

이러한 상황은 한 번에 한 명의 사용자만이 객체에 접근할 수 있도록 허용되는 경우에 해당할 것이다. 그러나 다중 사용자 환경에서는 특정 객체가 많은 수의 사용자에 의해 사용될 수 있으며, 각 사용자는 다른 액세스 권한을 가질 수도 있다. 일반적으로 이 경우는 단일 사용자에 대한 경우와 동일한 프레임워크 내에서 처리된다. 각 사용자의 프로세스는 해당 사용자에게 고유한 설명자 테이블(세그먼트)으로 표현되며, 이러한 객체를 가리키는 설명자는 해당 사용자에게 적합한 액세스 제어 정보를 설정할 수 있다. 실제 액세스 유형에 대한 검사는 현대 시스템에서 설명자가 참조될 때 하드웨어에서 이루어진다. 이 일반적인 프레임워크 내에서, 몇 가지 부차적인 문제가 발생한다. 그레이엄[3]은 보호를 서로 겹치지 않는 일련의 링으로 다루며, 하나의 보호 수준(同心원 또는 링으로 간주)에서 다른 수준으로 안전하게 제어를 전환하는 문제를 논의한다. 그는 하향(상위 루틴에서 하위 루틴으로) 및 상향 방향 모두에서 보호를 제공하기 위해, 세그먼트의 설명자에 링 경계를 추가하는 모델을 제안한다. 이 모델은 전달되는 요소가 경계 내에 있는 경우 자유롭게 액세스를 허용하지만, 어느 방향으로든 경계를 초과할 경우 특수 소프트웨어를 호출한다. 일반적으로, 특수 소프트웨어는 참조 방향에 관계없이 지정된 주소를 검증한다. 이 방식으로, 메커니즘은 운영 체제로부터 프로세스를 보호하는 동시에, 반대

반달라이트[4]는 그레이엄의 모델을 확장하여, 객체를 공유하는 사용자가 자신이 생성한 프로세스를 다른 사용자에게 사용하도록 허용하는 경우를 포함하는 모델을 제안했다. 그의 모델에서, 그는 프로세스의 활성화 수준에 따라 액세스 권한을 함수로 정의하고, 각 활성화 수준에서 설명자 세그먼트의 복사본을 생성하여 필요한 정확한 제어를 제공한다. 그는 공유 프로시저에 대한 직접 액세스로 발생하는 문제를 구분하고, 모델의 일부로 다음 정책을 채택한다: 프로시저의 소유자가 빌리어 준 프로시저에 대해서만 직접 공유가 허용되며, 소유자가 다른 소유주가 소유한 프로시저를 사용하여 공유될 프로시저를 작성하는 경우에만 간접 공유가 허용된다. 후자의 경우, 빌리어 받은 프로시저의 소유자가 간접적으로 빌린 프로시저에 액세스할 수 있을 뿐이다.

계층적 액세스 제어 모델

이 주제를 체계적으로 다룬 유일한 작업은 와이즈먼[5]의 연구이다. 그는 보안 객체(파일, 사용자, 터미널, 작업)와 객체와 관련된 보안 속성을 정의한다. 속성은 권한(보안 관할구역의 계층적 집합 - 분류), 카테고리(상호 배타적인 보안 관할구역 - 필요에 알맞은 정책의 형식화), 그리고 프랜차이즈(자격증명)이다.

논문의 나머지 부분은 ADEPT-50 시스템에서 채택된 정책을 집합론적 표현으로 개발하는 데 할애된다:

- a) 사용자는 시스템에 접근할 수 있는 경우에만 시스템이 알고 있는 사용자 집합의 구성원일 때만 시스템에 접근이 허용된다.