

# PDX UTOPIA 协议栈 部署手册

版本 2.1.0

# 目 录

1 PDX Utopia 简介.....	1
2 要求.....	1
3 架构.....	2
3.1 Utopia 独立部署.....	2
3.2 Utopia 联合 PDX BaaP 部署.....	2
3.3 Utopia 联合 PDX Unity 部署.....	3
4 独立部署.....	3
4.1 创建公链实例.....	4
4.2 创建私链实例.....	4
4.3 创建联盟链实例.....	6
4.4 启动区块链实例.....	9
4.5 停止区块链实例.....	10
4.6 删除区块链实例.....	10
5 Utopia 联合 PDX BaaP 部署.....	10
5.1 配置 PDX BaaP.....	10
5.2 集成 PDX Unity.....	10
5.3 启动 PDX Utopia + BaaP.....	11
5.4 停止 PDX Utopia + BaaP.....	11
6 Utopia 联合 PDX Unity 部署.....	11
6.1 启动 node.....	11
6.2 停止 node.....	11
6.3 更新 node.....	11
6.4 清除 node.....	11
7 附录 A: 生成 X509 v3 证书.....	11
7.1 创建 Root-CA 证书密钥对.....	12
7.2 创建 Root-CA 证书请求.....	12
7.3 通过自签名创建 Root-CA 证书.....	12
7.4 创建一个中间 Node-CA.....	12
7.5 创建一个节点证书.....	12

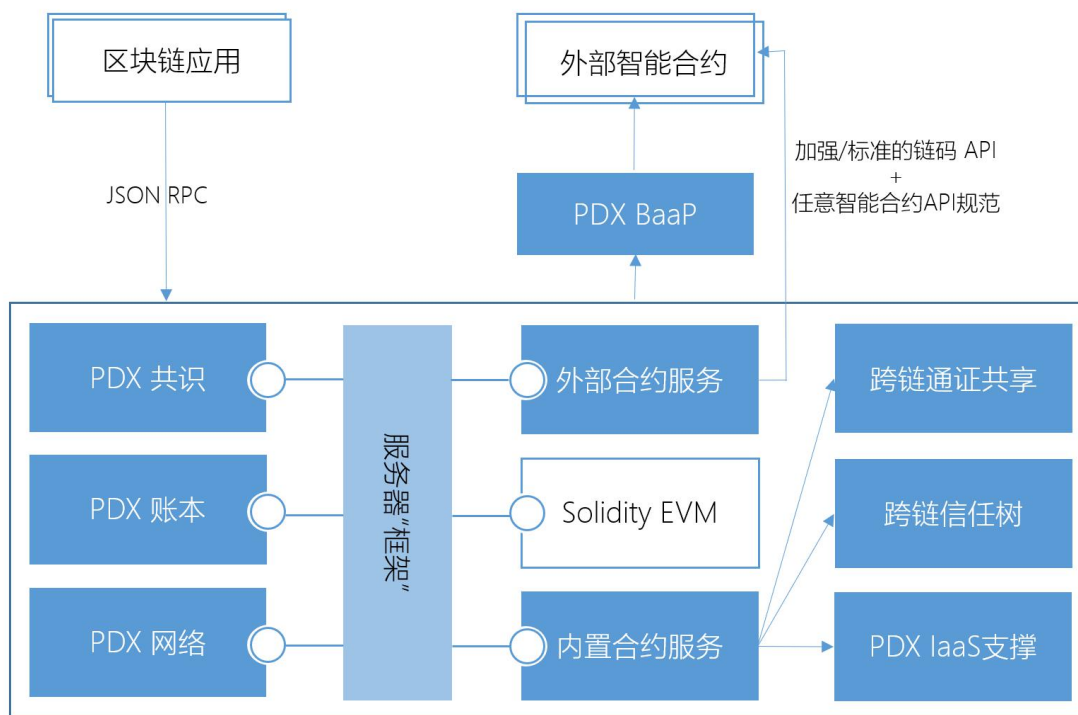
## 1 PDX Utopia 简介

PDX Utopia 区块链协议栈，基于专利保护的 PDX 共识算法、PDX 账本算法、PDX 智能合约架构，在安全、公平和多数共识的前提下，实现大规模、低延迟、高并发、高吞吐。PDX 共识算法，采用独创的双账本、非阻塞架构，在安全、公平、多数共识的前提下使区块确认的复杂度降到  $O(n)$ 。从算法上确保 PDX Utopia 区块链能够真正做到大规模、低延迟、高并发、高吞吐。PDX 共识的算法规范，可以从以下地址下载：  
<https://github.com/PDXbaap/design/blob/master/pdx-consensus.pdf>。

PDX Utopia 区块链协议栈，内生兼容以太坊的 Solidity 合约、以太坊钱包、ERC20、eWASM 和超级账本 chaincode 合约，并可扩展兼容未来的智能合约 API 接口规范。

PDX Utopia 内置的“跨链通证共享”智能合约，支持性能真正无损的、安全的跨链通证共享，方便形成多链跨链的激励生态。

PDX Utopia 内置的“跨链信任树”和“PDX IaaS 支撑”智能合约，实现跨链的信任背书以及与 PDX SureDigit 可信数字生态平台的融合。



## 2 要求

X64 硬件体系结构上的 Linux 4.4 或更高版本，具有至少 16GB RAM、500GB 磁盘和 100MB 网络带宽。此外，节点必须开放其 TCP 端口 30303（入网和出网），如果使用了 PDX Netmux（见下文）节点必须开放其 TCP 端口 5978（入网和出网），以便与其他区块链节点进行双向通信。

TCP 端口 8545（用于事务执行和查询的 json RPC）和 8546（用于区块链事件通知的 websocket）的打开是可选的，并且可以仅限于来自授权 TCP 地址空间的客户端。

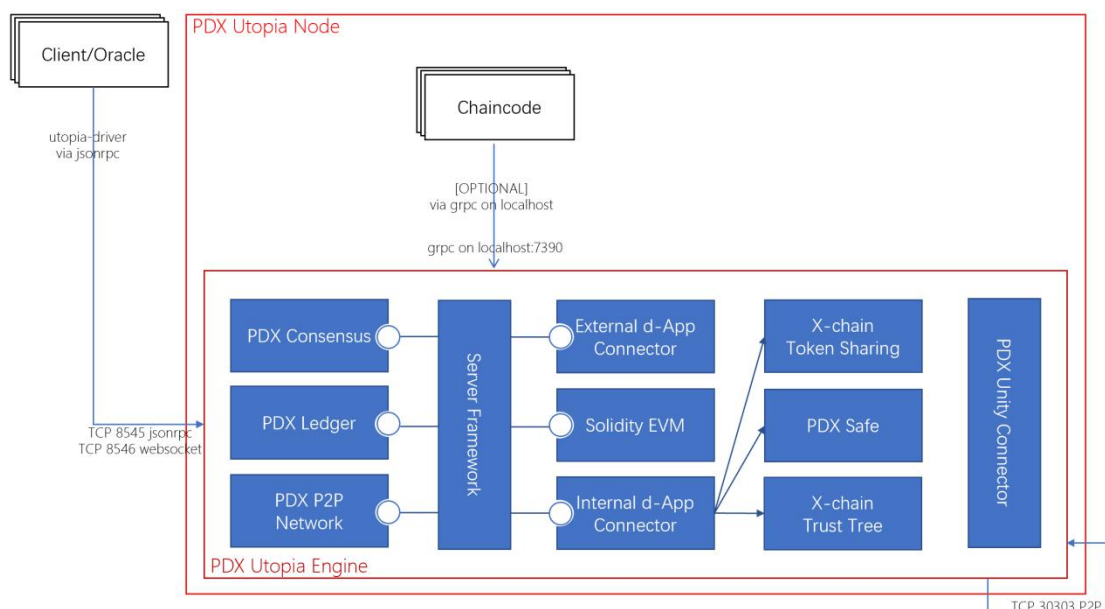
PDX Utopia 于出块时间有校验处理，请确保本地时间的准确性，从而可以正常参与 Utopia 共识。更新本地机器时间，可执行该命令 `sudo ntp -s pool.ntp.org`。

## 3 架构

PDX Utopia 区块链可以与 PDX BaaP 多链 d-App 容器一起部署，也可以与 PDX BaaP 和 PDX Unity 区块链协调平台一起独立部署。每个部署都有其特点，可以满足特定的部署要求。

### 3.1 Utopia 独立部署

这种部署模式适合于 PDX Utopia 区块链引擎的简单试用。在这种模式下，必须将 chaincode d-Apps 手动复制到每个区块链节点，并执行以连接到该节点上对应的 PDX Utopia 协议栈。虽然它完全支持基于 Solidity 的 d-App 的开发和部署，但不支持通过 UtopiaChaincodeDriver.deploy API 部署链码 d-Apps。除此之外，它是功能齐全的 PDX Utopia 区块链部署。



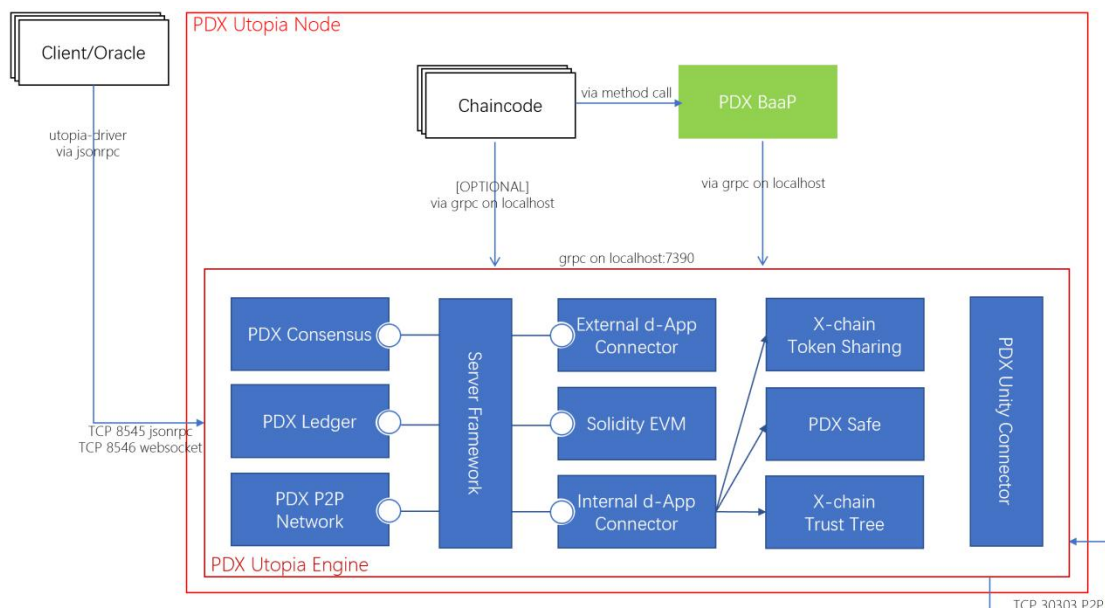
### 3.2 Utopia 联合 PDX BaaP 部署

使用 PDX BaaP 部署 Utopia 既可以手动完成，也可以作为 docker 镜像完成。如果手动进行，则必须先启动所有 PDX Utopia 区块链实例，然后再启动 PDX BaaP。

使用 PDX BaaP，可以自动部署和启动 chaincode d-App。此外，它可以部署在多个区块链实例上。

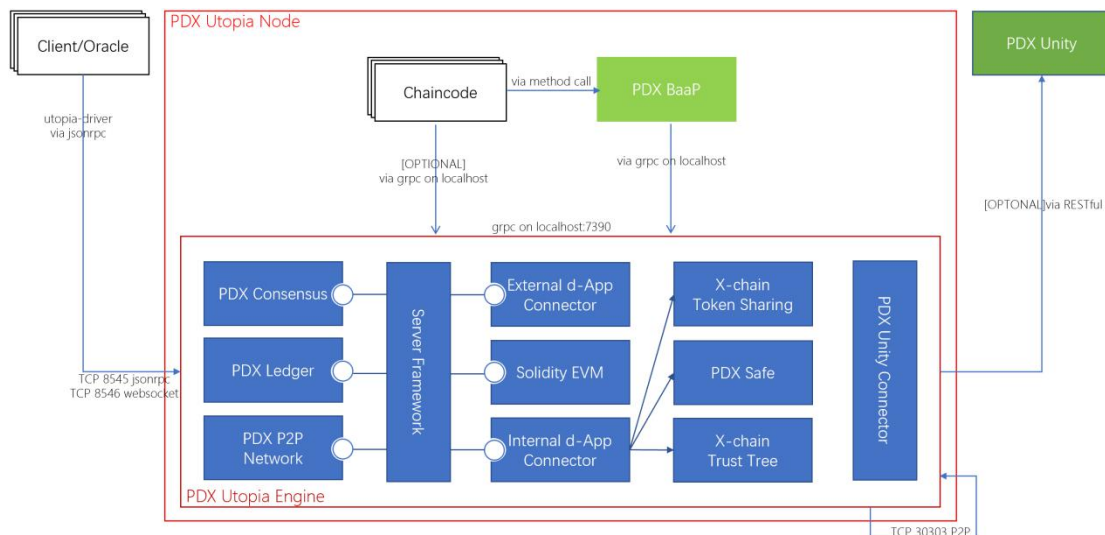
当在一个节点上部署多个区块链实例时，每个链的端口不能互相冲突。必须注意为每个要在节点上部署的区块链实例手动选择未使用的端口。

为了简化防火墙和连接性配置，可以使用 PDX Netmux（未展示），以便所有通信都可以覆盖在一个面向外部的 TCP 端口上。有关 PDX Netmux 的详细信息，请查看 <https://github.com/PDXbaap/netmux>。



### 3.3 Utopia 联合 PDX Unity 部署

使用 PDX Unity 部署 Utopia 是作为 docker 镜像完成的。不仅可以自动部署和启动 chaincode d-App，而且可以在参与节点上自动部署多个区块链实例。在这种模式下，PDX Utopia 节点会自动为节点上的每个区块链实例选择不冲突的 TCP 端口，并配置 PDX Netmux，以便所有通信都可以共用一个 TCP 端口。有关 PDX Netmux 的详细信息，请查看 <https://github.com/PDXbaap/netmux>。



## 4 独立部署

在独立部署 PDX Utopia 的方式中，需要从 <https://pdx.ltd/download> 下载 PDX Utopia 安装包。下载完成后解压到 \$UTOPIA\_HOME 目录下，例如 /opt/pdx/utopia。

## 4.1 创建公链实例

### 创建公链

创建一个公链实例，需要按如下所示运行\$UTOPIA\_HOME/bin/init.sh 脚本：

```
jz@matebook:/opt/pdx/utopia/bin$ ./init.sh
1) create a new blockchain
2) join an existing blockchain
Please pick an option: 1
Enter the blockchain id to create [739]:
Enter your password:
INFO [05-25|20:24:48.002] PDX Utopia Version222:=1.9
.0
INFO [05-25|20:24:48.002] Maximum peer count PDX=25 LES=0 to
tal=25
Your new account is locked with a password. Please give a password. Do not forget
this password.
Passphrase:
Repeat passphrase:
Address: {9f27f39354a01b1caed00c8a4b2ad8f68c36b9e1}
INFO [05-25|20:24:50.635] PDX Utopia Version222:=1.9
.0
INFO [05-25|20:24:50.635] Maximum peer count PDX=25 LES=0 to
tal=25
Enter the reward address :[ default is miner address: 9f27f39354a01b1caed00c8a4b2a
d8f68c36b9e1 ]

!!! Public blockchain 739: ready to start now !!!
```

### 生成目录

安装过程将会创建以下目录：

\$UTOPIA_HOME/chain/\$CHAIN_ID	存储区块链运行时数据，例如：账本。
\$UTOPIA_HOME/conf/\$CHAIN_ID	存储启动配置信息或 PDX baap 及其它 PDX 子系统的配置信息。
\$UTOPIA_HOME/temp/\$CHAIN_ID	存储运行时日志及进程 pid 文件。

## 4.2 创建私链实例

### 创建私链

创建一个私链实例，需要按如下所示运行\$UTOPIA\_HOME/bin/init.sh 脚本



```
jz@matebook:/opt/pdx/utopia/bin$ ./init.sh
1) create a new blockchain
2) join an existing blockchain
Please pick an option: 1
Enter the blockchain id to create [739]:123
1) public blockchain
2) consortium blockchain
Please pick an option: 1
Enter your password:
INFO [05-26|10:59:23.255] PDX Utopia Version222:=1.9.0
INFO [05-26|10:59:23.256] Maximum peer count PDX=25 LES=0 total=25
Your new account is locked with a password. Please give a password. Do not forget this password.
Passphrase:
Repeat passphrase:
[Address: {523d8de2dcdb763341b9acbfbb22edbacb1ca822}
[INFO [05-26|10:59:25.423] PDX Utopia Version222:=1.9.0
[INFO [05-26|10:59:25.423] Maximum peer count PDX=25 LES=0 total=25
Enter the reward address :[ default is miner address: 523d8de2dcdb763341b9acbfbb22edbacb1ca822 ]
Enter the FirstMinerAddress :[ default is miner address: 523d8de2dcdb763341b9acbfbb22edbacb1ca822 ]:
INFO [05-26|10:59:27.389] PDX Utopia Version222:=1.9.0
INFO [05-26|10:59:27.389] Maximum peer count PDX=25 LES=0 total=25
INFO [05-26|10:59:27.400] Allocated cache and file handles database=/Users/wuxinyang/Desktop/utopia/chain/123/data/utopia/chaindata cache=16.78mB handles=16
INFO [05-26|10:59:27.446] Writing custom genesis block
INFO [05-26|10:59:27.453] Persisted trie from memory database nodes=34 size=5.15kB time=459.491µs gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [05-26|10:59:27.454] Successfully wrote genesis state database=chaindata hash=99b024...fae62c
INFO [05-26|10:59:27.454] Allocated cache and file handles database=/Users/wuxinyang/Desktop/utopia/chain/123/data/utopia/lightchaindata cache=16.78mB handles=16
INFO [05-26|10:59:27.490] Writing custom genesis block
INFO [05-26|10:59:27.492] Persisted trie from memory database nodes=34 size=5.15kB time=980.191µs gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [05-26|10:59:27.492] Successfully wrote genesis state database=lightchaindata hash=99b024...fae62c

!!! Public blockchain 123: ready to start now !!!
```

## 生成目录

安装过程产生的目录同创建公链实例产生的目录一致。

## 加入私链

要加入一个私链，需要按如下所示运行\$UTOPIA\_HOME/bin/init.sh 脚本：

```
jz@matebook:/opt/pdx/utopia/bin$ ./init.sh
1) create a new blockchain
2) join an existing blockchain
Please pick an option: 2
Enter the blockchain id to join [739]:123
1) public blockchain
2) consortium blockchain
Please pick an option: 1
Enter an bootnode info:/ip4/127.0.0.1/mux/5978:12181/ipfs/16Uiu2HAmDc6gaSW93baPTfaPHenNus6
b5SvvjeFNoPS5rYgWT1Fv,/ip4/127.0.0.1/tcp/12181/ipfs/16Uiu2HAmDc6gaSW93baPTfaPHenNus6b5Svvj
eFNoPS5rYgWT1Fv
Enter your password:
INFO [05-26|11:46:44.708] PDX Utopia Version222:=1.9.0
INFO [05-26|11:46:44.709] Maximum peer count PDX=25 LES=0 total=25
Your new account is locked with a password. Please give a password. Do not forget this pas
sword.
Passphrase:
Repeat passphrase:
Address: {bcc2bb52f10cff990a18772c4b0f3def3a382bce}
INFO [05-26|11:46:46.833] PDX Utopia Version222:=1.9.0
INFO [05-26|11:46:46.833] Maximum peer count PDX=25 LES=0 total=25
Enter the reward address :[ default is miner address: bcc2bb52f10cff990a18772c4b0f3def3a38
2bce ]
Enter the FirstMinnerAddress :[ default is miner address: bcc2bb52f10cff990a18772c4b0f3def
3a382bce ]:
INFO [05-26|11:46:49.112] PDX Utopia Version222:=1.9.0
INFO [05-26|11:46:49.112] Maximum peer count PDX=25 LES=0 total=25
INFO [05-26|11:46:49.121] Allocated cache and file handles database=/Users/wuxinya
ng/Desktop/utopia/chain/123/data/utopia/chaindata cache=16.78mB handles=16
INFO [05-26|11:46:49.157] Writing custom genesis block
INFO [05-26|11:46:49.159] Persisted trie from memory database nodes=34 size=5.15kB ti
me=625.716µs gcnodes=0 gcsizesize=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [05-26|11:46:49.159] Successfully wrote genesis state database=chaindata
hash=494999...ad3e80
INFO [05-26|11:46:49.159] Allocated cache and file handles database=/Users/wuxinya
ng/Desktop/utopia/chain/123/data/utopia/lightchaindata cache=16.78mB handles=16
INFO [05-26|11:46:49.203] Writing custom genesis block
INFO [05-26|11:46:49.205] Persisted trie from memory database nodes=34 size=5.15kB ti
me=713.38µs gcnodes=0 gcsizesize=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [05-26|11:46:49.205] Successfully wrote genesis state database=lightchaindata
hash=494999...ad3e80

!!! Public blockchain 123: ready to start now !!!
```

新部署的节点要想加入其它存在的链的时候，需要 bootnode 信息，该信息被记录在 /utopia/conf/{chainid}/bootnode.txt 文件中。

## 4.3 创建联盟链实例

### 创建联盟链

创建一个联盟链实例，需要按如下所示运行 \$UTOPIA\_HOME/bin/init.sh 脚本：



```
jz@matebook:/opt/pdx/utopia/bin$ ./init.sh
1) create a new blockchain
2) join an existing blockchain
Please pick an option: 1
Enter the blockchain id to create [739]:111
1) public blockchain
2) consortium blockchain
Please pick an option: 2
Enter your password:
INFO [05-26|10:02:57.787] PDX Utopia Version222:=1.9.0
INFO [05-26|10:02:57.788] Maximum peer count PDX=25 LES=0 total=25
Your new account is locked with a password. Please give a password. Do not forget this password
.
Passphrase:
Repeat passphrase:
Address: {33373037825a9530b1b7d41a60e75e02f51511b6}
INFO [05-26|10:03:00.019] PDX Utopia Version222:=1.9.0
INFO [05-26|10:03:00.019] Maximum peer count PDX=25 LES=0 total=25
Enter the reward address :[ default is miner address: 33373037825a9530b1b7d41a60e75e02f51511b6
]
Enter the FirstMinerAddress :[ default is miner address: 33373037825a9530b1b7d41a60e75e02f51511b6
]
INFO [05-26|10:03:02.118] PDX Utopia Version222:=1.9.0
INFO [05-26|10:03:02.118] Maximum peer count PDX=25 LES=0 total=25
INFO [05-26|10:03:02.129] Allocated cache and file handles database=/Users/wuxinyang/Desktop/utopia/chain/111/data/utopia/chaindata cache=16.78MB handles=16
INFO [05-26|10:03:02.172] Writing custom genesis block
INFO [05-26|10:03:02.177] Persisted trie from memory database nodes=34 size=5.15kB time=361.11µs gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 liveness=0.00B
INFO [05-26|10:03:02.177] Successfully wrote genesis state database=chaindata
hash=fd0a8d...a70f16
INFO [05-26|10:03:02.177] Allocated cache and file handles database=/Users/wuxinyang/Desktop/utopia/chain/111/data/utopia/lightchaindata cache=16.78MB handles=16
INFO [05-26|10:03:02.210] Writing custom genesis block
INFO [05-26|10:03:02.214] Persisted trie from memory database nodes=34 size=5.15kB time=2.074763ms gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 liveness=0.00B
INFO [05-26|10:03:02.214] Successfully wrote genesis state database=lightchaindata
hash=fd0a8d...a70f16
Before start make sure copy cert into /Users/wuxinyang/Desktop/utopia/chain/111/data/consortium
!!! Consortium blockchain 111: configure permissions before start !!!
```

## 生成目录

该安装过程产生的目录，除\$UTOPIA\_HOME/chain/\$CHAIN\_ID/data 目录以外，其它目录与公链一样。目录下会有一个联盟链的目录，用于存放节点的配置信息、dapp 和用户权限信息。

## 联盟链目录

如下图所示的联盟链目录，存储着 consortium.conf、证书颁发机构签发的授权节点证书、授权用户证书、本地证书。以下是 consortium.conf 的示例：

```
jz@matebook:/opt/pdx/utopia$ tree .
.
├── bin
│   ├── baap.jar
│   ├── baap_start.sh
│   ├── baap_stop.sh
│   ├── fini.sh
│   ├── init.sh
│   ├── start.sh
│   ├── status.sh
│   ├── stop.sh
│   ├── utopia
│   ├── utopia_start.sh
│   └── utopia_stop.sh
├── chain
│   ├── 111
│   │   ├── data
│   │   │   ├── consortium
│   │   │   │   ├── consortium.conf
│   │   │   │   ├── localhost.crt
│   │   │   │   ├── node-ca-abc.crt
│   │   │   │   ├── node-ca-pdx.crt
│   │   │   │   ├── user-ca-abc.crt
│   │   │   │   └── user-ca-pdx.crt
│   │   │   ├── keystore
│   │   │   │   └── UTC--2019-11-07T02-13-58.850212000Z--df1742653c60fbc9552daffec9764e030f208ed5
│   │   │   └── utopia
│   │   │       ├── LOCK
│   │   │       ├── chaindata
│   │   │       │   ├── 000002.ldb
│   │   │       │   ├── 000003.log
│   │   │       │   ├── CURRENT
│   │   │       │   ├── CURRENT.bak
│   │   │       │   ├── LOCK
│   │   │       │   ├── LOG
│   │   │       │   └── MANIFEST-000004
│   │   │       ├── ewasmcode
│   │   │       │   ├── 000001.log
│   │   │       │   ├── CURRENT
│   │   │       │   ├── LOCK
│   │   │       │   ├── LOG
│   │   │       │   └── MANIFEST-000000
│   │   │       ├── lightchaindata
│   │   │       │   ├── 000001.log
│   │   │       │   ├── CURRENT
│   │   │       │   ├── LOCK
│   │   │       │   ├── LOG
│   │   │       │   └── MANIFEST-000000
│   │   │       └── transactions.rlp
│   │   ├── genesis.json
│   │   └── genesis.template
│   └── conf
│       ├── 111
│       │   ├── bootnode.txt
│       │   └── chain-info.properties
│       └── baap
│           ├── baap.properties
│           ├── nodekey
│           └── pubkey
├── temp
│   ├── 111
│   │   ├── log.out
│   │   └── utopia.pid
│   └── baap
│       ├── chain-state.json
│       ├── dapps
│       └── java
└── unity
    └── unity.properties
```

## 联盟链配置信息

```
{
  "name" : "pdx-ecosys",
  "orgs" : [
    {
      "name" : "pdx",
      "node_ca" : ["node-ca-pdx.crt"],
      "user_ca" : ["user-ca-pdx.crt"]
    },
    {
      "name" : "abc",
      "node_ca" : ["node-ca-abc.crt"],
      "user_ca" : ["user-ca-abc.crt"]
    }
  ],
  "dapp_auth":true,
  "user_auth":false
}
```

### 联盟链 JWT

在这里，dapp\_auth（默认为 false）定义是否可以将 d-App 部署到 PDX Utopia 区块链上的权限，而 user\_auth（默认为 false）定义是否可以将交易发送到 PDX Utopia 区块链上的权限。

JWT (Json Web 令牌) 用于定义 d-App 部署和交易的权限。JWT 头部和载荷定义如下：

```
{
  "alg" : "ES256",
  "typ" : "JWT"
}
```

```
{
  "ak" : "hex without 0x of signer's public key",
  "sk" : "hex without 0x of sender's public key",
  "r" : "d", // role of sender: "d" for deployment, "u" for end-user
  "l" : 5, // invalidated after l blocks starting from first use
  "s" : 12345, // signer assigned sequence number, for future revocation
  "n" : "signed generated random string"
}
```

下表列出了根据 dapp-auth 和 user-auth 设置执行部署交易和非部署交易的角色：

dapp-auth	user-auth	role for deployment TX	role for non-deployment TX
true	true	d	U
false	true	u	U
true	false	d	-
false	false		-

给定签名者的私钥和发送者的公钥以及其他信息，可以在 <https://jwt.io/> 上生成 JWT。

在访问 PDX Utopia 联盟链时，如果需要 JWT，则开发人员必须将 jwt 字段设置为从签名机构获取的 JWT。

## 4.4 启动区块链实例

要启动区块链实例，请运行 \$UTOPIA\_HOME/bin/start.sh，如下所示：

```
jz@matebook:/opt/pdx/utopia/bin$ ./start.sh
PDX utopia blockchain 739 started, check /opt/pdx/utopia/temp/739/log.out for details.
```

## 4.5 停止区块链实例

要停止区块链实例，请运行\$UTOPIA\_HOME/bin/stop.sh，如下所示：

```
jz@matebook:/opt/pdx/utopia/bin$ ./stop.sh
PDX Utopia blockchain 739 is now stopped.
```

## 4.6 删除区块链实例

要删除区块链实例，请运行\$UTOPIA\_HOME/bin/fini.sh，如下所示：

```
jz@matebook:/opt/pdx/utopia/bin$ ./fini.sh
PDX Utopia blockchain 739 is now removed.
```

# 5 Utopia 联合 PDX BaaP 部署

## 5.1 配置 PDX BaaP

%BAAP\_HOME%/conf/baap/baap.properties 如下所示：

```
# GRPC port
pdx.baap.port=9052
# chainCode
pdx.baap.chaincode.max.active.num=100
pdx.baap.chaincode.invoke.max.time=5000
# 硬件浮动值 0.2 2000/mb 100/mb
pdx.baap.cpu.floating.value=0.2
pdx.baap.disk.floating.value=2000
pdx.baap.memory.floating.value=100
```

## 5.2 集成 PDX Unity

如果要在 PDX Unity 下使用 PDX Utopia/BaaP，请添加以下内容到路径为\$UTOPIA\_HOME/unity/unity.properties 的配置文件：

```
pdx.unity.url=https://unity.cloud.pdx.ltd
```

如果加入您自己的 PDX Blockchain Hypercloud，则将 URL 设置为您的 PDX Unity 实例。  
如果不需要与 PDX Unity 集成，则此文件一定不存在。

## 5.3 启动 PDX Utopia + BaaP

首先，使用 `$UTOPIA_HOME/bin/utopia_start.sh` 启动 PDX Utopia，然后使用 `$UTOPIA_HOME/bin/baap_start.sh` 启动 PDX BaaP。

## 5.4 停止 PDX Utopia + BaaP

首先，使用 `$UTOPIA_HOME/bin/utopia_stop.sh` 停止 PDX Utopia，然后使用 `$UTOPIA_HOME/bin/baap_stop.sh` 停止 PDX BaaP。

## 6 Utopia 联合 PDX Unity 部署

在 Utopia 联合 PDX Unity 的方式中，需要从 <https://pdx.ltd/download> 下载 Node 压缩包。下载完成后解压到 `$UTOPIA_HOME` 目录下，例如 `/opt/pdx/node`。

此方式可以方便用户加入 Unity 生态，加入生态后会自动加入主链，同时 Node 程序的宿主机在参与平台活动中会获得一定的奖励。

### 6.1 启动 node

使用 `$UTOPIA_HOME/bin/start.sh` 启动 node。

### 6.2 停止 node

使用 `$UTOPIA_HOME/bin/stop.sh` 停止 node。

### 6.3 更新 node

使用 `$UTOPIA_HOME/bin/updatet.sh` 更新 node。

### 6.4 清除 node

使用 `$UTOPIA_HOME/bin/remove.sh` 清除 node 所有数据。

## 7 附录 A：生成 X509 v3 证书

本附录说明了如何使用 openssl 创建 `node_ca`, `user_ca` 授权并生成节点证书。



## 7.1 创建 Root-CA 证书密钥对

```
openssl ecparam -genkey -name secp256k1 -out ./RootCA/private/root-ca.key
```

## 7.2 创建 Root-CA 证书请求

```
openssl req -new -days 365 -key ./RootCA/private/root-ca.key -out root-ca.csr -subj  
"/C=CN/ST=Beijing/L=Haidian/O=PDX/OU=PDX/CN= Root-CA"
```

## 7.3 通过自签名创建 Root-CA 证书

```
openssl ca -selfsign -in root-ca.csr -out root-ca.crt
```

## 7.4 创建一个中间 Node-CA

创建 Node-CA 证书密钥对

```
openssl ecparam -genkey -name secp256k1 -out ./NodeCA/private/node-ca.key
```

创建 Node-CA 证书请求

```
openssl req -new -days 365 -key ./NodeCA/private/node-ca.key -out node-ca.csr -subj  
"/C=CN/ST=Beijing/L=Haidian/O=PDX/OU=PDX/CN=Node-CA "
```

Root-CA 创建 Node-CA 证书

```
openssl ca -in node-ca.csr -out node-ca.crt -days 365 -cert root-ca.crt  
-keyfile ./RootCA/private/root-ca.key
```

## 7.5 创建一个节点证书

通过矿工密钥对生成节点证书请求文件

```
./utopia genscr --keystorefile <keystore file> --password <password file>
```

Node CA 生成节点证书

```
openssl ca -in node-x.csr -out node-x.crt -days 365 -cert node-ca.crt  
-keyfile ./NodeCA/private/node-ca.key
```