Vormetric Data Security

암호화 키 관리 및 전송

Vormetric | Data Security Simplified

암호화 키 관리 개요

- **데이터 암호화 키 보안**
  - 키 생성 시 키 값은 자동 생성
  - 자동 생성한 키 값은 키 관리자 또는 VDF 솔루션의 어떤 관리자도 알 수 없음
  - 키는 솔루션 전체 라이프 사이클에 걸쳐 항상 암호화 유지
- **데이터 암호화 키 생성 (간단한 키 관리)**
  - 키 이름 및 길이 (Algorithm)
  - 키 유형: Cached on Host, Stored On Server, Stored On Server (Unique to Host)
  - 다른 암호화 솔루션에서 암호화 한 데이터 관리를 위해 수동 입력 모드 지원
- **데이터 암호화 키 저장 및 배포**
  - 암호화 키 저장소: 키 관리 서버 (DSM: Data Security Manager) 중앙 관리 방식
  - DSM은 Network HSM으로, 내부에 키의 안전한 저장을 위한 Key Store가 있음
  - 배포 시기: 호스트 (암호화 데이터를 저장하는 서버들) 재부팅 또는 에이전트 재시작 시

암호화 키 유형

- **Stored On Server**
  - 암호화 키를 서버에 저장하고 호스트에는 남겨두지 않음
  - 호스트 재부팅 등 키 배포 시점에 DSM이 연결되지 않으면 암호화 데이터 액세스 불가능
- **Cached on Host**
  - DSM에 저장된 암호화 키를 호스트 암호로 암호화하여 호스트에 저장
  - 키 배포 시점에 DSM에 연결하여 키 수신을 시도하고 연결되지 않으면 암호화 데이터에 액세스하는 어플리케이션 Hang
  - Vmsec 명령을 통해 캐시된 키를 메모리에 로드하면 운영 가능
- **Cached on Host (Unique to Host)**
  - 보안을 위해 호스트 별 데이터 암호화 키를 다르게 설정 (한 호스트에서 암호화 키가 유출되어도 다른 호스트의 데이터는 안전)
  - 키 배포 시점에 DSM에 연결하여 키 수신을 시도하고 연결되지 않으면 암호화 데이터에 액세스하는 어플리케이션 Hang
  - Vmsec 명령을 통해 캐시된 키를 메모리에 로드하면 운영 가능

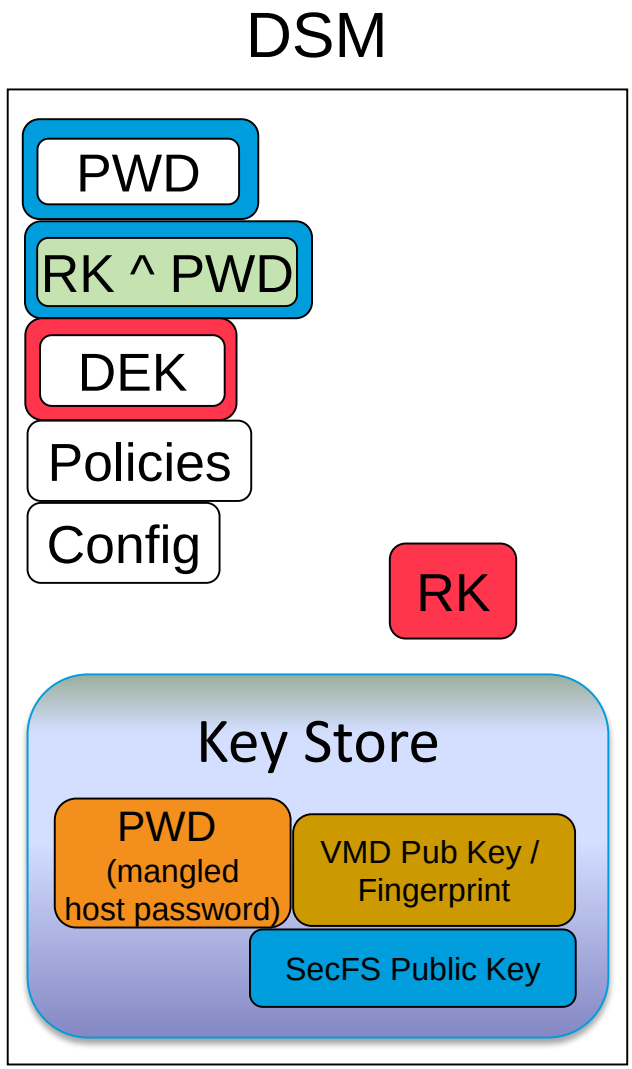
암호화 키의 안전한 전송을 위하여...

- **에이전트 구성 요소 및 보안 통신**
  - 사용자 영역에 통신을 담당하는 vmd, 커널 영역에 암호화를 담당하는 SecFS로 구성
  - vmd 및 SecFS 모두 자신의 인증서 보유
  - DSM에게 Public Key를 제공하고 인증된 DSM에게만 접속 허용
- **암호화 키 전송 절차 개요**
  1. 데이터 암호화 키 (DEK)를 암호화 하기 위해 일회용 AES256 "Random Key" (RK)를 생성하고 DEK를 RK로 암호화
  2. 호스트 암호 (PWD), RK^PWD 값을 암호화 키를 사용할 SecFS의 Public Key로 암호화 (이 값은 SecFS의 Private Key를 가지고 있는 개체, 즉 SecFS 만 해독 가능)
  3. 에이전트 정책 및 구성 정보를 평문으로 패키지에 추가
  4. 에이전트의 vmd 및 DSM 모두 SSL 서버이므로 양방향 SSL 채널 설정
  5. 전송되는 보안의 수준을 향상시키기 위해 현재 OpenSSL에서 지원되는 가장 강력한 Cipher Suite인 "AES256-SHA (RSA_WITH_AES_256_CBC_SHA)"적용
     (전송되는 패키지를 AES256 세션 키로 한 번 더 암호화 하여 보안 수준 향상)

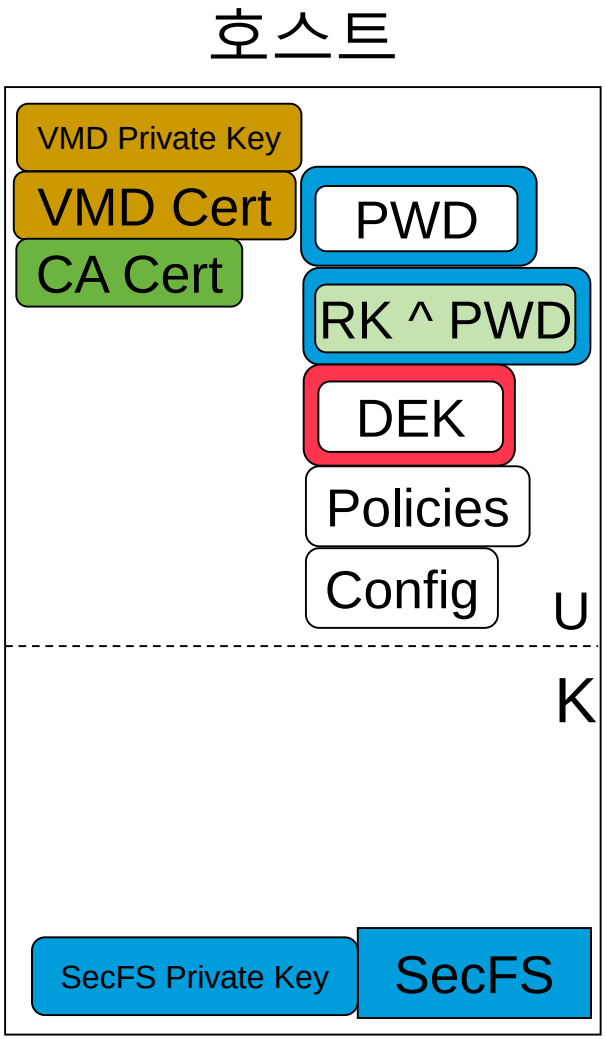
키 전송 전 DSM (그림에서 Server) 준비 사항

- **PWD: 내구 구현 로직에 의해 변형**
- **DEK 암호화**
  - RK (AES256) 생성
  - 전송할 키를 RK로 암호화
- **SecFS Public Key로 암호화**
  - 변형된 PWD
  - RK와 PWD를 Exclusive OR 한 값
- **패키지에 관리 정보 추가**
  - 암호화 정책
  - 구성 정보

패키지를 전송 시 AES256 세션 키로 한 번 더 암호화

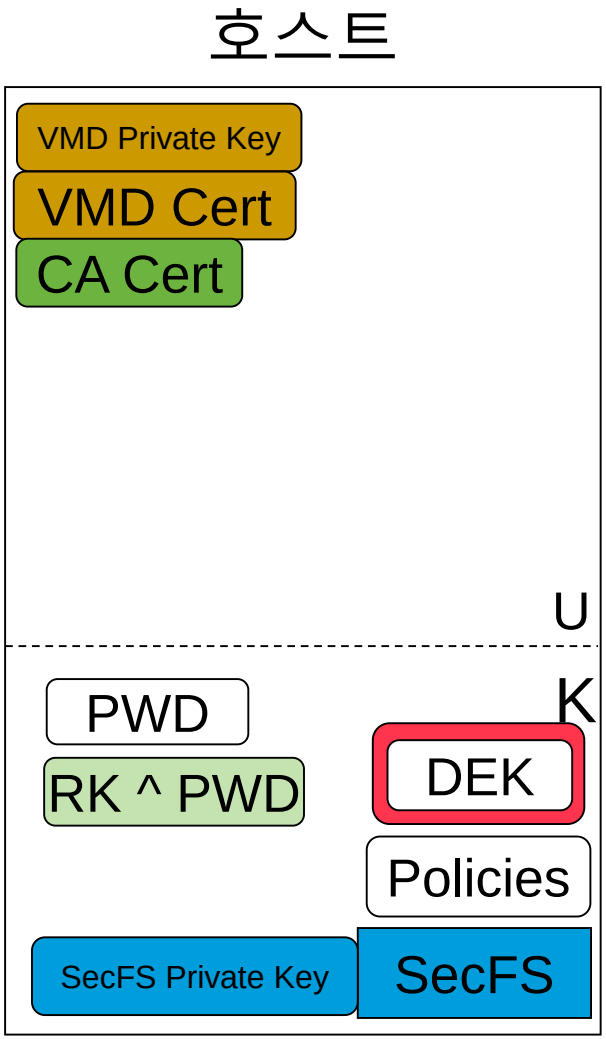


전송 후 에이전트 상태 – 사용자 영역

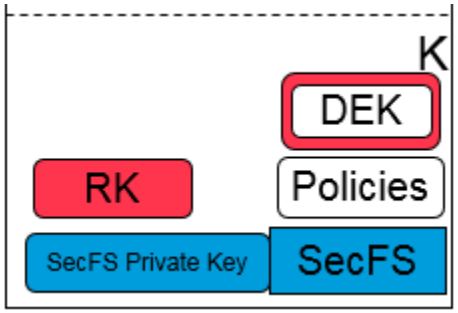


- **vmd가 메모리에서 작업**
  - 암호화 정책
  - 구성 정보
  - PWD, RK^PWD는 SecFS Public Key로 암호화 되어 읽을 수 없음
  - DEK는 RK로 암호화 되어 읽을 수 없음
- **처리 작업**
  - 구성 정보 적용
  - 암호화 정책을 SecFS에 전달
  - PWD, RK^PWD, RK로 암호화 된 패키지를 SecFS에 전달

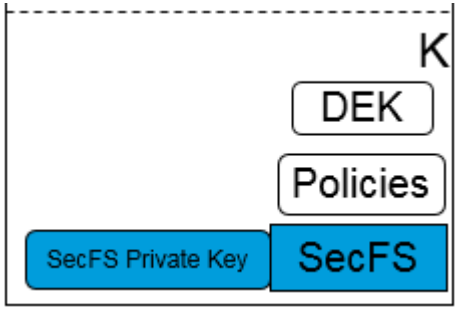
전송 후 에이전트 상태 – 커널 영역



- **SecFS 작업 절차**
  - SecFS Private Key로 PWD, RK^PWD 복호화
  - PWD와 RK^PWD를 연산하여 RK 값 복원

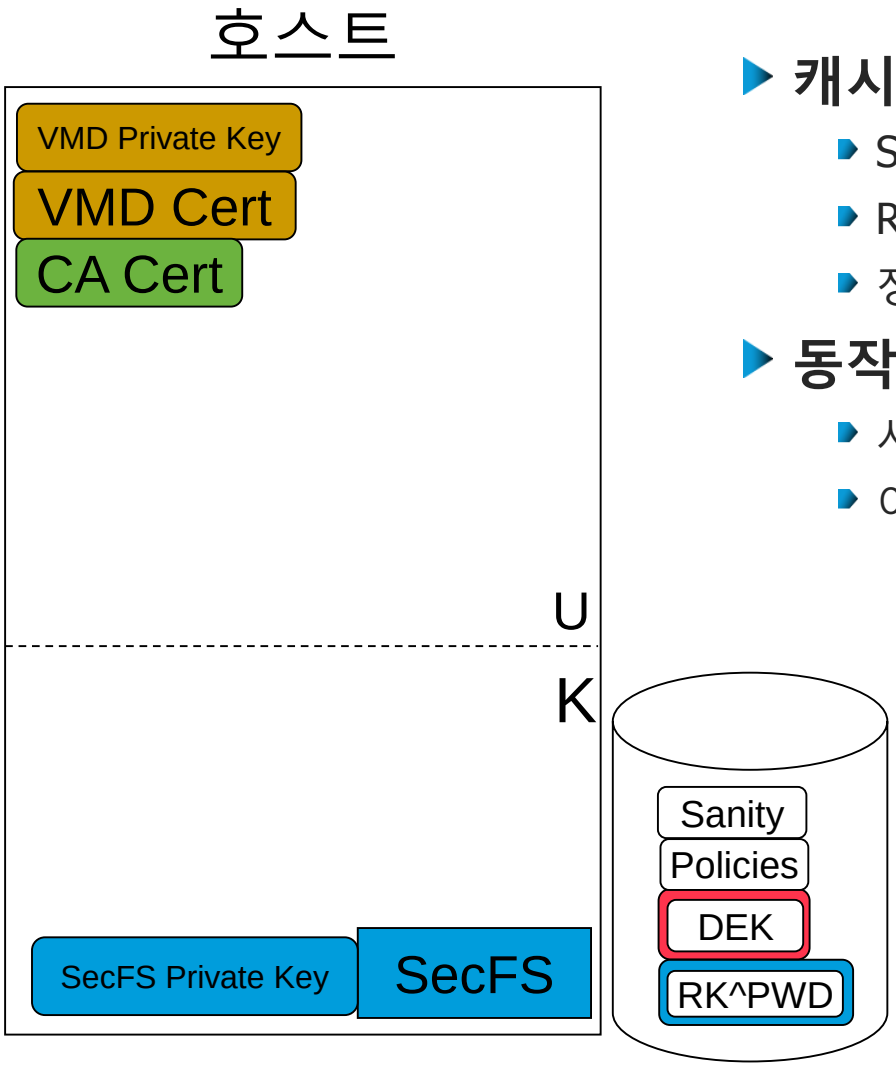


  - RK로DEK 복호화



  - 데이터 암호화/복호화 서비스

Cached on Host 동작 방식



- **캐시되는 내용**
  - SecFS Private Key로 풀기 전의 RK^PWD
  - RK로 암호화 된 DEK
  - 정책
- **동작 방식**
  - 사용자 영역으로부터 PWD 수신
  - 이후 동일 절차 적용

암호화 키 전송 다이어그램