

1. Objetivo

Comprender el funcionamiento de una cadena privada y permisionada.

Experimentar estos conceptos desplegando una cadena basada en Multichain.

2. Introducción

Multichain (<https://www.multichain.com/>) es una herramienta abierta, ligera y amigable para construir una red blockchain permisionada. Es una variante modular del blockchain de Bitcoin. Permite configurar la red con distintos mecanismos de consenso, otorgar diferentes permisos a los usuarios y almacenar datos privados, entre otros.

Esta práctica está inspirada en el [tutorial de MultiChain](#).

3. Desarrollo

3.1. Instalación

Va a trabajar en grupos de cuatro computadoras que se nombrarán WinA, WinB, WinC, y WinD.

Si no lo ha hecho, descargue el programa desde <https://www.multichain.com/download-install/> en cada una de ellas. Las computadoras pueden ser Linux, MacOS o Windows 7 en adelante. *Las computadoras del laboratorio ya tienen el programa instalado.*

Es muy importante que todas las computadoras tengan la misma versión de protocolo.

3.2 Inicio de la cadena

La cadena se crea inicialmente en WinA, por lo que esta computadora será el Administrador. La cadena se llamará **tutorialMC**.

En WinA abra una ventana de comandos y cree la cadena:

```
WinA> multichain-util create tutorialMC
```

```
Multichain 1.0.3 Utilities (latest protocol 10009)
```

```
Blockchain parameter set was successfully generated.
```

```
You can edit it in C:\Users\user01\AppData\Roaming\MultiChain\tutorialMC\params.dat  
before running multichaind for the first time.
```

```
To generate blockchain please run "multichaind tutorialMC -daemon".
```

Podemos comprobar que la cadena se creó correctamente. Ahora debe inicializarse y crear el bloque de génesis. Si el sistema operativo lo solicita, de autorización a la ejecución del programa. (*Nota: en Windows no funciona el argumento -daemon. Esta ventana se queda activa y se debe abrir otra ventana para introducir los demás comandos.*):

```
WinA> multichaind tutorialMC -daemon
```

```
Multichain 1.0.3 Daemon (latest protocol 10009)
```

```
Looking for genesis block...
```

```
Genesis block found
```

```
Other nodes can connect to this node using:
```

```
multichaind tutorialMC@192.168.83.1:9541
```

```
Listening for API requests on port 9540 (local only - see rpcallowip setting)
```

```
Node ready.
```

Inició correctamente y da como punto de conexión para esta cadena la siguiente dirección:

tutorialMC@192.168.83.1:9541

=====

Anote la dirección donde se encuentra SU servidor (es decir, la dirección de WinA

=====

3.3. Conexión desde otras máquinas

En las otras computadoras, abra una terminal y lance el comando `multichainind` con la dirección de la cadena recién creada (se muestran los comandos para *WinB*; las computadoras *WinC* y *WinD* deben hacer lo mismo).

Alternativamente, *WinC* y *WinD* pueden crear su propia cadena -por ejemplo, *tutorialMC2*- e interactuar entre sí hasta la sección 3.8).

Sustituya la dirección por la de su administrador en WinA

```
WinB> multichainind tutorialMC@192.168.83.1:9541
```

```
MultiChain Core Daemon build 1.0 beta 1 protocol 10008
```

```
Retrieving blockchain parameters from the seed node 192.168.83.1:9541 ...
```

```
Blockchain successfully initialized.
```

```
Please ask blockchain admin or user having activate permission to let you connect and/or transact:
```

```
multichain-cli tutorialMC grant 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv connect
```

```
multichain-cli tutorialMC grant 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv connect,send,receive
```

Anote los hashes de las direcciones de las computadoras. Los necesitará para que el administrador les otorgue permisos y para poder identificar las computadoras entre sí.

Como se indica en el mensaje, hay que solicitar al administrador (en WinA) que otorgue permisos para conectarse a la cadena y para mandar y recibir transacciones.

Abra otra ventana de comandos en WinA y para cada una de las otras computadoras, ejecute los siguientes comandos cambiando la dirección como corresponda:

```
WinA> multichain-cli tutorialMC grant 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv connect
```

```
{"method": "grant", "params": ["1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv", "connect"], "id": 1, "chain_name": "tutorialMC"}
```

```
1ac6e022be5779eba00649ca88c1c18f6df1137575da70da4e3f452668b16086
```

Ahora intente nuevamente conectarse. Como la cadena ya está inicializada en cada una de las computadoras, sólo se invoca lanzando el daemon.

El símbolo ‘&’ en el siguiente comando, permite lanzar el proceso en el background. Esto funciona

en Linux y MacOSX, pero no en Windows. En esas computadoras debe abrir una segunda ventana de comandos.

Si el firewall le solicita permitir el acceso, permítalo.

```
WinB> multichaind tutorialMC &

MultiChain Core Daemon build 1.0 beta 1 protocol 10008

MultiChain server starting

...

Node started
```

3.4 Algunos comandos interactivos

Para familiarizarse con la API de Multichain (que es muy similar a la del blockchain de Bitcoin), se muestran algunos comandos.

Nota. Linux y MacOSX permiten un modo interactivo. Simplemente ejecute el comando `multichain-cli tutorialMC` y aparecerá un prompt para insertar comandos.

```
WinB> multichain-cli tutorialMC listpermissions

{"method": "listpermissions", "params": [], "id": 1, "chain_name": "tutorialMC"}


[

    {

        "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
        "for" : null,
        "type" : "connect",
        "startblock" : 0,
        "endblock" : 4294967295
    },
    {

        "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
        "for" : null,
        "type" : "mine",
        "startblock" : 0,
        "endblock" : 4294967295
    }
]
```

```

    "endblock" : 4294967295
},
{
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
},
{
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : null,
    "type" : "activate",
    "startblock" : 0,
    "endblock" : 4294967295
},
...
]
WinB>

```

=====

¿Qué permisos tienen WinA, WinB, WinC y WinD?

=====

```

WinB> multichain-cli tutorialMC getpeerinfo
{"method":"getpeerinfo","params":[],"id":1,"chain_name":"tutorialMC"}

[
{
    "id" : 2,
    "addr" : "192.168.83.1:9541",
    "addrlocal" : "192.168.83.26:47662",
    "services" : "0000000000000001",
    "lastsend" : 1492565430,
    "lastrecv" : 1492565431,
    "bytessent" : 4427,
    "bytesrecv" : 24440,
    "conntime" : 1492565190,
    "pingtime" : 0.30682600,
}

```

```

    "version" : 70002,
    "subver" : "/MultiChain:0.1.0.8/",
    "handshakelocal" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "handshake" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "inbound" : false,
    "startingheight" : 61,
    "banscore" : 0,
    "synced_headers" : 61,
    "synced_blocks" : 61,
    "inflight" : [
    ],
    "whitelisted" : false
}
]
WinB>

```

La información corresponde a WinA, el peer de WinB. Si las cuatro computadoras están conectadas, podrá ver información de otros pares.

```
=====
¿Quiénes son los peers de WinB? ¿De WinD?
=====
```

```

WinB> multichain-cli tutorialMC listaddresses
{"method":"listaddresses","params":[],"id":1,"chain_name":"tutorialMC"}

[
{
    "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "ismine" : true
}
]
WinB>

```

Consultemos ahora la información desde WinA:

```

WinA> multichain-cli.exe tutorialMC listpermissions

{"method":"listpermissions","params":[],"id":1,"chain_name":"tutorialMC"}

[

```

```
{  
    "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",  
    "for" : null,  
    "type" : "connect",  
    "startblock" : 0,  
    "endblock" : 4294967295  
},  
{  
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",  
    "for" : null,  
    "type" : "mine",  
    "startblock" : 0,  
    "endblock" : 4294967295  
},  
{  
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",  
    "for" : null,  
    "type" : "admin",  
    "startblock" : 0,  
    "endblock" : 4294967295  
},  
},
```

...

```
WinA> multichain-cli.exe tutorialMC listaddresses  
{ "method": "listaddresses", "params": [], "id": 1, "chain_name": "tutorialMC" }
```

```
[  
    {  
        "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",  
        "ismine" : true  
    }  
]
```

```
WinA> multichain-cli.exe tutorialMC getpeerinfo
```

```
{ "method": "getpeerinfo", "params": [], "id": 1, "chain_name": "tutorialMC" }
```

```
[  
    {  
        "id" : 3,
```

```
"addr" : "148.205.36.46:47662",
"addrlocal" : "148.205.36.71:6779",
...
```

3.5. Intercambio de transacciones entre nodos

Para poder hacer transacciones entre usuarios, se debe empezar por crear un “activo”. El servidor que tiene los permisos para emitir activos es WinA (con dirección 1Puw...).

En él se crea el *activo1* con 1000 unidades que pueden ser subdivididas en 100 partes:

```
WinA> multichain-cli.exe tutorialMC issue 1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj act
ivo1 1000 0.01

{"method":"issue","params":[{"1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj","activo1",1000,
0.01000000],"id":1,"chain_name":"tutorialMC"}

527ada1b8a1b8f38a1c9ccd757056b5ea9ea9e102da29296d23bb5a4705bdf8c
```

Verifique que el activo está visible en los demás servidores y que su identificador es la cadena que se mostró al crearlo:

```
WinB> multichain-cli.exe listassets

{"method":"listassets","params":[],"id":1,"chain_name":"tutorialMC"}

[
{
    "name" : "activo1",
    "issuetxid" : "527ada1b8a1b8f38a1c9ccd757056b5ea9ea9e102da29296d23bb5a4705bd
f8c",
    "assetref" : "62-266-31314",
    "multiple" : 100,
    "units" : 0.01000000,
    "open" : false,
    "details" : {
    },
    "issueqty" : 1000.00000000,
    "issueraw" : 100000,
    "subscribed" : false
}
```

]

Ahora proceda a revisar el balance de activos en cada uno de los servidores

```
WinA> multichain-cli.exe tutorialMC gettotalbalances

{"method":"gettotalbalances","params":[],"id":1,"chain_name":"tutorialMC"}

[
  {
    "name" : "activo1",
    "assetref" : "62-266-31314",
    "qty" : 1000.00000000
  }
]

; XXXXXXXXXXXXXXXX En WinB: XXXXXXXXXXXXXXXXXXXXXXXX

WinB> multichain-cli tutorialMC gettotalbalances

{"method":"gettotalbalances","params":[],"id":1,"chain_name":"tutorialMC"}

[
```

WinA tiene 1000 unidades y WinB ninguna.

=====

¿Cuántos activos tienen WinC y WinD?

=====

Ahora se van a transferir 100 unidades de WinA a WinB. **Observe que la dirección para el envío es la dirección del receptor.**

```
WinA> multichain-cli.exe tutorialMC sendasset 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv
      activo1 100

{"method":"sendasset","params":["1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv","activo1",1
00],"id":1,"chain_name":"tutorialMC"}

error code: -704
```

error message:

Destination address doesn't have receive permission

Se produce un error (704) porque WinB sólo tiene permisos para conectarse a la cadena, no para enviar o recibir transacciones. WinA (el único administrador actual) debe asignar permisos -al menos para recepción. Asínelos e intente nuevamente la transferencia:

```
WinA> multichain-cli.exe tutorialMC grant 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv receive,send
```

```
{"method": "grant", "params": ["1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv", "receive,send"], "id": 1, "chain_name": "tutorialMC"}
```

```
6ea0332c2ba1307ea2303eb6a23631b1a0d67b7c258aa184d92822c9d2dd5ef3
```

```
WinA> multichain-cli.exe tutorialMC sendasset 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv activo1 100
```

```
{"method": "sendasset", "params": ["1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv", "activo1", 100], "id": 1, "chain_name": "tutorialMC"}
```

```
9a67d6b49b702fb4dfe2b87acabeb2520fdfb7b3c18fc229ee2173cb3eaf9bd8
```

```
WinA> multichain-cli.exe tutorialMC gettotalbalances
```

```
{"method": "gettotalbalances", "params": [], "id": 1, "chain_name": "tutorialMC"}
```

```
[  
 {  
   "name" : "activo1",  
   "assetref" : "62-266-31314",  
   "qty" : 900.00000000  
 }  
]
```

```
WinB> multichain-cli tutorialMC gettotalbalances
```

```
{"method": "gettotalbalances", "params": [], "id": 1, "chain_name": "tutorialMC"}
```

```
[  
 {  
   "name" : "activo1",  
 }
```

```

    "assetref" : "62-266-31314",
    "qty" : 100.00000000
}
]

```

Efectivamente, ahora WinA tiene 900 unidades y WinB 100.

Otorgue permisos a las demás computadoras y haga una transferencia a WinC y una a WinD por 150 unidades

Es interesante ver las transacciones en cada nodo para entender cómo se modificaron los balances (el último argumento indica cuántas transacciones listar, empezando desde la última. Si no se especifica, lista todas):

```

WinB> multichain-cli tutorialMC listwallettransactions 1

{
  "method": "listwallettransactions",
  "params": [
    {
      "id": 1,
      "chain_name": "tutorialMC"
    }
  ],
  "balance": {
    "amount": 0.00000000,
    "assets": [
      {
        "name": "activo1",
        "assetref": "62-266-31314",
        "qty": 100.00000000
      }
    ]
  },
  "myaddresses": [
    "1BnBpyCMEorLckTPAMF3gFkfiByh1g87SjeTqv"
  ],
  "addresses": [
    "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
  ],
  "permissions": [
  ],
  "items": [
  ],
  "data": [
  ]
}

```

```
        ],
        "confirmations" : 11,
        "blockhash" : "00fd473e2b742717af379397efd863729473efb7d7878776c8106a431000f
bc0",
        "blockindex" : 2,
        "blocktime" : 1492639516,
        "txid" : "9a67d6b49b702fb4dfe2b87acabeb2520fdfb7b3c18fc229ee2173cb3eaf9bd8",
        "valid" : true,
        "time" : 1492639509,
        "timereceived" : 1492639509
    }
]
```

WinA> multichain-cli.exe tutorialMC listwallettransactions

```
{"method":"listwallettransactions","params":[],"id":1,"chain_name":"tutorialMC"}
```

```
[
{
    "balance" : {
        "amount" : 0.00000000,
        "assets" : [
            ]
    },
    "myaddresses" : [
        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "addresses" : [
        ],
    "permissions" : [
        ],
    "items" : [
        ],
    "data" : [
        ]
}
```

```
"53504b62473045022100ea97c344936859cda74db9b4e97db0c3d20ed9d43a65dff93bbb09702544740
602205c43675904555a6f20d92ad67ddecadf45f34c8d
546fbbf89ed23a42b100022102e4b753018ecf49c9e53dfac3d68ec8e65ef0c52819b3b4150156807b2b
11b290",
],
"confirmations" : 94,
```

```
"generated" : true,
"blockhash" :
"002e27f1efdf03dd9a99bbd27536c54508b10f5dab5b03a77b7f360f1b152432",
"blockindex" : 0,
"blocktime" : 1492564149,
"txid" :
"da02e2b37a0f2755a6c797d71586c6b22147bb5fd53cfbceb66093a3212458b4",
"valid" : true,
"time" : 1492564149,
"timereceived" : 1492564149
},
{
"balance" : {
    "amount" : 0.00000000,
    "assets" : [
    ]
},
"myaddresses" : [
    "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
],
"addresses" : [
    "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
],
"permissions" : [
{
    "for" : null,
    "connect" : true,
    "send" : false,
    "receive" : false,
    "create" : false,
    "issue" : false,
    "mine" : false,
    "admin" : false,
    "activate" : false,
    "startblock" : 0,
    "endblock" : 4294967295,
    "timestamp" : 1492564869,
    "addresses" : [
        "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
    ]
}
]
```

```
        ],
        "items" : [
        ],
        "data" : [
        ],
        "confirmations" : 44,
        "blockhash" :
"00c3827dccc773d6d22d2ed0b172cc32197fa55e676accfcdeb9186ea92d6c6c",
        "blockindex" : 1,
        "blocktime" : 1492564873,
        "txid" :
"1ac6e022be5779eba00649ca88c1c18f6df1137575da70da4e3f452668b16086",
        "valid" : true,
        "time" : 1492564869,
        "timereceived" : 1492564869
},
{
    "balance" : {
        "amount" : 0.00000000,
        "assets" : [
            {
                "name" : "activo1",
                "assetref" : "62-266-31314",
                "qty" : 1000.00000000
            }
        ]
    },
    "myaddresses" : [
        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "addresses" : [
    ],
    "permissions" : [
    ],
    "issue" : {
        "name" : "activo1",
        "assetref" : "62-266-31314",
        "multiple" : 100,
        "units" : 0.01000000,
        "open" : false,
        "details" : {
```

```
        },
        "qty" : 1000.0000000,
        "raw" : 100000,
        "addresses" : [
            "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
        ]
    },
    "items" : [
    ],
    "data" : [
    ],
    "confirmations" : 33,
    "blockhash" :
    "00883f24890dda541af59537356a2e9eb7d4a028782df3fe934cd2efe94c1f48",
    "blockindex" : 1,
    "blocktime" : 1492566985,
    "txid" :
    "527ada1b8a1b8f38a1c9ccd757056b5ea9ea9e102da29296d23bb5a4705bdf8c",
    "valid" : true,
    "time" : 1492566975,
    "timereceived" : 1492566975
},
{
    "balance" : {
        "amount" : 0.0000000,
        "assets" : [
        ]
    },
    "myaddresses" : [
        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "addresses" : [
    ],
    "permissions" : [
    ],
    "items" : [
    ],
    "data" : [
    ],
    "confirmations" : 22,
    "blockhash" :
```

```
"000f203050b86d249eb0f44a265848bab8b36e1760c7b3dd0ef129502a3fadd0",
    "blockindex" : 1,
    "blocktime" : 1492638982,
    "txid" :
"b4041a13d7588f5117f04b50072434449f23ad12b8fd95aaaf7881d106ad18113",
    "valid" : true,
    "time" : 1492638971,
    "timereceived" : 1492638971
},
{
    "balance" : {
        "amount" : 0.00000000,
        "assets" : [
        ]
    },
    "myaddresses" : [
        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "addresses" : [
        "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
    ],
    "permissions" : [
        {
            "for" : null,
            "connect" : false,
            "send" : true,
            "receive" : true,
            "create" : false,
            "issue" : false,
            "mine" : false,
            "admin" : false,
            "activate" : false,
            "startblock" : 0,
            "endblock" : 4294967295,
            "timestamp" : 1492639503,
            "addresses" : [
                "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
            ]
        }
    ],
    "items" : [
```

```
],
  "data" : [
    ],
    "confirmations" : 11,
    "blockhash" :
"00fd473e2b742717af379397efd863729473efb7d7878776c8106a431000fbc0",
    "blockindex" : 1,
    "blocktime" : 1492639516,
    "txid" :
"6ea0332c2ba1307ea2303eb6a23631b1a0d67b7c258aa184d92822c9d2dd5ef3",
    "valid" : true,
    "time" : 1492639503,
    "timereceived" : 1492639503
},
{
  "balance" : {
    "amount" : 0.00000000,
    "assets" : [
      {
        "name" : "activo1",
        "assetref" : "62-266-31314",
        "qty" : -100.00000000
      }
    ]
  },
  "myaddresses" : [
    "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
  ],
  "addresses" : [
    "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
  ],
  "permissions" : [
    ],
    "items" : [
    ],
    "data" : [
    ],
    "confirmations" : 11,
    "blockhash" :
"00fd473e2b742717af379397efd863729473efb7d7878776c8106a431000fbc0",
    "blockindex" : 2,
```

```

    "blocktime" : 1492639516,
    "txid" :
"9a67d6b49b702fb4dfe2b87acabeb2520fdfb7b3c18fc229ee2173cb3eaf9bd8",
    "valid" : true,
    "time" : 1492639508,
    "timereceived" : 1492639508
}
]

```

Más adelante iremos tratando de entender mejor toda esta información, pero parece claro que en la transacción se agregaron dos bloques a la cadena

Haga una transacción de WinD a WinC por 50 unidades y analice en cada una de ellas la lista de transacciones.

3.6. Metadata en transacciones

Multichain permite enviar metadatos junto con las transacciones. Envíe una transacción de WinA a WinB con 50 unidades (y de WinC a WinD si lo desea) y algunos metadatos (*Nota: la notación en Windows es distinta a la usada en linux porque se deben incluir diagonales invertidas como secuencias de escape*).

```

WinA> multichain-cli.exe tutorialMC sendwithdata 1BnBpyCMEorLckTPAMf3gFkfiByh1g87Sje
Tqv "{\"activo1\":50} 123456abcdef

{"method":"sendwithdata","params": ["1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv", {"activo
1":50}, "123456abcdef"], "id":1, "chain_name": "tutorialMC"}

549997ba9fdf7a5d8aecfc08232adf774a7fb3b8f1d674579c03b6a923433f7b

```

Con el ID de la transacción es posible consultar la transacción y la metadata:

```

WinB> multichain-cli tutorialMC getwallettransaction 549997ba9fdf7a5d8aecfc08232adf7
74a7fb3b8f1d674579c03b6a923433f7b

{"method":"getwallettransaction","params": ["549997ba9fdf7a5d8aecfc08232adf774a7fb3b8
f1d674579c03b6a923433f7b"], "id":1, "chain_name": "tutorialMC"}

```

```
{
    "balance" : {
        "amount" : 0.00000000,
        "assets" : [
            {
                "name" : "activo1",
                "assetref" : "62-266-31314",
                "qty" : 50.00000000
            }
        ]
    },
    "myaddresses" : [
        "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
    ],
    "addresses" : [
        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "permissions" : [
    ],
    "items" : [
    ],
    "data" : [
        "123456abcdef"
    ],
    "confirmations" : 11,
    "blockhash" : "00b1871f3921cb6e3f09510bd4c1d10a70cd0154c98bbcb4ca96169789d0234a"
    ,
    "blockindex" : 1,
    "blocktime" : 1492640477,
    "txid" : "549997ba9fdf7a5d8aecfc08232adf774a7fb3b8f1d674579c03b6a923433f7b",
    "valid" : true,
    "time" : 1492640458,
    "timereceived" : 1492640458
}
}
```

En el campo *balance* se comprueba la transferencia de 50 unidades del *activo1* y en el campo *data* se ve la metadata que se agregó a la transacción.

Si no tiene el Id de la transacción (por ejemplo, porque está en la computadora receptora), puede ejecutar el comando `listwallettransactions` para obtenerlo.

=====

Intercambie activos con datos entre WinA y WInD, y WinB y WinC.

Asegúrese de tener suficientes fondos.

=====

3.7. Streams

MultiChain introduce el concepto de **Streams** como una abstracción que permite usar blockchain para almacenar, recuperar y vender en tiempo (asignar un *timestamp*) datos. Con ellos, blockchain se puede usar para implementar tres tipos de bases de datos:

- Una base de datos de tuplas *{key-value}* (“el **qué**”)
- Una base de datos en la que las entradas están ordenadas temporalmente (“el **cuándo**”)
- Una base de datos basada en identidades en la que las entradas están clasificadas con base en su autor (“el **quién**”).

Los ítems en un `stream` sólo pueden agregarse (no se pueden modificar ni borrar). Cada ítem tiene las siguientes características:

- Tiene uno o más **publishers** que han firmado digitalmente ese ítem
- Una **key** opcional para facilitar su recuperación posterior
- **Datos** que pueden ser desde un pequeño fragmento de texto hasta varios MB de bytes binarios
- Un **timestamp** tomado del encabezado del bloque en el que se confirmó el ítem.

Cree un `stream` en WinA con el siguiente comando:

```
WinA> multichain-cli.exe tutorialMC create stream flujo1 false

{"method": "create", "params": ["stream", "flujo1", false], "id": 1, "chain_name": "tutorialMC" }

ca13900df9a34a04a725280ce665f64c59e8d7596a4228aaa73e576a3e273815
```

La opción *false* indica que en este stream sólo pueden agregar ítems quienes tienen permiso para ello. Veamos qué permisos tiene el stream:

```
WinA> multichain-cli.exe tutorialMC listpermissions flujo1.*

{"method": "listpermissions", "params": ["flujo1.*"], "id": 1, "chain_name": "tutorialMC" }
```

```
[
  {
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : {
      "type" : "stream",
      "name" : "flujo1",
      "streamref" : "117-265-5066"
    },
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
  },
  {
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : {
      "type" : "stream",
      "name" : "flujo1",
      "streamref" : "117-265-5066"
    },
    "type" : "activate",
    "startblock" : 0,
    "endblock" : 4294967295
  },
  {
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : {
      "type" : "stream",
      "name" : "flujo1",
      "streamref" : "117-265-5066"
    },
    "type" : "write",
    "startblock" : 0,
    "endblock" : 4294967295
  }
]
```

Por ahora sólo WinA tiene permisos de escritura. Desde esa computadora, publique algunos datos identificados con la llave key1 (**Nota. Los datos deben ser valores hexadecimales**):

```
WinA> multichain-cli.exe tutorialMC publish flujo1 key1 123456
```

```
{"method": "publish", "params": ["flujo1", "key1", "123456"], "id": 1, "chain_name": "tutorialMC"}
```

69731305396c13285a49d37556d975f08845d0938d0c9481539b6e0abc91d8a8

Lo que se recibe es el ID de la “transacción”.

Compruebe si este flujo es visible desde los demás servidores

```
WinB> multichain-cli tutorialMC liststreams
```

```
{"method": "liststreams", "params": [], "id": 1, "chain_name": "tutorialMC"}
```

```
[  
 {  
     "name" : "root",  
     "createtxid" : "b2e819e6da16ec7ec4707f2c03313bd8b1048d0b3e85e10b45f92541e65d  
22b2",  
     "streamref" : "0-0-0",  
     "open" : true,  
     "details" : {  
     },  
     "subscribed" : true,  
     "synchronized" : true,  
     "items" : 0,  
     "confirmed" : 0,  
     "keys" : 0,  
     "publishers" : 0  
 },  
 {  
     "name" : "flujo1",  
     "createtxid" : "ca13900df9a34a04a725280ce665f64c59e8d7596a4228aaa73e576a3e27  
3815",  
     "streamref" : "117-265-5066",  
     "open" : false,  
     "details" : {  
     },  
     "subscribed" : false  
 }  
 ]
```

Sí, y también hay un flujo llamado *root* que fue creado por default.

WinB o cualquier otra computadora se puede suscribir a este flujo y hasta puede consultar ítems en él:

```
WinB> multichain-cli tutorialMC subscribe flujo1

{"method": "subscribe", "params": ["flujo1"], "id": 1, "chain_name": "tutorialMC"}


WinB>multichain-cli tutorialMC liststreamitems flujo1

{"method": "liststreamitems", "params": ["flujo1"], "id": 1, "chain_name": "tutorialMC"}


[

    {

        "publishers" : [
            "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
        ],
        "key" : "key1",
        "data" : "123456",
        "confirmations" : 9,
        "blocktime" : 1492641625,
        "txid" : "69731305396c13285a49d37556d975f08845d0938d0c9481539b6e0abc91d8a8"
    }
]
```

Suscriba todos los servidores al flujo ... pero todavía no tienen permisos para publicar en el flujo. Estos permisos deben otorgarse desde el administrador. Se necesitan permisos tanto para enviar y recibir al blockchain como para escribir en el flujo. Esto se hace así:

```
WinA> multichain-cli.exe tutorialMC grant 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv flujo1.write

{"method": "grant", "params": [
    "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv", "flujo1.write"
], "id": 1, "chain_name": "tutorialMC"}
```

86904e23f866de45f1a60128d936c1c8194e2912a3165255921126f3363937cd

Con los permisos otorgados, ya es posible agregar ítems al flujo desde WinB. Agregue dos ítems desde esa computadora:

```
WinB> multichain-cli tutorialMC publish flujo1 key1 11335577

{"method":"publish","params":["flujo1","key1","11335577"],"id":1,"chain_name":"tutorialMC"}
```

5a4e63ced5bf3407754c6588b2b18374a40f7cbb6f559ae488d3d6be69cb9327

```
WinB> multichain-cli tutorialMC publish flujo1 key2 22446688

{"method":"publish","params":["flujo1","key2","22446688"],"id":1,"chain_name":"tutorialMC"}
```

f5a908103863357bbb426aff7e1556c1d216c89e06831e11170fd459ee316f73

Para finalizar, se mostrará cómo consultar la base de datos creada en este flujo desde WinA, de varias maneras. Para ello, primero hay que suscribirse al flujo.

```
WinA> multichain-cli.exe tutorialMC subscribe flujo1

{"method":"subscribe","params":["flujo1"],"id":1,"chain_name":"tutorialMC"}
```

;¿Qué items hay? (Tres)

```
WinA> multichain-cli.exe tutorialMC liststreamitems flujo1

{"method":"liststreamitems","params":["flujo1"],"id":1,"chain_name":"tutorialMC"}
[
```

{

```
    "publishers" : [
        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "key" : "key1",
    "data" : "123456",
    "confirmations" : 38,
    "blocktime" : 1492641625,
    "txid" :
"69731305396c13285a49d37556d975f08845d0938d0c9481539b6e0abc91d8a8"
```

},

{

```
    "publishers" : [
        "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
```

```

        ],
        "key" : "key1",
        "data" : "11335577",
        "confirmations" : 5,
        "blocktime" : 1492642258,
        "txid" :

"5a4e63ced5bf3407754c6588b2b18374a40f7cbb6f559ae488d3d6be69cb9327"

},
{
    "publishers" : [
        "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
    ],
    "key" : "key2",
    "data" : "22446688",
    "confirmations" : 5,
    "blocktime" : 1492642258,
    "txid" :

"f5a908103863357bbb426aff7e1556c1d216c89e06831e11170fd459ee316f73"
}
]

```

;¿Cuántas llaves hay? (son dos)

WinA> multichain-cli.exe tutorialMC liststreamkeys flujo1

```
{"method":"liststreamkeys","params":["flujo1"],"id" 1,"chain_name":"tutorialMC"}
```

```

[
{
    "key" : "key1",
    "items" : 2,
    "confirmed" : 2
},
{
    "key" : "key2",
    "items" : 1,
    "confirmed" : 1
}
]
```

;¿Qué ítems están asociados a la llave key1? (dos)

WinA> multichain-cli.exe tutorialMC liststreamkeyitems flujo1 key1

```
{"method":"liststreamkeyitems","params":["flujo1","key1"],"id":1,"chain_name":"tutorialMC"}
```

[

```
{
  "publishers" : [
    "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
  ],
  "key" : "key1",
  "data" : "123456",
  "confirmations" : 43,
  "blocktime" : 1492641625,
  "txid" :
"69731305396c13285a49d37556d975f08845d0938d0c9481539b6e0abc91d8a8"
},
```

```
{
  "publishers" : [
    "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv"
  ],
  "key" : "key1",
  "data" : "11335577",
  "confirmations" : 10,
  "blocktime" : 1492642258,
  "txid" :
"5a4e63ced5bf3407754c6588b2b18374a40f7ccb6f559ae488d3d6be69cb9327"
}
```

]

;¿Quiénes han publicado en este flujo? (dos)

```
WinA> multichain-cli.exe tutorialMC liststremepublishers flujo1
```

```
{"method":"liststremepublishers","params":["flujo1"],"id":1,"chain_name":"tutorialMC"}
```

[

```
{
  "publisher" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
  "items" : 1,
  "confirmed" : 1
},
```

```

{
    "publisher" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "items" : 2,
    "confirmed" : 2
}
]

```

=====
Publique dos items desde las computadoras restantes y ejecute los comandos anteriores para observar que se publicaron con éxito
=====

3.8. Permisos y consensos

Multichain agrega el parámetro de **consenso** para administrar los permisos que se van otorgando a los nodos.

Los parámetros de consenso únicamente pueden ser modificados al momento de crear la cadena o antes de ejecutar el comando `multichaind (nombre cadena) -daemon` por primera vez.

En Multichain, los permisos se dividen en tres categorías: bajo, medio y alto riesgo.

- Los de bajo riesgo son **connect**, **send** y **receive**. Los usuarios a los que se les otorgan estos permisos únicamente pueden conectarse al blockchain o realizar transacciones propias.
- Los de medio riesgo son **issue**, **create** y **activate**. Los usuarios que reciben estos permisos pueden emitir activos, crear flujos y manejar permisos de bajo riesgo de otros usuarios.
- Los de alto riesgo son **mine** y **admin**. Los nodos con permiso **mine** participan en el algoritmo de consenso, y los que cuentan con el permiso **admin** pueden modificar los permisos de todos los usuarios.

Los parámetros que se definen por omisión en el blockchain pueden consultarse con el siguiente comando:

```

WinA> multichain-cli.exe tutorialMC getblockchainparams

{
    "method": "getblockchainparams", "params": [], "id": 1, "chain_name": "tutorialMC"
}

{
    "chain-protocol" : "multichain",

```

```
"chain-description" : "MultiChain tutorialMC",
"root-stream-name" : "root",
"root-stream-open" : true,
"chain-is-testnet" : false,
"target-block-time" : 15,
"maximum-block-size" : 8388608,
"default-network-port" : 2779,
"default-rpc-port" : 2778,
"anyone-can-connect" : false,
"anyone-can-send" : false,
"anyone-can-receive" : false,
"anyone-can-receive-empty" : true,
"anyone-can-create" : false,
"anyone-can-issue" : false,
"anyone-can-mine" : false,
"anyone-can-activate" : false,
"anyone-can-admin" : false,
"support-miner-precheck" : true,
"allow-arbitrary-outputs" : false,
"allow-p2sh-outputs" : true,
"allow-multisig-outputs" : true,
"setup-first-blocks" : 60,
"mining-diversity" : 0.30000000,
"admin-consensus-upgrade" : 0.50000000,
"admin-consensus-admin" : 0.50000000,
"admin-consensus-activate" : 0.50000000,
"admin-consensus-mine" : 0.50000000,
"admin-consensus-create" : 0.00000000,
"admin-consensus-issue" : 0.00000000,
"lock-admin-mine-rounds" : 10,
"mining-requires-peers" : true,
"mine-empty-rounds" : 10.00000000,
"mining-turnover" : 0.50000000,
"first-block-reward" : -1,
"initial-block-reward" : 0,
"reward-halving-interval" : 52560000,
"reward-spendable-delay" : 1,
"minimum-per-output" : 0,
"maximum-per-output" : 1000000000000000,
"minimum-relay-fee" : 0,
"native-currency-multiple" : 1000000000,
```

```

"skip-pow-check" : false,
"pow-minimum-bits" : 8,
"target-adjust-freq" : -1,
"allow-min-difficulty-blocks" : false,
"only-accept-std-txs" : true,
"max-std-tx-size" : 4194304,
"max-std-op-returns-count" : 10,
"max-std-op-return-size" : 2097152,
"max-std-op-drops-count" : 5,
"max-std-element-size" : 8192,
"chain-name" : "tutorialMC",
"protocol-version" : 10009,
"network-message-start" : "f9f2f7f7",
"address-pubkeyhash-version" : "005c92cf",
"address-scripthash-version" : "05deba6f",
"private-key-version" : "80bc3b13",
"address-checksum-value" : "69b3c3e6",
"genesis-pubkey" : "03f72e3e2d9202684281e348f71794dd2687cbbb1ab7c5ab567b31194328
edd707",
"genesis-version" : 1,
"genesis-timestamp" : 1518550995,
"genesis-nbits" : 536936447,
"genesis-nonce" : 170,
"genesis-pubkey-hash" : "a770ade9dc20f42fe073795e914249467db76719",
"genesis-hash" : "00f607f0d554385b51319b57259104c2b3c67dab965743d3b4dc769704d607
75",
"chain-params-hash" : "03a3595bb9b10b1b7d5aab2ca83a90747fe18c16d9af45a9228310d41
e57ec8f"
}

```

Observe el parámetro `admin-consensus-admin`. Su valor (en el ejemplo, 0.50000000) al ser multiplicado por el número de administradores (redondeado en función techo) indica la cantidad de administradores que deben otorgar un permiso para que éste sea validado. Puede entenderse este valor como si se requiere del consenso del 50% de administradores para otorgar un permiso.

Empiece por otorgar permisos de administrador a WinB y observe en la lista de permisos qué computadoras son administradores:

```

WinA> multichain-cli tutorialMC grant 1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv admin
{"method":"grant","params":["1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv","admin"],"id":1

```

```

,"chain_name":"tutorialMC"}
```

ecaе215a31fa1e1b040ea0693a7923d8ebdab61631aef2586479bee79d5b9e8

```

WinA> multichain-cli.exe tutorialMC listpermissions admin
{"method":"listpermissions","params":["admin"],"id":1,"chain_name":"tutorialMC"}
```

```

[
  {
    "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
  },
  {
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
  }
]
```

Son curiosos los parámetros `startblock` y `endblock`. Multichain permite otorgar permisos de administrador para sólo un rango de bloques cadena si lo desea.

Ejecute el mismo comando para WinC

En este punto tenemos tres nodos que cuentan con permisos de administrador.

```

WinA> multichain-cli.exe tutorialMC listpermissions admin
{"method":"listpermissions","params":["admin"],"id":1,"chain_name":"tutorialMC"}
```

```

[
  {
    "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
  }
]
```

```

},
{
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
},
{
    "address" : "1Z43MPRZGSCmV6Pn7bw6dUJzZaYRq3zzpMT4hz",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
}
]

```

Desde winA, ahora intente otorgar el permiso de administrador a winD.

```

WinA> multichain-cli tutorialMC grant 17mnRm3fnxnA2PNNqmiy3Sw1LgAHz4L37kGTRw admin

{"method":"grant","params":[{"17mnRm3fnxnA2PNNqmiy3Sw1LgAHz4L37kGTRw","admin"],"id":1,"chain_name":"tutorialMC"}

471df8a25f20921dbfb7bb24cc34548a6ad9d833e2bc47bb7c5b164026664460

WinA> multichain-cli.exe tutorialMC listpermissions admin

{"method":"listpermissions","params":["admin"],"id":1,"chain_name":"tutorialMC"}


[
{
    "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
},
{
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : null,

```

```

    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
},
{
    "address" : "1Z43MPRZGSCmV6Pn7bw6dUJzZaYRq3zzpMT4hZ",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
}
]

```

winD aún no aparece en la lista de administradores. Con tres administradores y un valor admin-consensus-admin=0.5 , se requiere de al menos un administrador adicional que otorgue ese permiso.

Con el siguiente comando, que es una petición más detallada de `listpermissions` , podrá comprobar en el campo “pending” que efectivamente se requiere aún de un administrador para validar el permiso.

```

WinA> multichain-cli.exe tutorialMC listpermissions admin 17mnRm3fnxnA2PNNqmiy3Sw1Lg
AHz4L37kGTRw true

{
    "method": "listpermissions", "params": [ "admin", "17mnRm3fnxnA2PNNqmiy3Sw1LgAHz4L37kGTRw", true ], "id": 1, "chain_name": "tutorialMC" }

[
{
    "address" : "17mnRm3fnxnA2PNNqmiy3Sw1LgAHz4L37kGTRw",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 0,
    "admins" : [
    ],
    "pending" : [
        {
            "startblock" : 0,
            "endblock" : 4294967295,
            "admins" : [

```

```

        "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj"
    ],
    "required" : 1
}
]
}
]
```

Para comprobar lo anterior, otorgue el permiso de administrador a WinD desde WinB o WinC

```

WinB> multichain-cli tutorialMC grant 17mnRm3fnxnA2PNNqmiy3Sw1LgAHZ4L37kGTRw admin
{"method":"grant","params":[{"17mnRm3fnxnA2PNNqmiy3Sw1LgAHZ4L37kGTRw","admin"],"id":1,"chain_name":"tutorialMC"}

ae208e5bc22b3a12404b132b43c0f1981d5b201d0be4862c29fdbd9551c0f03ff

```

Ahora podrá comprobar que WinD ya es también administrador:

```

WinA> multichain-cli.exe tutorialMC listpermissions admin
{"method":"listpermissions","params":["admin"],"id":1,"chain_name":"tutorialMC"}
```

```

[
{
    "address" : "1BnBpyCMEorLckTPAMf3gFkfiByh1g87SjeTqv",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
},
{
    "address" : "1Puw2KnxCqJ6eSh5oXLV7nmMD8Wfy1YuCutMgj",
    "for" : null,
    "type" : "admin",
    "startblock" : 0,
    "endblock" : 4294967295
},
{
    "address" : "1Z43MPRZGSCmV6Pn7bw6dUJzZaYRq3zzpMT4hZ",
    "for" : null,
    "type" : "admin",
}
```

```
        "startblock" : 0,
        "endblock" : 4294967295
    },
    {
        "address" : "17mnRm3fnxnA2PNNqmiy3Sw1LgAHz4L37kGTRw",
        "for" : null,
        "type" : "admin",
        "startblock" : 0,
        "endblock" : 4294967295
    }
]
```

Ahora puede practicar otorgando otro tipo de permisos a los nodos y hasta retirando el permiso de administrador a WinA.

Felicidades. Ahora tiene una buena perspectiva de blockchains privados y permisionados.