Quorum의 주요 기능 중 하나는 트랜잭션 프라이버시입니다. 이를 위해 ‘퍼블릭 트랜잭션’ 및 ‘프라이빗 트랜잭션’ 개념을 설명드리도록 하겠습니다. 이것은 개념적인 콘셉트일 뿐이며 Quorumㅇ새로운 트랜잭션 유형을 도입하지 않지만 이더리움 트랜잭션 모델에서 선택적인 privateFor 파라미터를 포함하도록 확장되었으며(이로 인해 Quorum에서 트랜잭션이 프라이빗한 것으로 처리됨) 트랜잭션 타입에는 이러한 트랜잭션을 식별하는 새로운 IsPrivate 요소가 추가되었습니다.

[Constellation](https://github.com/jpmorganchase/constellation) / [Tessera](https://github.com/jpmorganchase/tessera)는 암호화 및 그 관련 작업을 수행하며 프라이빗 페이로드(payloads)를 의도된 수신자에게 전송하기 위한 목적으로 Quorum에서 사용됩니다.

## 퍼블릭 트랜잭션 (Public Transactions)

소위 말하는 ‘퍼블릭 트랜잭션’은 같은 Quorum 네트워크의 모든 참가자가 페이로드를 볼 수 있는 트랜잭션입니다. 이들은 [일반적인 방식의 표준 이더리움 트랜잭션으로 생성됩니다.](https://github.com/ethereum/wiki/wiki/JavaScript-API#web3ethsendtransaction)

퍼블릭 트랜잭션의 예로는 일부 서비스 제공업체의 마켓 데이터 업데이트 또는 채권 보안 정의에 대한 수정과 같은 일부 참고 데이터 업데이트가 있습니다.

노트: ‘퍼블릭’ 트랜잭션은 퍼블릭 이더리움 네트워크의 거래가 아닙니다. ‘공동’ 혹은 ‘글로벌’ 트랜잭션이 더욱 적절한 표현일 수 있지만, ‘프라이빗’ 트랜잭션과 대조하기 위해 ‘퍼블릭’이라는 용어를 사용합니다.

## 프라이빗 트랜잭션 (Private Transactions)

소위 말하는 ‘프라이빗 트랜잭션’은 프라이빗 키가 트랜잭션의 privateFor 파라미터에 지정된 네트워크 참여자에게만 페이로드가 보여지는 트랜잭션입니다. privateFor은 쉼표로 구분된 목록에서 여러 개의 주소를 추출할 수 있습니다. ([[Quorum 사용하기|Quorum 사용하기]] 섹션에서 ‘프라이빗 트랜잭션 생성’ 참조)

Quorum 노드가 null이 아닌 privateFor 값을 가진 트랜잭션과 만나면 트랜잭션 서명의 V 값을 37 또는 38로 설정합니다. (이더리움 옐로우 페이퍼에 명기된 바와 같이 표준 이더리움 상에서 트랜잭션이 ‘퍼블릭’임을 나타내기 위한 값이 27 또는 28인 것과는 대조적)

## 트랜잭션 처리

### 퍼블릭 vs 프라이빗 트랜잭션 핸들링

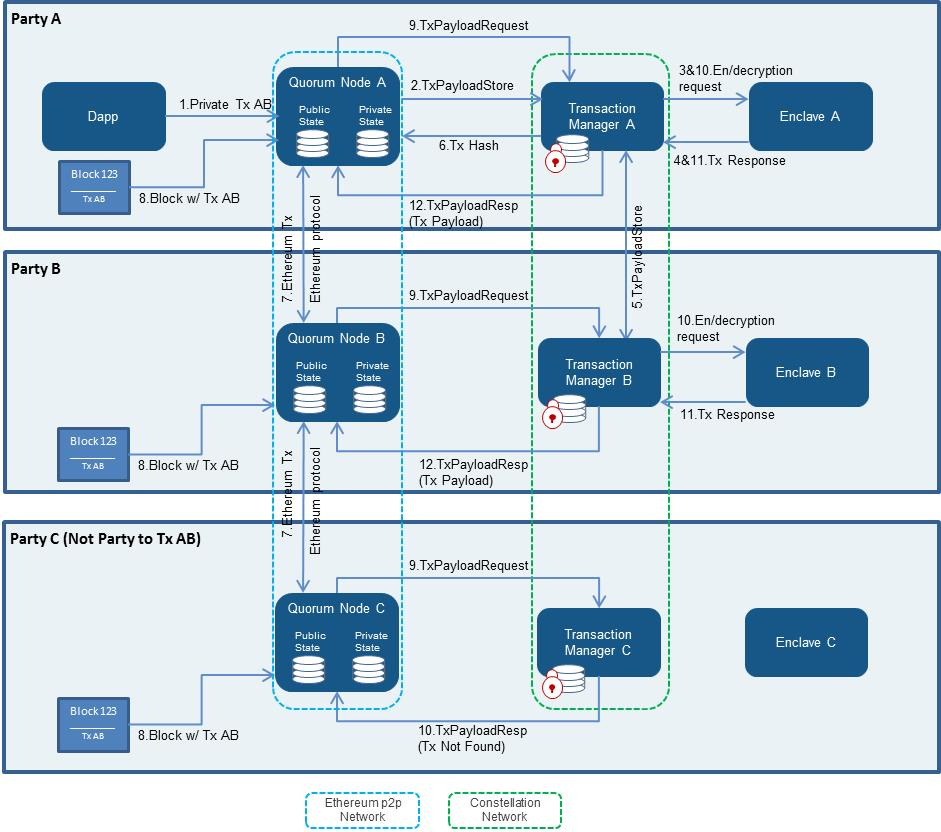
퍼블릭 트랜잭션은 표준 이더리움 방식으로 실행되므로 퍼블릭 트랜잭션이 콘트랙트 코드를 보유한 계정으로 전송되면 각 참여자는 동일한 코드를 실행하고 이에 따라 StateDB가 업데이트 됩니다.

하지만 프라이빗 트랜잭션은 표준 이더리움에 따라 실행되지 않습니다. 발신자의 Quorum 노드가 트랜잭션을 나머지 네트워크로 전달하기 전에 원래 트랜잭션 페이로드를 Constellation/Tessera에서 받은 암호화된 페이로드의 해시로 바꿉니다. 거래 당사자인 참여자들은 Constellation/Tessera 인스턴스를 통해 실제 페이로드로 해시를 대체할 수 있으나 당사자가 아닌 참여자들은 해시만 볼 수 있습니다.

그 결과, 프라이빗 트랜잭션이 콘트랙트 코드를 보유한 계정으로 전송되면 거래 당사자가 아닌 참여자들은 거래를 생략하고 결국 콘트랙트 코드를 실행하지 않게 됩니다. 그러나 거래 당사자인 참여자들은 실행을 위해 EVM을 호출하기 전에 해시를 원래 페이로드로 대체하고 그에 따라 StateDB가 업데이트됩니다. geth 클라이언트에 상응하는 변경 사항이 없을 경우, 이 두 세트의 참여자는 서로 다른 StateDB를 가진 채로 끝나게 되며 합의에 도달하지 못하게 됩니다. 따라서 콘트랙트 상태의 이러한 분기점을 해결하기 위해 Quorum은 퍼블릭 콘트랙트 상태를 전역적으로 동기화된 프라이빗 스테이트 트리(Trie)에 저장하고 프라이빗 콘트랙트 상태를 전역적으로 동기화되지 않은 프라이빗 스테이트 트리(Trie)에 저장합니다. 이와 관련하여 컨센서스가 달성되는 방법에 대한 자세한 내용은 [[컨센서스|Quorum-컨센서스]] 섹션을 참조하십시오.

### 프라이빗 트랜잭션 프로세스 플로우

다음은 프라이빗 트랜잭션이 Quorum에서 처리되는 방법에 대한 설명입니다.



아래 예시에서 참여자 A와 B는 트랜잭션AB의 당사자이고 참여자 C는 당사자가 아닙니다.

1. 참여자 A는 트랜잭션 페이로드를 지정하고 privateFor를 참여자 A와 B의 퍼블릭 키로 설정하여 Quorum 노드에 트랜잭션을 전송합니다.
2. 참여자 A의 Quorum 노드는 트랜잭션을 페어링된 트랜잭션 관리자에게 전달하여 트랜잭션 페이로드를 저장하도록 요청합니다.
3. 참여자 A의 트랜잭션 매니저는 발신자의 유효성을 검사하고 페이로드를 암호화하기 위해 엔클레이브를 호출합니다.
4. 참여자 A의 엔클레이브는 참여자 A의 프라이빗 키를 확인하고 유효함이 확인되면 트랜잭션 변환을 수행합니다. 이는 다음을 수반합니다:
   1. 대칭키 및 랜덤 논스(Nonce) 생성
   2. 1번의 대칭키로 트랜잭션 페이로드 및 논스 암호화
   3. 2번에서 암호화 된 페이로드의 SHA3-512 해시를 계산
   4. 트랜잭션 수신자 목록 (이 경우 참여자 A와 B)을 반복하고 수신자의 퍼블릭키(PGP 암호화)를 이용하여 1번에서의 대칭키를 암호화
   5. 단계 2에서 암호화 된 페이로드를 및 단계 3에서의 해시, 그리고 4단계에서 암호화된 키 (각 수신자 별)를 트랜잭션 관리자에게 반환
5. 참여자 A의 트랜잭션 매니저는 암호화된 페이로드 (대칭키로 암호화됨)와 해시를 인덱스로 사용하여 암호화된 대칭키를 저장한 다음, 해시, 암호화된 페이로드 및 참여자 B의 퍼블릭키를 통해 암호화된 대칭키를 안전하게 (HTTPS를 통해) 참여자 B의 트랜잭션 매니저에게 전송합니다. 참여자 B의 트랜잭션 매니저는 Ack/Nack response로 응답합니다. 참여자 A가 응답을 받지 못하거나 참여자 B로부터 Nack을 수신하면 트랜잭션이 네트워크에 전파되지 않습니다. 수신자가 전달된 페이로드를 저장하는 것은 전제 조건입니다.
6. 참여자 B의 트랜잭션 관리자에게의 데이터 전송이 성공하면 참여자 A의 트랜잭션 매니저는 해시를 Quorum 노드에 반환하고 Quorum 노드는 트랜잭션의 원래 페이로드를 해당 해시로 바꾼 다음, 트랜잭션의 V값을 37 또는 38로 변경합니다. 이는 다른 노드들에게 이 해시는 무의미한 바이트 코드가 있는 퍼블릭트랜잭션과는 달리 연관된 암호화 된 페이로드가 있는 프라이빗 트랜잭션을 나타냄을 보여줍니다.
7. 트랜잭션은 표준 이더리움 P2P 프로토콜을 사용하여 나머지 네트워크로 전파됩니다.
8. 트랜잭션 AB를 포함하는 블록이 생성되어 네트워크의 각 참여자에게 배포됩니다.
9. 블록을 처리하는 과정에서 모든 참여자들은 트랜잭션 처리를 시도합니다. 각 Quorum 노드는 37 또는 38의 V값을 인식하여 페이로드가 해독을 필요로 하는 것으로 트랜잭션을 식별하고 로컬 트랜잭션 관리자를 호출하여 트랜잭션을 보류할지 확인합니다. (조회 시, 해시를 인덱스로 사용)
10. 참여자 C는 트랜잭션을 보유하지 않으므로 NotARecipient 메시지를 받고 트랜잭션을 생략하고 프라이빗 StateDB는 업데이트되지 않습니다. 참여자 A와 B는 로컬 트랜잭션 관리자에서 해시를 조회하고 트랜잭션을 보유하고 있음을 확인합니다. 그런 다음 각각 엔클레이브로 호출을 걸어 암호화 된 페이로드, 암호화 된 대칭키 및 서명을 전달합니다.
11. 엔클레이브는 서명의 유효성을 검사한 다음 참여자의 개인키를 활용하여 대칭키를 해독하고 현재 공개된 대칭키를 사용하여 트랜잭션 페이로드를 해독한 후, 트랜잭션 매니저에게 암호화된 페이로드를 반환합니다.
12. 참여자 A와 B에 대한 트랜잭션 매니저는 콘트랙트 코드 실행을 위해 해독된 페이로드를 EVM으로 보냅니다. 이 실행은 Quorum 노드의 프라이빗 StateDB에서만 상태를 업데이트합니다. 참고: 일단 코드가 실행되면 폐기되므로 위의 프로세스를 거치지 않고 읽을 수 없습니다.