

# 热更新API接口

## 安装Tinker(无参)

```
Beta.installTinker();
```

此接口仅用于反射Application方式接入。

## 安装Tinker

```
Beta.installTinker(this);
```

此接口仅用于改造Application方式接入，参数为ApplicationLike对象。

## 用户自定义扩展Tinker接口

```
// or you can just use DefaultLoadReporter
LoadReporter loadReporter = new SampleLoadReporter(getApplication());
// or you can just use DefaultPatchReporter
PatchReporter patchReporter = new SamplePatchReporter(getApplication());
// or you can just use DefaultPatchListener
PatchListener patchListener = new SamplePatchListener(getApplication());
// you can set your own upgrade patch if you need
AbstractPatch upgradePatchProcessor = new UpgradePatch();
// you can set your own repair patch if you need
AbstractPatch repairPatchProcessor = new RepairPatch();
TinkerManager.TinkerPatchResultListener patchResultListener = new TinkerM
anager.TinkerPatchResultListener() {
    @Override
    public void onPatchResult(PatchResult result) {
        // you can get the patch result
    }
};

Beta.installTinker(this, loadReporter, patchReporter, patchListener, patc
hResultListener, upgradePatchProcessor, repairPatchProcessor);
```

此接口仅用于改造Application方式接入。

## 加载指定路径的补丁

```
Beta.applyTinkerPatch(getApplicationContext(), Environment.getExternalStorageDirectory().getAbsolutePath() + "/patch_signed_7zip.apk");
```

### 加载Arm架构so库

```
Beta.loadArmLibrary(getApplicationContext(), "mylib");
```

只能用于加载armabi架构的so文件。

### 加载Arm-v7架构so库

```
Beta.loadArmV7Library(getApplicationContext(), "mylib");
```

只能用于加载armabi-v7a架构的so文件。

### 通用加载so库接口

```
Beta.loadLibrary("mylib");
```

通用加载so接口，我们自动帮你加载指定架构的so。

### 清除补丁

```
Beta.cleanTinkerPatch();
```

注：清除补丁之后，就会回退基线版本状态。

### 主动检查更新

```
Beta.checkUpgrade();
```

用于主动检查补丁策略的接口。