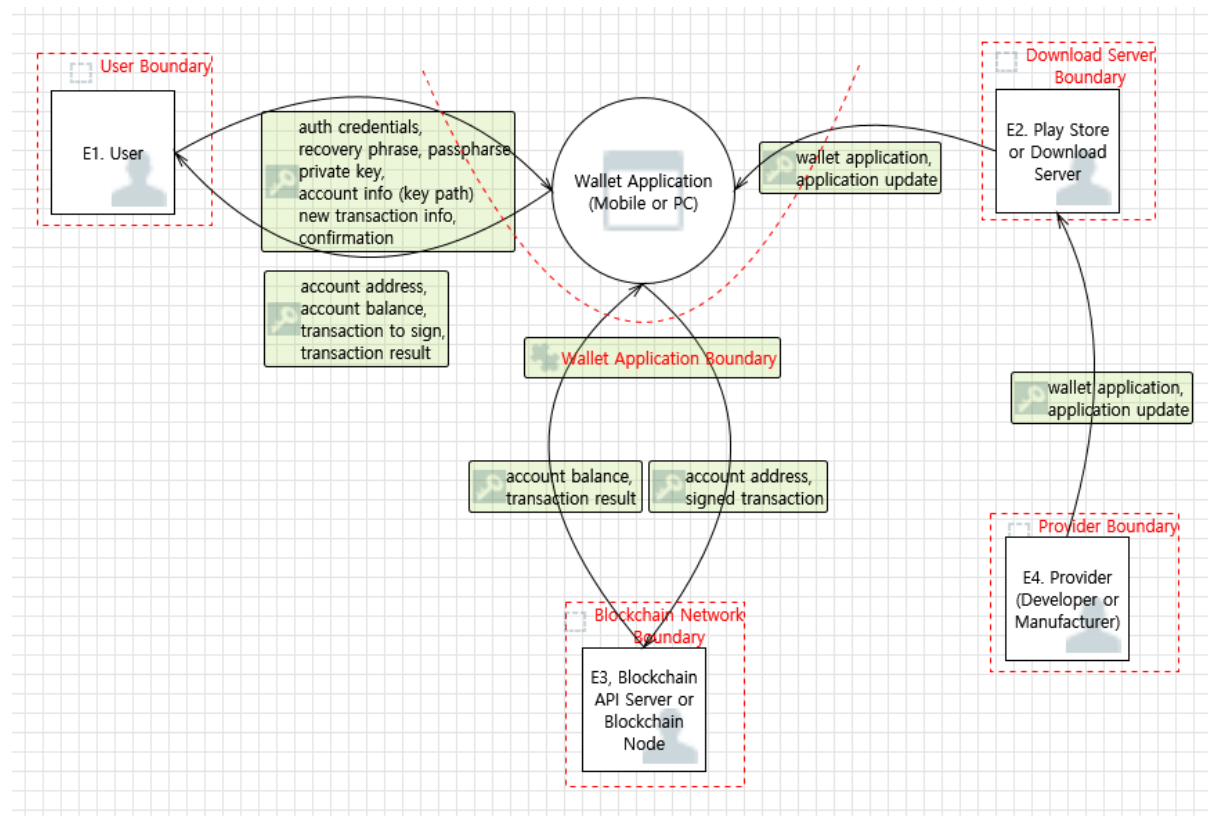
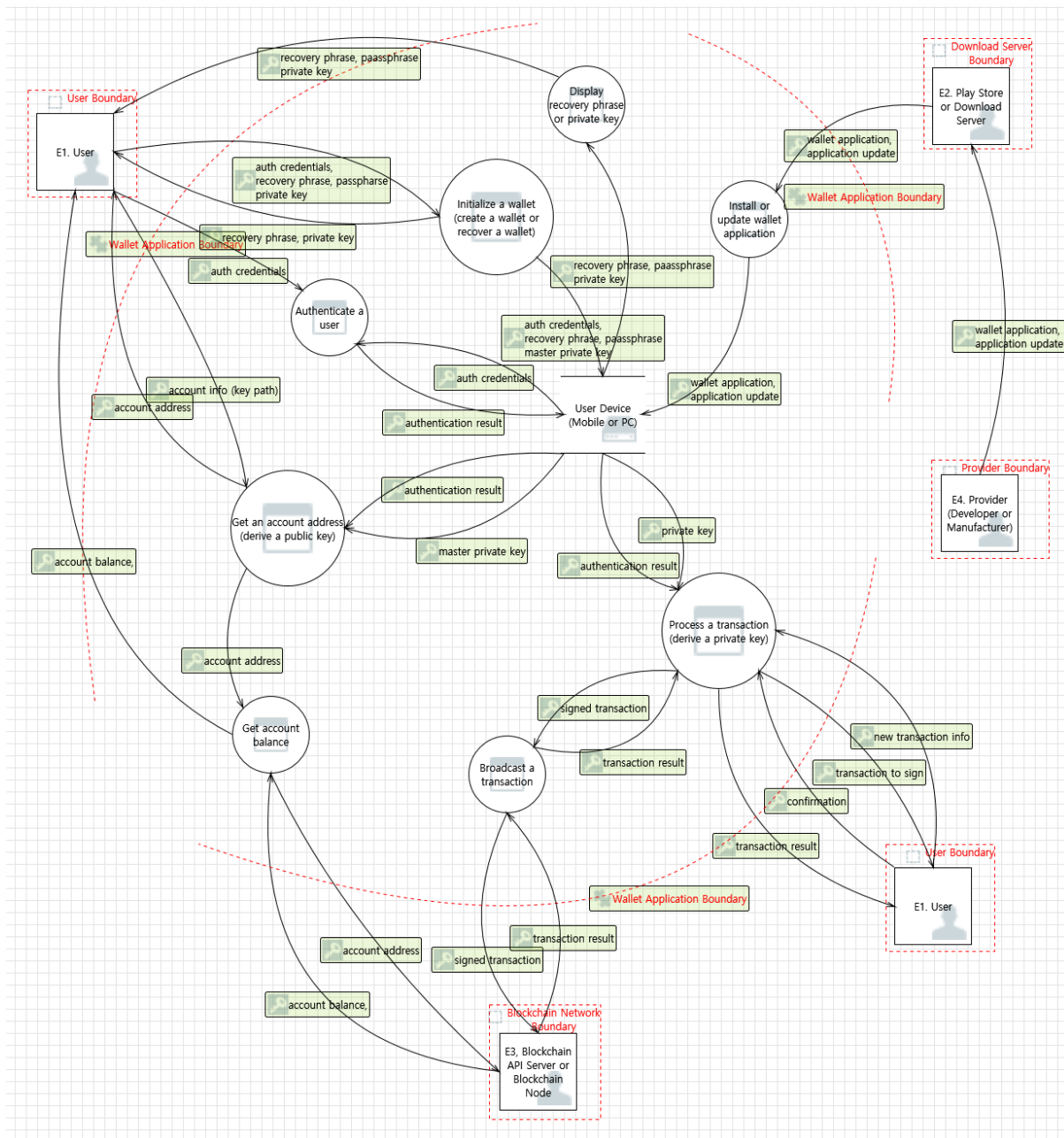


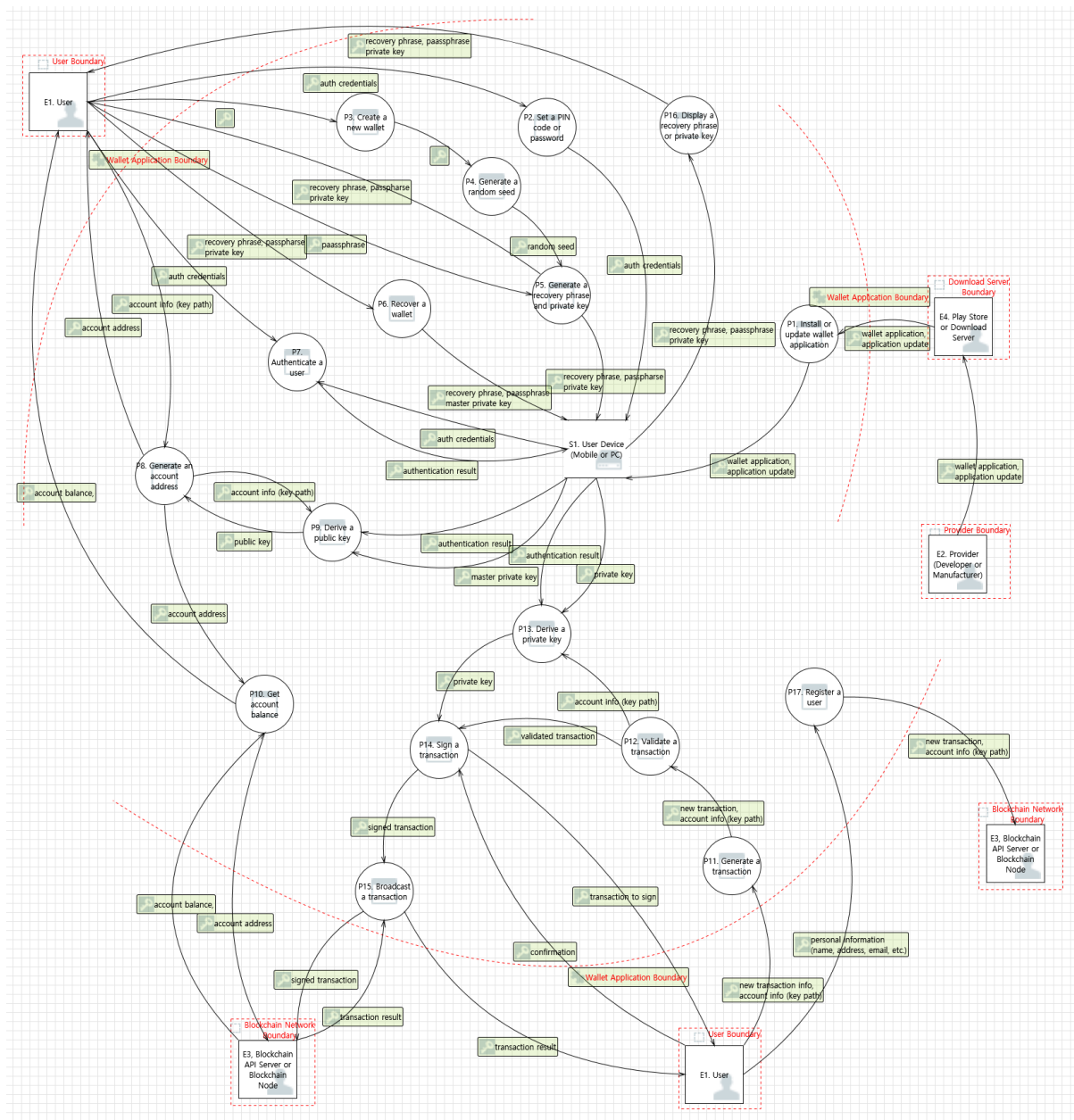
## 1.1 Cryptocurrency Hot Wallet DFD – Level 0



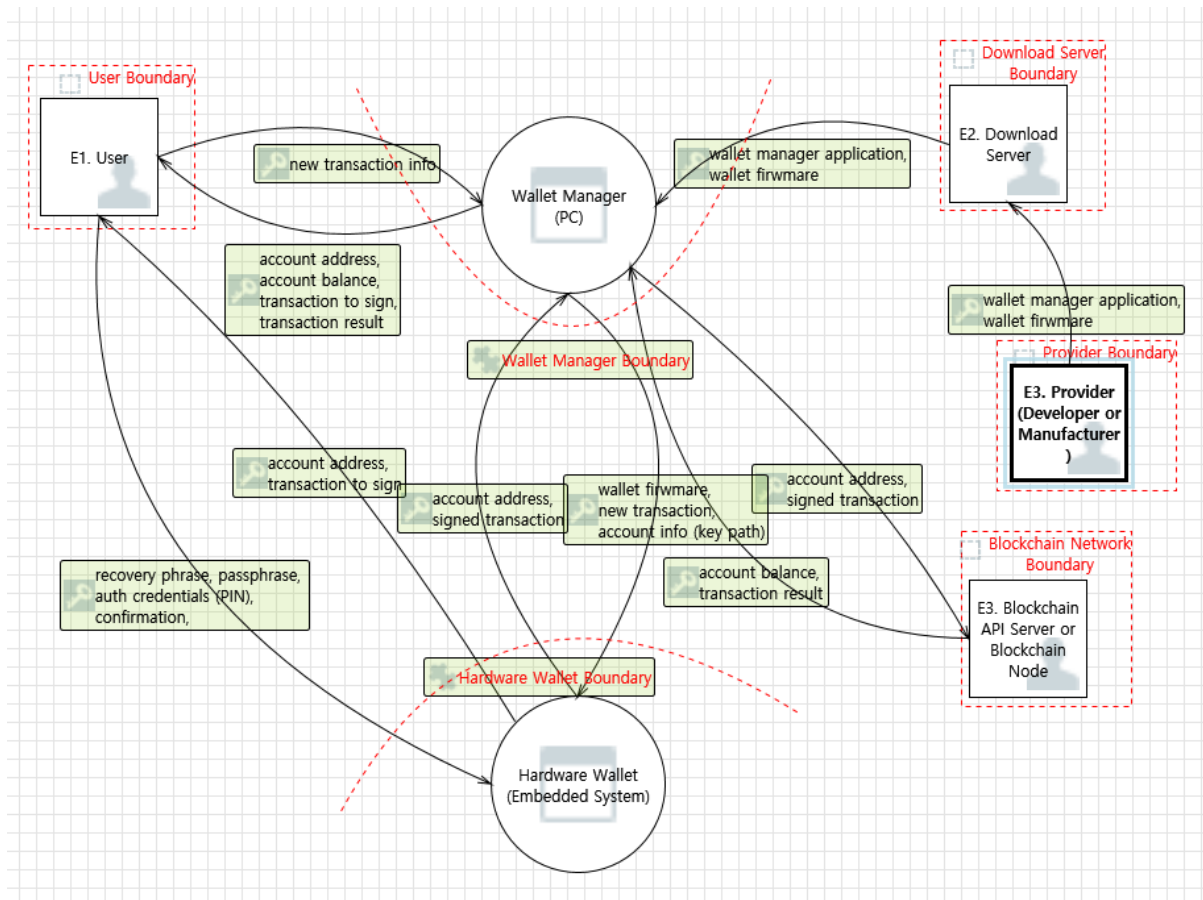
## 1.2 Cryptocurrency Hot Wallet DFD – Level 1



### 1.3 Cryptocurrency Hot Wallet DFD – Level 2



## 2.1 Cryptocurrency Cold Wallet DFD – Level 0



The diagram illustrates the interactions within a Blockchain Wallet System, organized into several boundaries and involving multiple actors and use cases.

**Boundaries:**

- Blockchain Network Boundary:** Contains E3, Blockchain API Server or Blockchain Node.
- User Boundary:** Contains E1. User.
- Download Server Boundary:** Contains E2. Download Server.
- Provider Boundary:** Contains E4. Provider (Developer or Manufacturer).
- Wallet Manager Boundary:** Contains wallet manager application, wallet firmware, transaction result, signed transaction, new transaction info, transaction to sign, signed transaction, transaction result, Broadcast a transaction, Generate a transaction, signed transaction, transaction to sign, confirmation, signed transaction, new transaction, account info (key path), Update firmware, wallet firmware, Hardware Wallet Device (Embedded System), master private key, authentication result, auth credentials (PIN), authentication result, auth credentials (PIN), master private key, Initialize a wallet (create a wallet or recover a wallet), recovery phrase, private key, auth credentials (PIN), recovery phrase, passphrase.
- Hardware Wallet Boundary:** Contains Hardware Wallet Device (Embedded System).

**Actors:**

- E1. User
- E2. Download Server
- E3. Blockchain API Server or Blockchain Node
- E4. Provider (Developer or Manufacturer)

**Use Cases and Interactions:**

- Install or update wallet application:** Interacts with E2. Download Server and E1. User.
- Download firmware:** Interacts with E2. Download Server and E1. User.
- Get an account address:** Interacts with E1. User and E3.
- Get account balance:** Interacts with E1. User and E3.
- Generate a transaction:** Interacts with E1. User and E3.
- Process a transaction (derive a private key):** Interacts with E1. User and E3.
- Initialize a wallet (create a wallet or recover a wallet):** Interacts with E1. User and E3.
- Authenticate a user:** Interacts with E1. User and E3.
- Update firmware:** Interacts with E1. User and E3.
- Hardware Wallet Device (Embedded System):** Interacts with E1. User and E3.
- Blockchain API Server or Blockchain Node:** Interacts with E1. User and E3.
- Download Server:** Interacts with E1. User and E3.
- Provider (Developer or Manufacturer):** Interacts with E1. User and E3.

The diagram illustrates the interactions within a Blockchain Wallet System, organized into several boundaries:

- User Boundary:** Includes actors E1: User and use cases P1: Install or update wallet application, P2: Download firmware, P3: Get an account address, P4: Derive a private key, P5: Update firmware, P6: Recover a wallet, P7: Generate a recovery phrase and private key, P8: Authenticate a user, P9: Generate an account address, P10: Get account balance, P11: Derive a public key, P12: Derive a public key, P13: Generate an account address, P14: Generate a transaction, P15: Validate a transaction, P16: Derive a private key, P17: Sign a transaction, P18: Broadcast a transaction, and P19: Create a new wallet.
- Blockchain Network Boundary:** Includes actor E3: Blockchain API Server or Blockchain Node and use cases P1: Install or update wallet application, P2: Download firmware, P3: Get an account address, P4: Derive a private key, P5: Update firmware, P6: Recover a wallet, P7: Generate a recovery phrase and private key, P8: Authenticate a user, P9: Generate an account address, P10: Get account balance, P11: Derive a public key, P12: Derive a public key, P13: Generate an account address, P14: Generate a transaction, P15: Validate a transaction, P16: Derive a private key, P17: Sign a transaction, P18: Broadcast a transaction, and P19: Create a new wallet.
- Download Server Boundary:** Includes actor E2: Download Server and use cases P1: Install or update wallet application, P2: Download firmware, P3: Get an account address, P4: Derive a private key, P5: Update firmware, P6: Recover a wallet, P7: Generate a recovery phrase and private key, P8: Authenticate a user, P9: Generate an account address, P10: Get account balance, P11: Derive a public key, P12: Derive a public key, P13: Generate an account address, P14: Generate a transaction, P15: Validate a transaction, P16: Derive a private key, P17: Sign a transaction, P18: Broadcast a transaction, and P19: Create a new wallet.
- Provider Boundary:** Includes actor E4: Provider (Developer or Manufacturer) and use cases P1: Install or update wallet application, P2: Download firmware, P3: Get an account address, P4: Derive a private key, P5: Update firmware, P6: Recover a wallet, P7: Generate a recovery phrase and private key, P8: Authenticate a user, P9: Generate an account address, P10: Get account balance, P11: Derive a public key, P12: Derive a public key, P13: Generate an account address, P14: Generate a transaction, P15: Validate a transaction, P16: Derive a private key, P17: Sign a transaction, P18: Broadcast a transaction, and P19: Create a new wallet.
- Hardware Wallet Boundary:** Includes actor S2: Hardware Wallet Device (Embedded System) and use cases P1: Install or update wallet application, P2: Download firmware, P3: Get an account address, P4: Derive a private key, P5: Update firmware, P6: Recover a wallet, P7: Generate a recovery phrase and private key, P8: Authenticate a user, P9: Generate an account address, P10: Get account balance, P11: Derive a public key, P12: Derive a public key, P13: Generate an account address, P14: Generate a transaction, P15: Validate a transaction, P16: Derive a private key, P17: Sign a transaction, P18: Broadcast a transaction, and P19: Create a new wallet.

The diagram shows a flow of interactions between these actors and use cases, with data elements like transaction results, signed transactions, wallet manager applications, wallet firmware, account addresses, account balances, master private keys, authentication results, recovery phrases, passphrases, private keys, public keys, and credentials (PIN) being exchanged. Red dashed lines indicate the boundaries between these different components.