QQ는 key 생성이 개별로 이루어지는 것이 아닌 service provider로부터 이루어지고 전달되기 때문에 만약 service provider가 공격을 받게 되면 모든 user의 database에도 접근이 가능하다, QQ는 user의 선택에 따라 databases를 나누어서 여러 곳에 저장하는 것이 가능하지만, 대부분의 user가 그 기능을 사용하지 않기 때문에 접근하기 쉽다,

NateOn은 모든 devices에서 동일한 key generation procedure가 이루어지기 때문에 특정 user의 암호화된 char 기록이 주어지면, 해독이 가능하다. 우선 모든 user의 숫자인 Nu는 10자리로 정해져 있어서 brute-force attack을 통한 추측도 가능하고, 만약 user의 phone number가 알려진 다른 user를 친구를 추가하면, 그 user의 Nu를 획득이 가능하다. message들이 담긴 database files들이 고정된 위치에 저장되어 접근하기 쉽다.

KakaTalk의 key를 구하는 과정은 보안이 잘 되어있다. UUID, MN, SN의 경우는 개인 컴퓨터 내에서 밖에 구할 수 없는 요소들이 필요하기 때문에 key를 얻어내는 것은 굉장히 어렵다. 하지만 NateOn과 마찬가지로 message들이 담긴 database files들이 file system의 고정된 위치에 저장되기 때문에 접근하기 쉽다.

KaKaoTalk과 NateOn에서 file system에 저장된 database files에 접근하여 user의 chat messages들이 어느 secret information없이 복원되는 것이 가능하다.

우선 NateOn과 QQ의 경우는 KakaTalk과 같은 방식으로 key를 generation하는 과정으로 바꾸는 것이 좋을 것이다. 접속하는 devices에 따라 다른 UUID, MN, SN을 가지고 있기 때문에 attacker가 key를 획득하기 무척 어렵기 때문이다. 다음으로 3개의 application 모두 messages들이 담긴 database files에 접근하기 쉽기 때문에 저장하는 방법을 바꿀 필요성이 있다.

만약 application에 접속할 때 마다 저장하는 위치를 다르게 해주면, attacker가 원하는 chat log를 찾기 어렵게 될 것이다. 다만, 랜덤으로 배치하게 되면 매번 chat log를 찾는 번거로움이 생길 것이기 때문에, 접속하는 시간과 기기에 따라 어떠한 요소가 server에 전달되면, server에서 종합하여 저장할 위치를 정해주고, 저장했던 위치를 암호화하여 따로 저장하게 되면, 파일 탐색시간도 줄일 수 있을 것이다. 저장위치의 내용을 담은 파일을 attacker가 찾아도 암호화되어 있기 때문에 보안 또한 유지될 수 있을 것이다.