**#17869** -<https://redmine.vnc.biz/issues/17869>

1. **Description**:

|  |
| --- |
| This is a nice to have requirement. Implementing this requirement is not mandatory for this release.  Requirement name: send a file by drag&drop  Description: Allow the user to send a file using the drag&drop technique  Origin: Rainer  When was this requirement proposed? 05.12.2014  Requirement designed by: Rainer  When was this requirement designed? 05.12.2014  Rationale: It may be very useful for the user to only drop the files that need to be transmitted into the chat windows and that the file transfer for each file will be started.  Difficulty: normal  Customer satisfaction: The customer will be very satisfied.  Relations: File Transfer (13062)  Priority: low  Validation: Using the drag&drop technique select a few files and transmitt them to the other user.  Document structure: admin and user documentation |

1. **Analytic**
   1. **Solutions**:

Use Drag & Drop event of jQuery

Allow drag and drop file to chat 1-1 window, now only permit drop one by one file.

Allow drag and drop file after user reload page and keep chat 1-1 window

1. **Implementation**
   1. **Code:**

**Create function for add event drap&drop:**

|  |
| --- |
| **self.AllowDropFileTransfer = function(hash, xid){  var obj = jQuery("#received-" + hash);  obj.on('dragenter', function (e) {  e.stopPropagation();  e.preventDefault();  //console.log("Dragenter");  });   obj.on('dragover', function (e) {  e.stopPropagation();  e.preventDefault();  //console.log("Dragover");  });   obj.on('drop', function (e) {  e.stopPropagation();  e.preventDefault();  console.log("Drop file to transfer");  var files = e.originalEvent.dataTransfer.files;  //send dropped files to Server  self.\_startFileTransfer(xid, files);  }); }** |

Change startfiletransfer function:

|  |
| --- |
| //Start file transfer by selecting file self.startFileTransfer = function(buddy){   var fileDialog = document.createElement("INPUT");  fileDialog.setAttribute("type", "file");  fileDialog.addEventListener('change', function(evt){  var files = evt.target.files;  self.\_startFileTransfer(buddy, files);   },  false);  fileDialog.click(); }; |

Add new function \_startFileTransfer:

|  |
| --- |
| //Star file transfer with file self.\_startFileTransfer = function(buddy, files){  if (files.length > 1) {  console.log("Drop too many files!");  var dialog = appCtxt.getMsgDialog();// get a simple message dialog  dialog.reset(); // reset the dialog  dialog.setMessage("Don't select many files, please drop one by one!"); // set the message "info" style  dialog.popup();  return; }; var file = files[0];   self.createFileTransferStatus();  self.switchPane('chat-file\_transfer', 'file\_transfer');   /\*  if(file.size >= 10485760) {  console.log("This size too large !");  var dialog = appCtxt.getMsgDialog();// get a simple message dialog  dialog.reset(); // reset the dialog  dialog.setMessage(biz\_zcs\_vnc\_talk\_zimlet.vnc\_talk\_limit\_size); // set the message "info" style  dialog.popup();  return;  }  \*/   log('Offering:', file.name, file.size, file.type, file.lastModifiedDate);   transferSession = self.client.jingle.createFileTransferSession(buddy + "/file\_transfer");   var session = {  issend: true,  sid: transferSession.sid,  from: transferSession.parent.jid.bare,  to: buddy,  filename: file.name,  size: file.size,  session: transferSession,  };   transferSession.sender.on('progress', function (sent, total) {  log('Sent to ' + buddy + ': ' + sent + '/' + total);  session.value = sent;  self.displayFileTransferStatus(session,'progress');  });   transferSession.on('terminated',function(transferSession,reason){  session.value = reason.condition;  self.displayFileTransferStatus(session,reason.condition);  });   transferSession.start(file);   var hash = hex\_md5(buddy);  var meta = {"name": file.name,"algo":"sha-1","hash":hash};  transferSession.sender.on('sentFile',function(meta){  self.displayFileTransferStatus(session,'done');  });   self.displayFileTransferStatus(session,'started');   //Set TimeOut:  var timer = setTimeout(function(){  console.log("Session timeout");  transferSession.end('timeout',false);  },  1800000 // 30 min  );   //Clear timer  transferSession.on('accepted',function(){  console.log("accepted");  clearTimeout(timer);  });  }; |

Add drag&drop event when create chat 1-1 window:

|  |
| --- |
| self.chat = function(type, xid, nick, hash, pwd, show\_pane,isAcceptDefault) {  ......  // Alow drop file transfer  if (type == 'chat') {  self.AllowDropFileTransfer(hash, xid);  };  ...... }; |

Add drag&drop event after reload page:

|  |
| --- |
| self.create = function(domain, user, password) {  ......  //Restore drop file transfer  $(dom).find('.jm\_conversation').each(function(i, div){  var hash = $(div).attr("data-hash");  var xid = $(div).attr("data-xid");  if (xid) {  console.log("Old hash ", hash);  console.log("Old xid ", xid);  self.AllowDropFileTransfer(hash,xid);  };  })  ...... }; |

Committed to release\_2.5.0 <https://redmine.vnc.biz/projects/prosody-packaging/repository/revisions/47ef906645552f64696a62c6a3d9aa23f08edefc>