안드로이드에서 클립보드 변조 기법에 대한 분석 및 대응 기법 연구

발표자 : 강해인

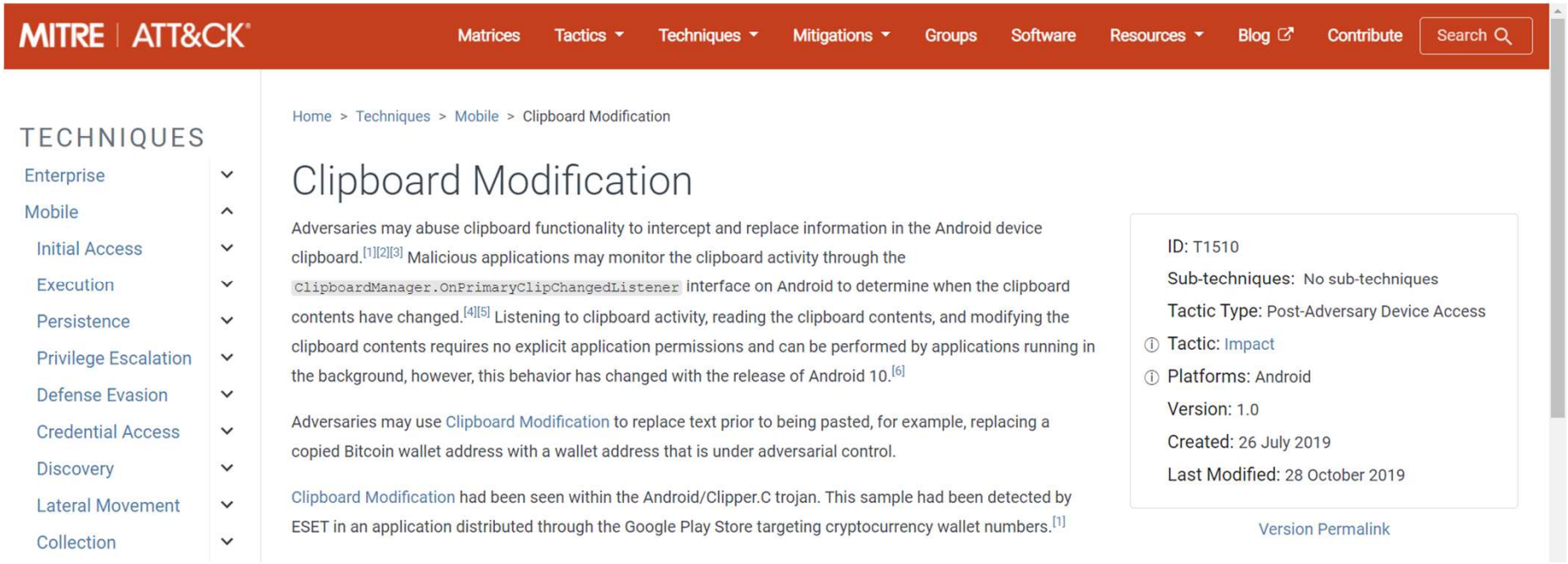
목차

- 연구 소개
- 실제 사례 및 배경 지식
- 연구 목적 및 필요성
- 클립보드 모니터링 및 변조 기법 분석 내용
- 클립보드 모니터링 및 변조 공격 대응 방안
- 결론 및 향후 연구

연구 소개

연구 소개

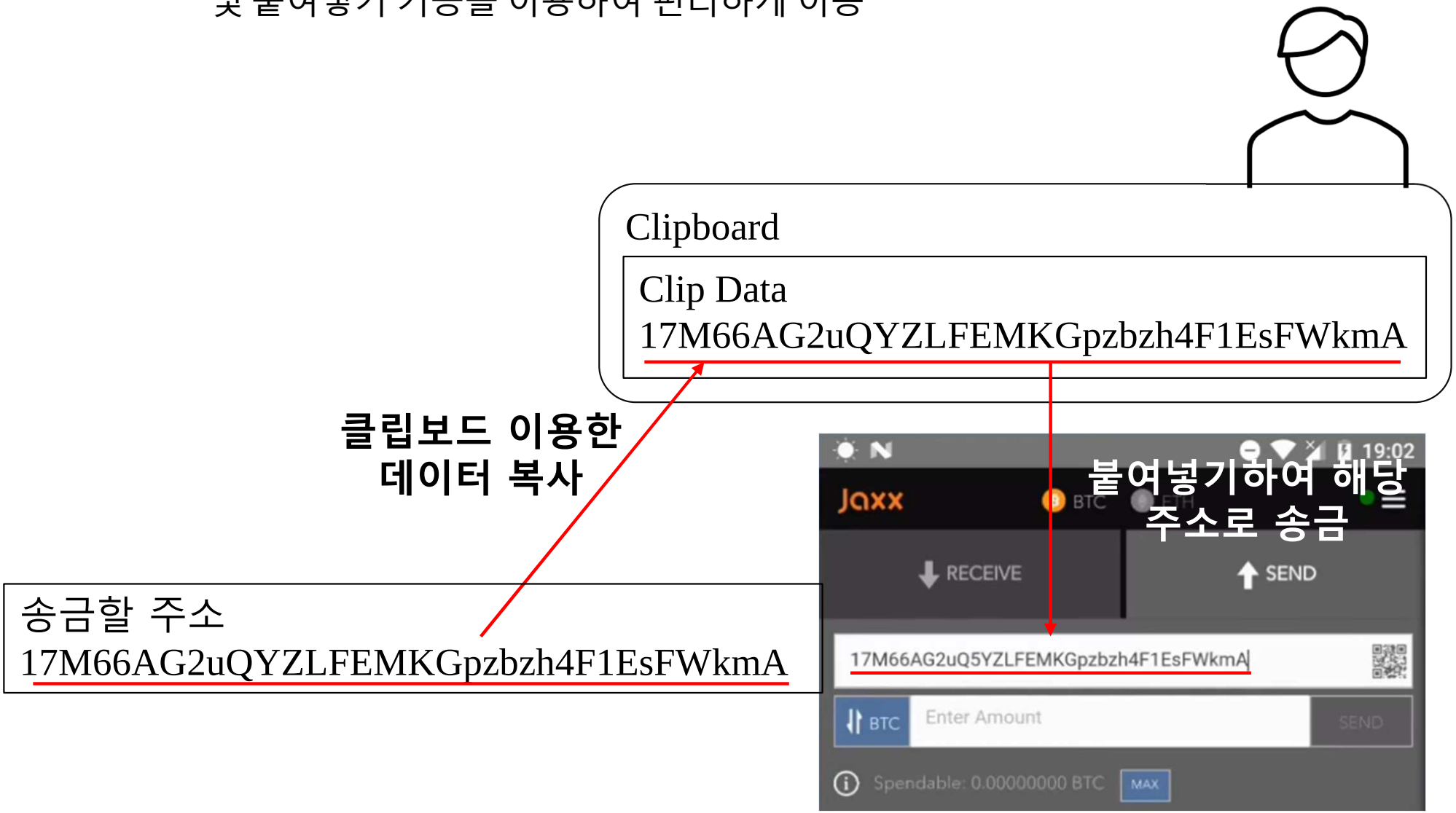
- 클립보드 모니터링 및 변조 기법
  - MITRE ATT&CK가 정의한 적대적 기법으로 모바일 플랫폼에서 제공하는 클립보드 데이터를 가로채거나 변경하는 악성 행위



실제 사례 및 배경 지식

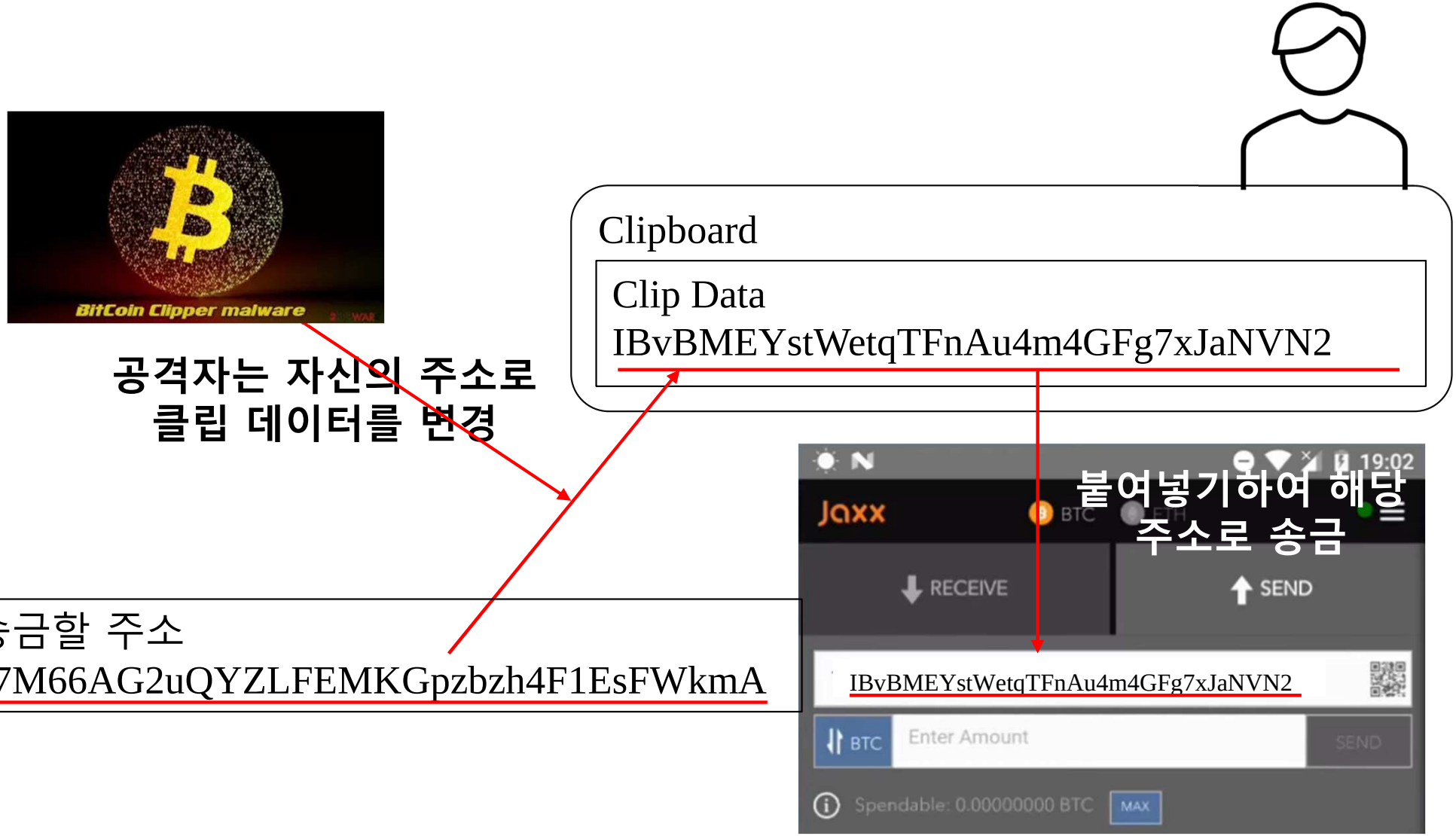
클립보드 모니터링 및 변조 공격 실제 사례 1

- Bitcoin Clipper Malware
  - 정상적인 행위
    - 암호화폐 지갑 주소는 길고 복잡하기 때문에 사용자는 안드로이드 내 존재하는 복사 및 붙여넣기 기능을 이용하여 편리하게 이동



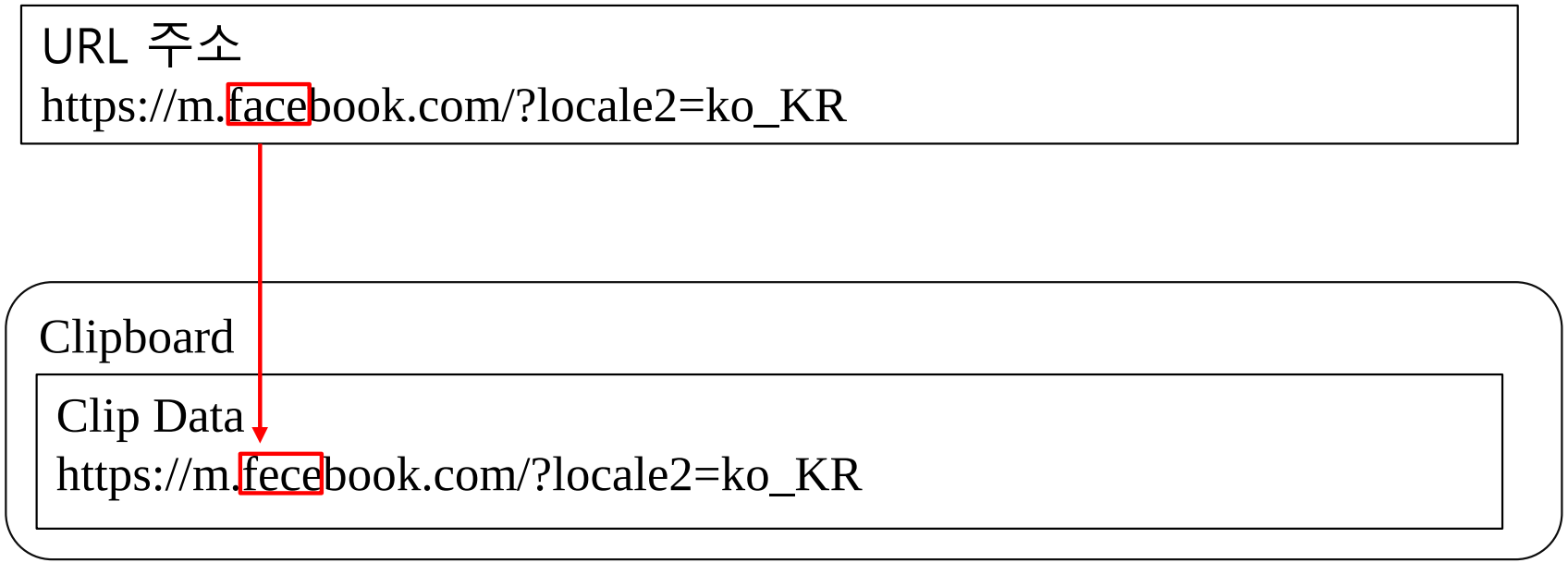
클립보드 모니터링 및 변조 공격 실제 사례 1

- Bitcoin Clipper Malware
  - 암호 화폐 지갑 주소를 모니터링하거나 변조하여 신분을 도용하고 암호 화폐를 탈취하는 악성 애플리케이션



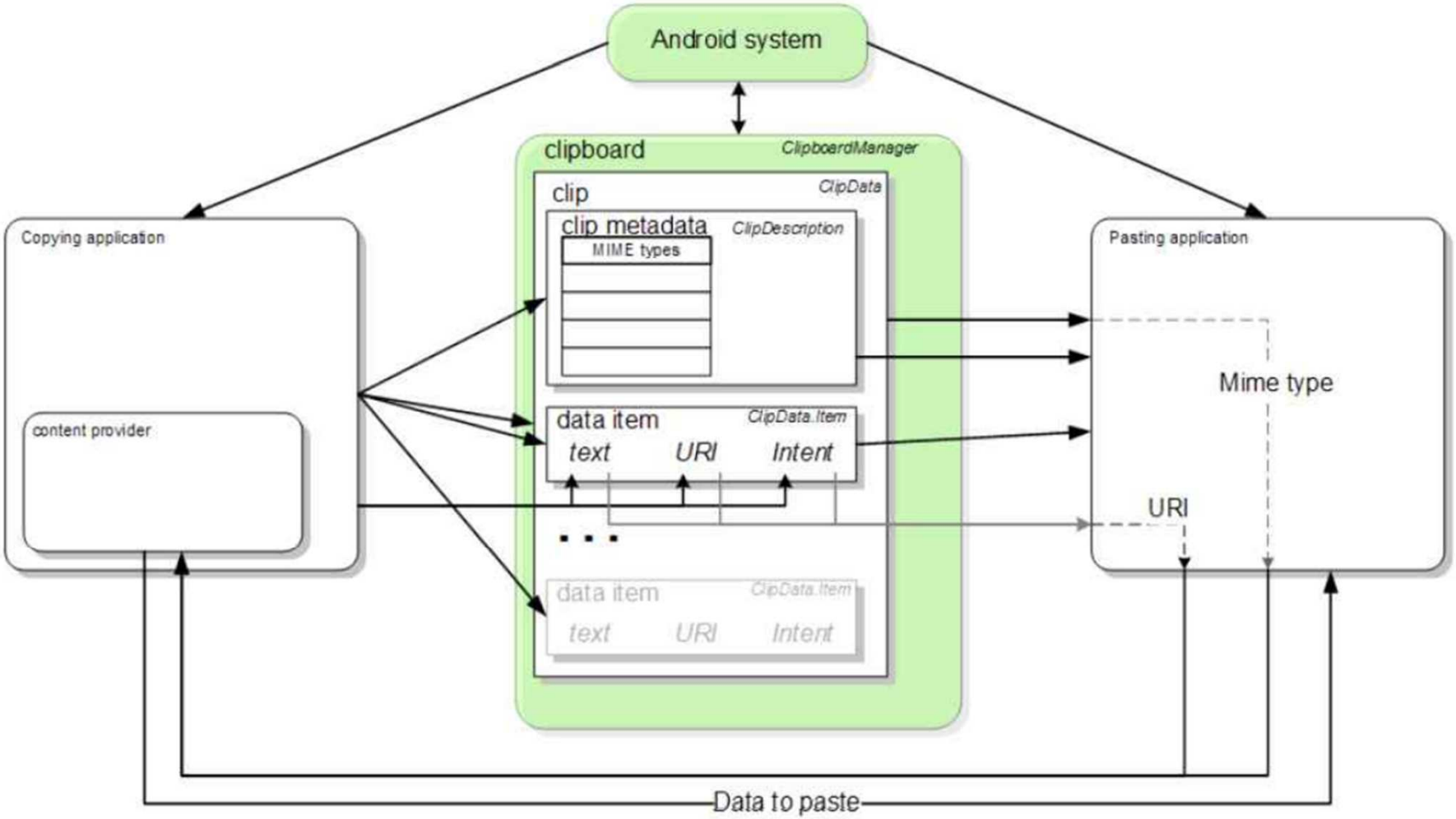
클립보드 모니터링 및 변조 공격 실제 사례 2

- 웹/앱 피싱
  - URL을 교묘하게 변경하고 신뢰성이 있는 사이트처럼 위장
  - 사용자는 로그인을 위해 개인 정보 입력, 공격자는 이를 탈취



클립보드 프레임워크

- Clipboard Manager
  - 클립데이터 : 데이터가 저장되는 공간
  - 클립 데이터에 저장되는 절차에서 클립보드 모니터링 및 변조 행위 발생



연구 목적 및 필요성

연구 목적 및 필요성

- 연구 목적
  - 클립보드 모니터링 및 변조 기법 원리를 분석하여 위험성을 파악하고 공격 대응 방안을 제안하기 위한 목적

- 연구 필요성
  - 클립 데이터에 담기는 내용
    - 민감한 개인 정보 : 데이터 도용, 금전적 손실
    - url : 웹/앱 피싱
  - Google Play 앱
    - 사용자가 개인 정보에 대해 복사 및 붙여넣기 를 수행할 수 있도록 허용
    - 악성 앱의 공격 가능성 증가
  - Java 코드로 쉽게 구현 가능
    - 공격 원리를 파악하고 방어할 필요성이 존재

클립보드 모니터링 및 변조 기법 분석

클립보드 모니터링 기법 및 변조 행위 사전 준비 작업

- 백그라운드에서 동작하는 시나리오
  - Service Component에서 구현
- getSystemService(Context.CLIPBOARD_SERVICE)
  - Clipboard manager 사용
- addPrimaryClipChangeListener
  - 클립데이터 변화를 감지하는 리스너로 사용
- onPrimaryClipChangedListener 인터페이스 구현
  - onPrimaryClipChanged : 데이터 변경될 때마다 콜백되어 사용

```
public class ClipboardService extends Service implements ClipboardManager.OnPrimaryClipChangedListener {

@Override
public void onCreate() {
    super.onCreate();
    cpManager = (ClipboardManager) getSystemService(Context.CLIPBOARD_SERVICE);
    cpManager.addPrimaryClipChangedListener( what: this);
}
```

클립보드 모니터링 기법

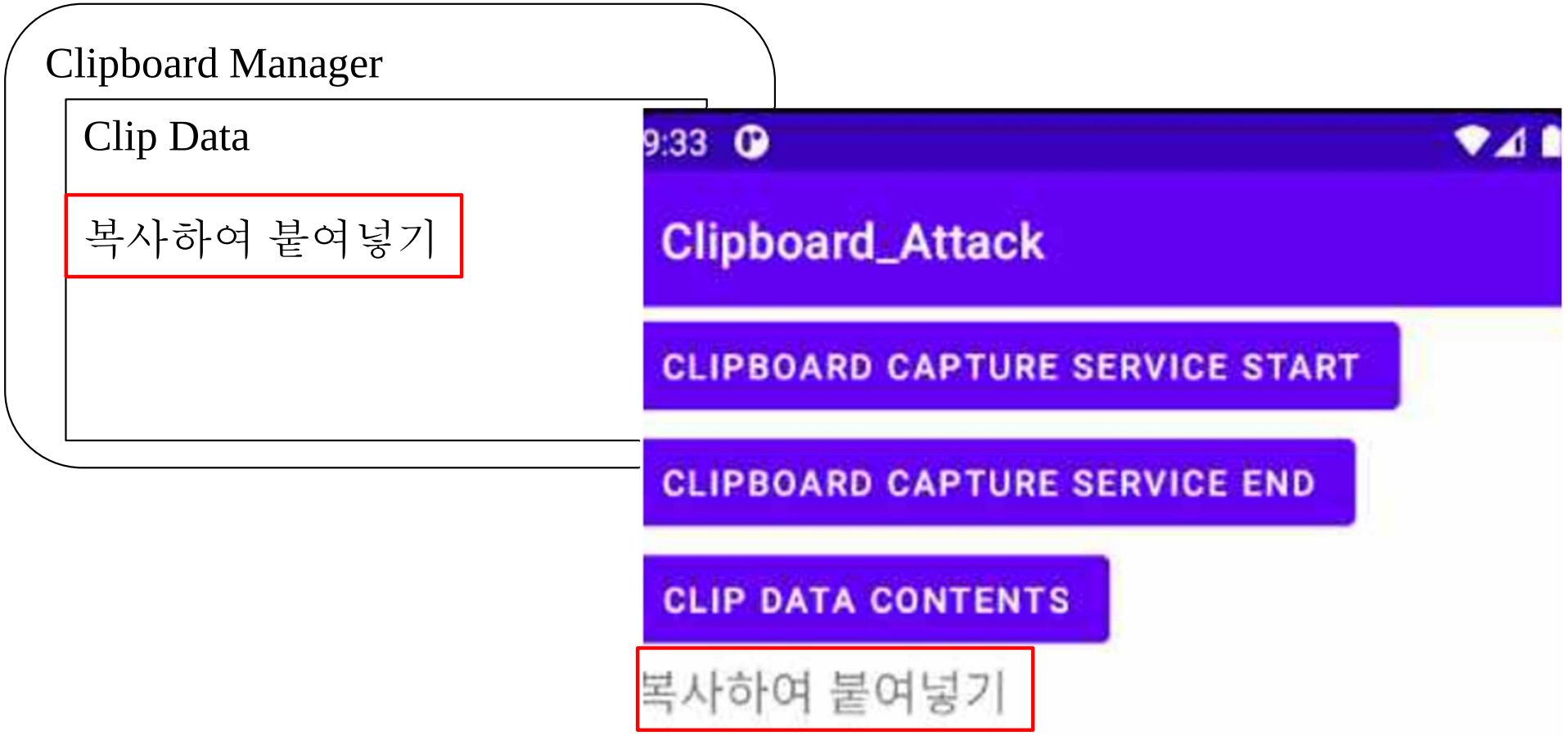
- getPrimaryClip API
  - 클립 데이터 객체 내용을 가지고 옴

- coerceToText API
  - 클립 데이터 객체 내용을 텍스트로 강제 변환시켜 내용 확인 가능

```
@Override
public void onPrimaryClipChanged() {
    if (ClipboardManager != null && ClipboardManager.getPrimaryClip() != null) {
        ClipData data = ClipboardManager.getPrimaryClip();
        int datacnt = data.getItemCount();
        for (int i = 0; i < datacnt; i++) {
            if (data.getItemAt(i).coerceToText( context: this) != null) {
                clipinfo  = " " + data.getItemAt(i).coerceToText( context: this);
            }
        }
    }
}
```

클립보드 모니터링 기법

- 실행 결과 화면
  - 복사 시 모니터링 행위 발생



클립보드 변조 기법

❖ setPrimaryClip API : 클립데이터 저장공간에 들어가길 원하는 데이터를 미리 지정하여 해당 내용으로 변경

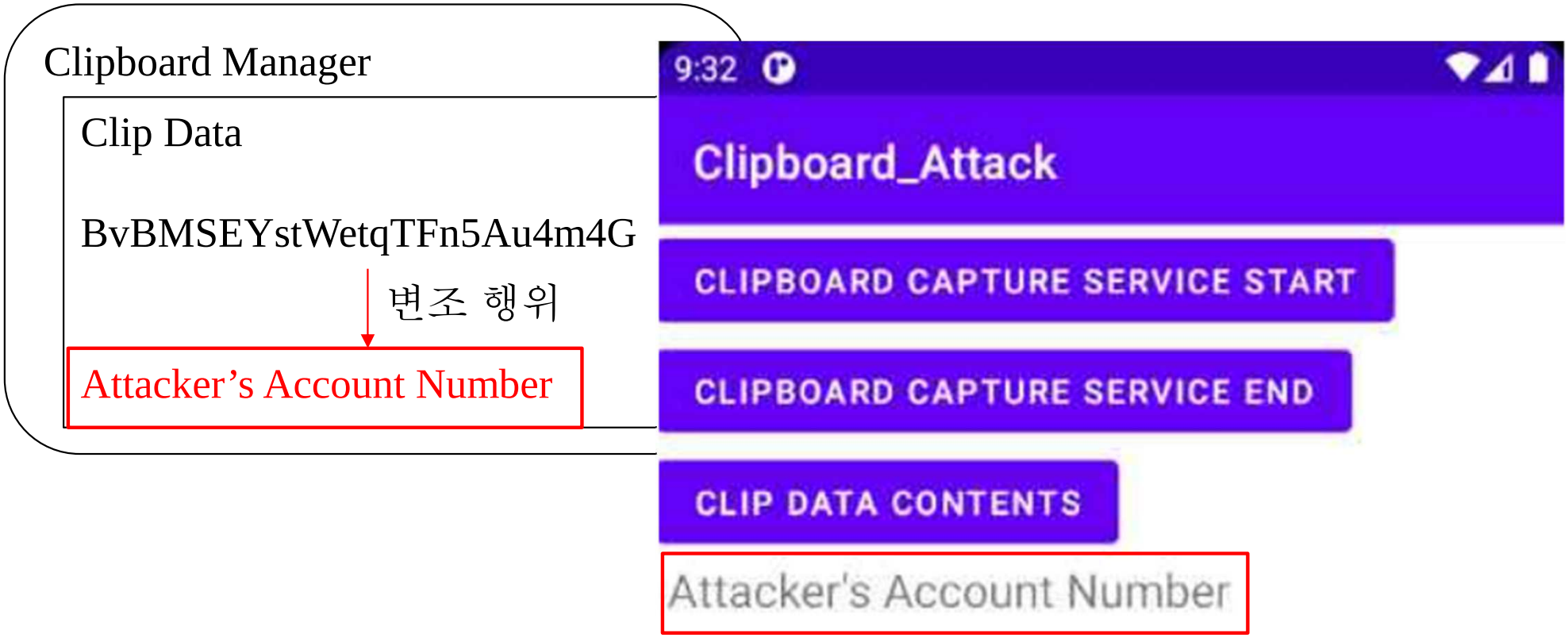
```
public void onPrimaryClipChanged() {
    if(clipcall) {
        //새로운 ClipData 객체로 데이터 복사하기
        ClipData clip = ClipData.newPlainText( label: "T1414&T1510 Attack",  text: "Attacker's Account Number");
        //새로운 클립데이터 객체를 클립보드에 배치
        ClipboardManager.setPrimaryClip(clip);
        clipcall = false;
    }
    if (ClipboardManager != null && ClipboardManager.getPrimaryClip() != null) {
        ClipData data = ClipboardManager.getPrimaryClip();
        int datacnt = data.getItemCount();
        for (int i = 0; i < datacnt; i++) {
            if (data.getItemAt(i).coerceToText( context: this) != null) {
                clipinfo  = " " + data.getItemAt(i).coerceToText( context: this);
            }
        }
    }
}
```

변조

모니터링

클립보드 변조 기법

- 실행 결과 화면
  - 복사 시 변조 행위와 모니터링 행위가 발생



클립보드 모니터링 변조 공격 대응 방안

클립보드 모니터링 및 변조 공격 대응 방안

- AES, DES, RSA와 같은 암호화 기술을 사용하여 클립 데이터를 암호화
  - 제 3자는 복호화하지 못하도록 암호화 키를 관리하는 시스템 도입이 필요
- 복사 시 해시 값과 붙여넣기 시 해시 값을 비교하는 인터페이스 구현
  - 해시 값이 다른 경우, 해당 사실을 사용자에게 알려 위협에 대응
- 클립보드 활성화 시간을 제한하여 공격 가능성 감소
  - 1 Password : AgileBits사가 개발한 암호 관리자로 패스워드에 대한 복사 및 붙여넣기 기능을 제공하는 소프트웨어
  - 보안-사용자 편의성 반영하여 활성화 시간을 45초로 제한

결론 및 향후 연구

결론 및 향후 연구

- 결론
  - 클립보드 모니터링 및 변조 기법에 대해 분석
  - 공격 대응 방안 제시
    - 암호화 기술 사용 클립 데이터 암호화
    - 복사 및 붙여넣기 데이터 해시 값 비교
    - 클립보드 활성화 시간 제한
  - 일반 사용자 대응 방안
    - 최신 OS 사용
    - 민감한 개인 정보에 대해서는 클립보드 내 복사 및 붙여넣기 기능 사용하지 않음
    - 불가피한 경우, 복사한 내용과 붙여넣기한 내용이 같은지 확인
- 향후 연구
  - 클립보드 모니터링 및 변조 공격 대응 방안에 대한 구체적인 연구 진행 예정

Thank You !