

Act 2

node in action

party.js

```
var EventEmitter = require('events').EventEmitter  
    , party = new EventEmitter();
```

```
party.on('arrive', function (friend) {  
    console.log(friend + ' is here!');  
});
```

```
party.emit('arrive', 'Annie'); // => Annie is here!  
party.emit('arrive', 'Becky'); // => Becky is here!  
party.emit('arrive', 'Carl');  // => Carl is here!
```

server.js

```
var http = require('http')
    , i = 1;

http.createServer(function (req, res) {

    res.writeHead(200, { 'Content-Type': 'text/plain' });

    res.end('Hello visitor #' + i + '\n');

    ++i;

}).listen(4000, '127.0.0.1');

console.log('Server running on port 4000!');
```

compress.js

```
#!/usr/bin/env node
```

```
var zlib = require('zlib')  
    , gzip = zlib.createGzip();
```

```
process.stdin.pipe(gzip)  
              .pipe(process.stdout);
```

compressing

```
$ cat big.json | ./gzip.js | small.json
```

decompress.js

```
a.pipe(b).pipe(c);
```

mac/linux

```
$ cat mystery.jpg.gz | ./decompress.js > out.jpg
```

windows

```
$ type mystery.jpg.gz | ./decompress.js > out.jpg
```

what does this do?

```
a.pipe(b).pipe(a);
```






chat.js

our final adventure!

open [exercises/chat/client.js](#)

twitch plays tug-o-war

``node client.js --port 8081 ``

hack time!

- build a bot?
- sabotage other bots?
- edit `solutions/chat/server.js`?

■

,

these slides are online
the code is online
... no promises about the chat server