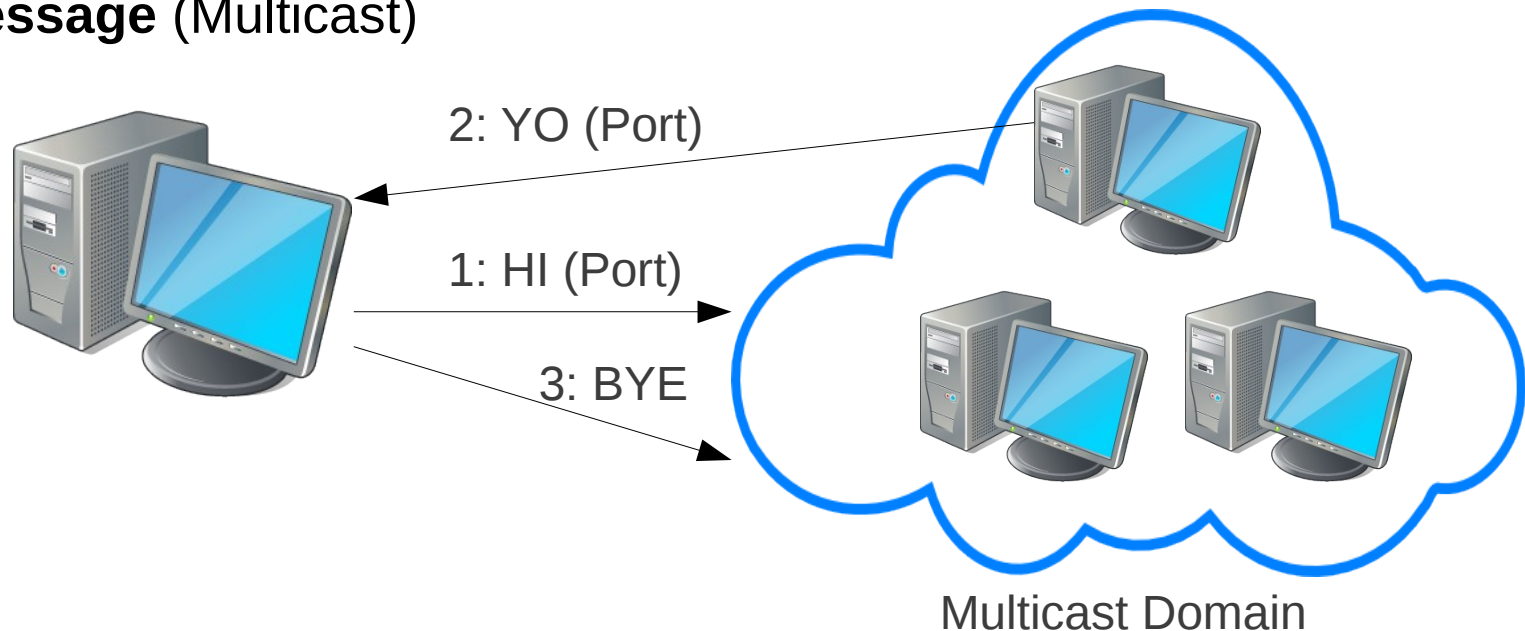
- Protocol used for ad-hoc file transfers between peers in the same network domain
- Focused on scenarios where the set of shared files is changed frequently
- Files are announced in a short-termed fashion
- Clients are conforming and well-behaved

Highlevel Overview – General Tasks

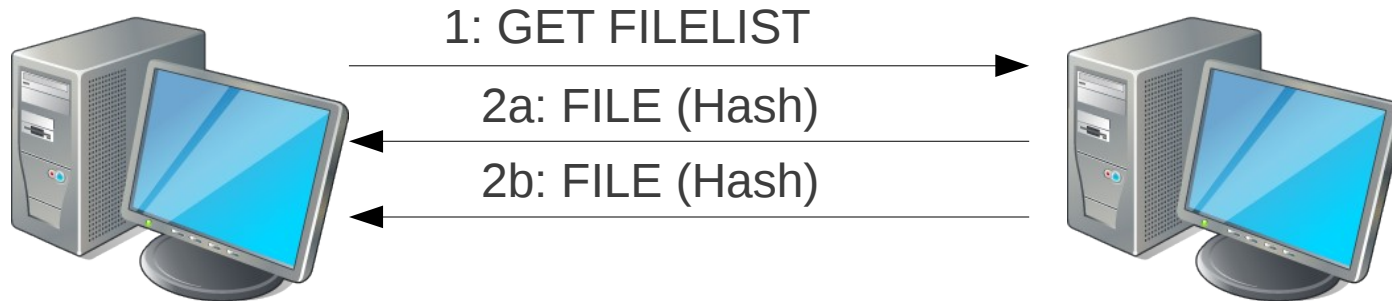
- Announcements of peers
- Announcements of files
 - Exploration of offered files
 - Updates of offered files
- File transfers

Announcements of peers

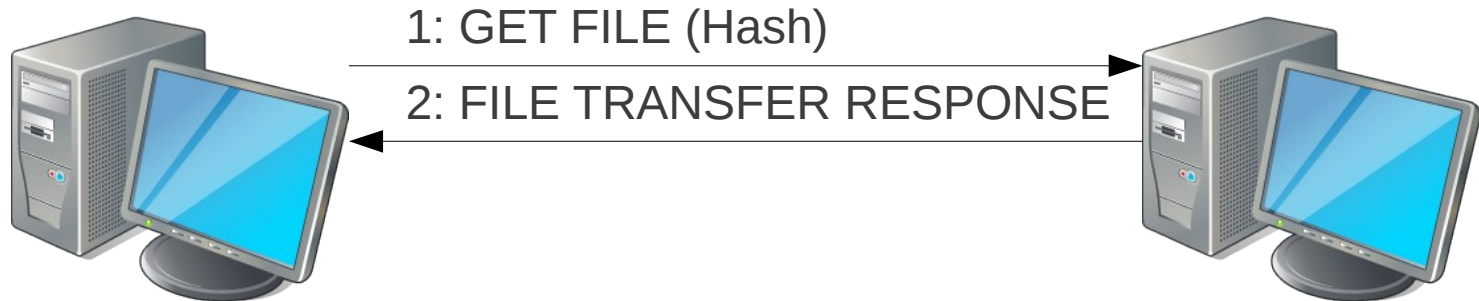
- New peer inform other peers of its presence via a **HI Message** (Multicast)
- Other peers reply the **HI Message** via a **YO Message** (Unicast/Multicast)
- Peer informs other peers when he leaves the network via a **BYE Message** (Multicast)



Announcements of Files



File Transfers



General Notes

- Multicast addresses
 - IPv4: 239.255.0.113
 - IPv6: ff05::7171

- UDP port
 - 4747

- Max. message length
 - 512 octets

Message Structure

■ All messages encoded as UTF-8 strings

■ Separators:

- Fields: ASCII control character 31 (Unit Separator, U+001F)
- Messages: ASCII control character 30 (Record Separator, U+001E)

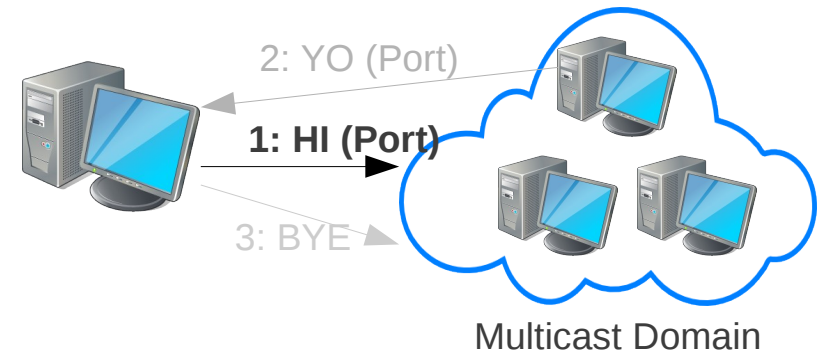
■ Structure:

Type	Field_1	Field_2	...	Field_n
------	---------	---------	-----	---------

- Type must be in uppercase

„HI“ Message

- Connect to Buschtrommel network

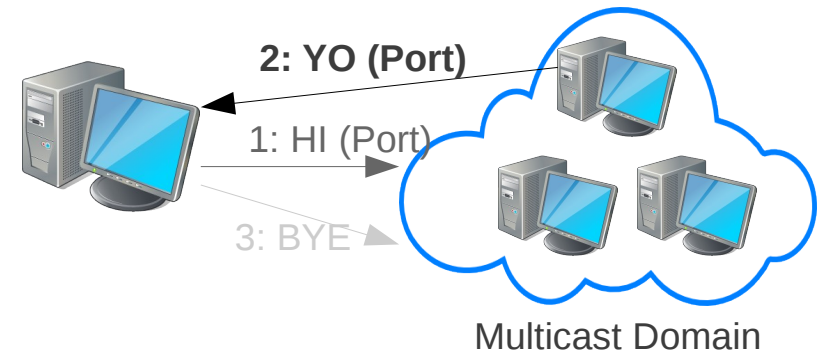


Type	Port	Alias
HI	<i>TCP port for further communication</i>	<i>human-readable name</i>

➡ „HI[U+001F]4000[U+001F]peerA[U+001E]“

„YO“ Message

- Reply to „HI“



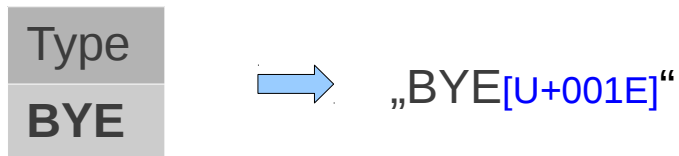
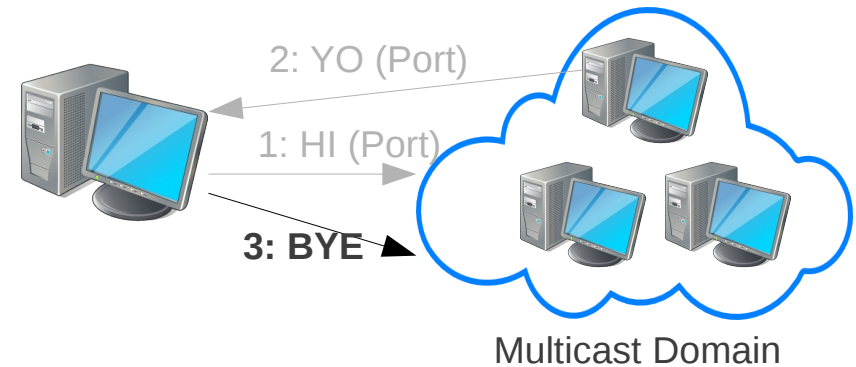
- Identical fields as „HI“ message

Type	Port	Alias
YO	<i>TCP port for further communication</i>	<i>human-readable name</i>

➡ „YO[U+001F]4000[U+001F]peerB[U+001E]“

„BYE“ Message

- Leaving the network



File Discovery

- Request a list of offered files directly from a single peer

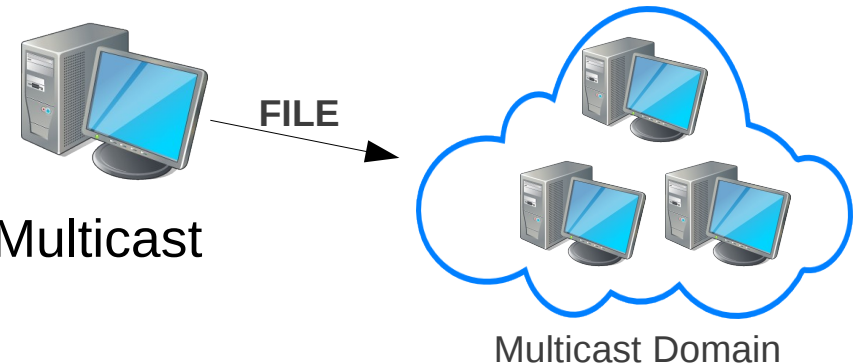


- Reply consists of multiple File Announcement messages

Type	Hash	TTL	Length
FILE	<i>SHA1 Hash of File Content</i>	<i>Expected TTL in seconds</i>	Length of File
Display name	Meta Information		
<i>Human-readable name for file</i>	<i>Additional information</i>		

File Announcement

- Updates about available files:
new, deleted, updated TTL
- Same format, but sent using UDP Multicast



Type	Hash	TTL	Length
FILE	<i>SHA1 Hash of File Content</i>	<i>Expected TTL in seconds</i>	Length of File
Display name	Meta Information		
<i>Human-readable name for file</i>	<i>Additional information</i>		

File Transfer

- „GET FILE“ message to request offered file.
- Sent using TCP connection

Type	Hash	Offset	Length
GET FILE	<i>SHA1 Hash of File Content</i>	<i>Byte Offset</i>	<i>Number of requested bytes</i>

- „Server“ responds with „FILE TRANSFER RESPONSE“

Type	Status	Expected Transfer Size
FILE TRANSFER RESPONSE	OK / TRY AGAIN LATER / NEVER TRY AGAIN	<i>Expected number of bytes to send</i>

Problems and Solutions

■ Mailing List communication

- Too many participants
- No group internal consent
- Hard to recognize which opinion belongs to a group
- Flawed moderation ↔ Weekly changes
- **Solution: Weekly summaries**

■ „Lost“ Groups (Draft/Meeting/Moderation/...)

■ Hard to organize meetings

- Intergroup meeting on short notice with too few participants
- **Quite suitable for better consent!**