

udp

Return Types

[JSPacket](#)

Method Summary

JSPacket	createNewPacket() Create a new empty packet.
JSPacket	getReceivedPacket() Get a packet from receive buffer, read buffer until empty (null is returned).
Boolean	sendPacket(destIpOrHostname, packet) Send a packet.
Boolean	sendPacket(destIpOrHostname, packet, port) Send a packet on another port.
Boolean	startSocket(port_number, method_to_call_when_packet_received_and_buffer_is_empty) Start a UDP socket for a port.
void	stopSocket() Stop the UDP socket for a port.
Boolean	testPacket(packet) Put a test packet in the receive buffer to test your method call and getReceivedPacket.

Method Details

createNewPacket

[JSPacket](#) **createNewPacket()**
Create a new empty packet.

Returns

[JSPacket](#)

Sample

```
var packet = plugins.udp.createNewPacket()
packet.writeUTF('hello world!')//writes UTF
packet.writeInt(12348293)//writes 4 bytes
packet.writeShort(14823)//writes 2 bytes
packet.writeByte(123)//writes 1 byte
```

getReceivedPacket

[JSPacket](#) **getReceivedPacket()**
Get a packet from receive buffer, read buffer until empty (null is returned).

Returns

[JSPacket](#)

Sample

```
var packet = null
while( ( packet = plugins.udp.getReceivedPacket() ) != null)
{
    var text = packet.readUTF()
    var count = packet.readInt()
}
```

sendPacket

[Boolean](#) **sendPacket(destIpOrHostname, packet)**
Send a packet.

Parameters

`{String}` destIpOrHostname - the ip of the destination or the hostname
`{JSPacket}` packet - the JSPacket to send

Returns

`Boolean`

Sample

```
var packet = plugins.udp.createNewPacket()  
packet.writeUTF('hello world!')  
plugins.udp.sendPacket('10.0.0.1', packet)
```

sendPacket

`Boolean` **sendPacket** (destIpOrHostname, packet, port)

Send a packet on another port.

Parameters

`{String}` destIpOrHostname - the ip of the destination or the hostname
`{JSPacket}` packet - the JSPacket to send
`{Number}` port - the port on which to send the packet

Returns

`Boolean`

Sample

```
var packet = plugins.udp.createNewPacket()  
packet.writeUTF('hello world!')  
plugins.udp.sendPacket('10.0.0.1', packet, 4321)
```

startSocket

`Boolean` **startSocket** (port_number, method_to_call_when_packet_received_and_buffer_is_empty)

Start a UDP socket for a port.

Parameters

`{Number}` port_number - the local port that this UDP socket will bind to.
`{Object}` method_to_call_when_packet_received_and_buffer_is_empty - when the socket receives one or more packages, it calls this method once. The method will no longer be called even if new packages are received - until a call to `UDPPProvider#js_getReceivedPacket()` returns null. So you should consume all available packets before you expect this method to be called again.

Returns

`Boolean`

Sample

```
plugins.udp.startSocket(1234, my_packet_process_method)
```

stopSocket

`void` **stopSocket** ()

Stop the UDP socket for a port.

Returns

`void`

Sample

```
plugins.udp.stopSocket()
```

testPacket

`Boolean` **testPacket** (packet)

Put a test packet in the receive buffer to test your method call and `getReceivedPacket`.

Parameters

`{JSPacket}` packet

Returns

`Boolean`

Sample

```
var packet = plugins.udp.createNewPacket()  
packet.writeUTF('hello world!')  
plugins.udp.testPacket(packet)
```