移动系统漏洞攻防：永恒战争中的持续创新

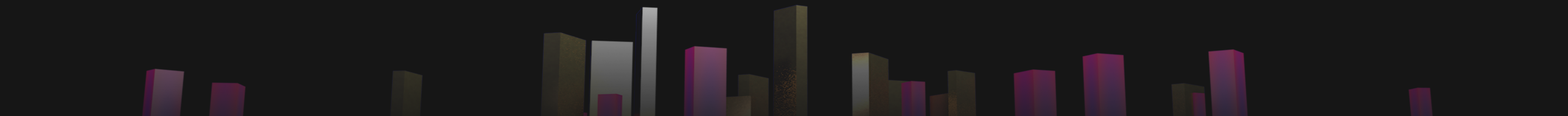
宋杨 Pandora Lab of Ali Security

1 远古 / 代码签名

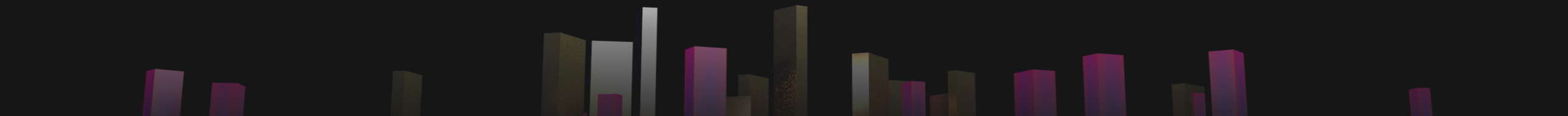
2 近代 / 沙箱机制

3 现代 / 缓解机制

4 未来 / 永恒战争



1 远古 / 代码签名



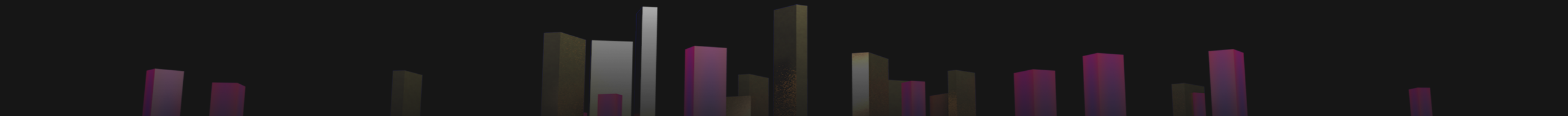
远古

在Android 4.x之前：

- App Sign
- Permission
- DAC
  - root/system/privileged/unique app uid/gid

这是一个幸福的时代

zergRush / Gingerbreak / …



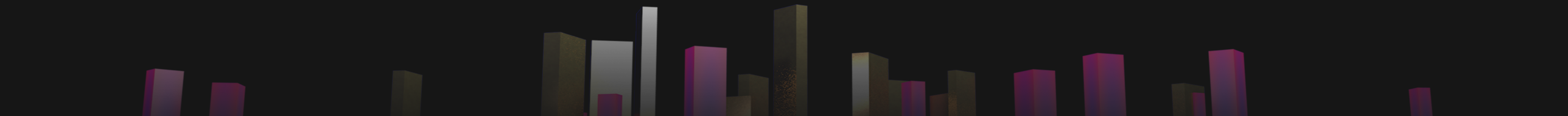
代码签名

Android

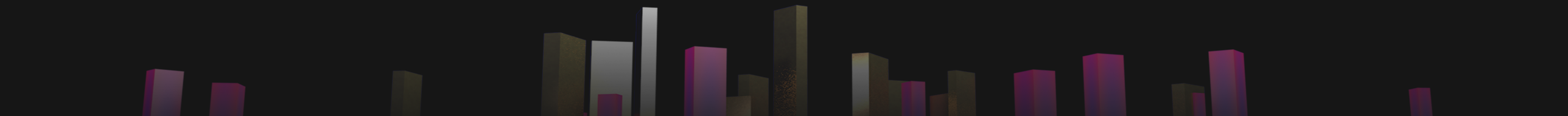
- 开发者自行生成并管理签名密钥
- 在App安装 / 加载时验证签名

iOS

- 加入开发者计划，正式发布时由Apple签名；
- Planform binary签名保存在trustcache中；
- 没有签名的Code无法执行，也不能在运行时动态变更
- Team ID控制App不能加载一个第三方的动态库；（iOS 8引入）



2 近代 / 沙箱机制



近代

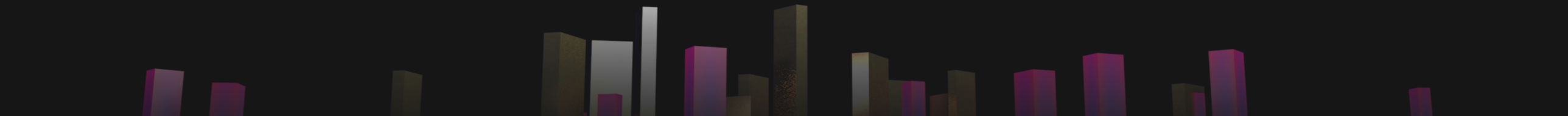
分水岭： Android 4.4 将SELinux状态从 “Permissive”改为“Enforce”模式

Discretionary Access Control

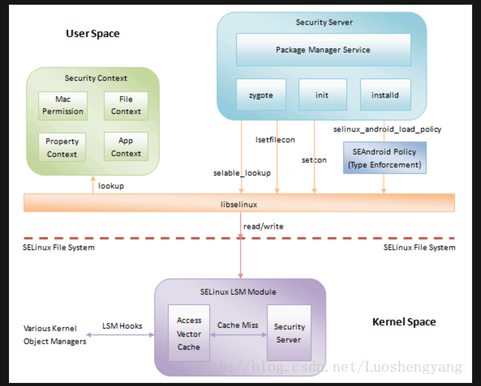
- root/system/privileged/unique app uid/gid
- root/mobile

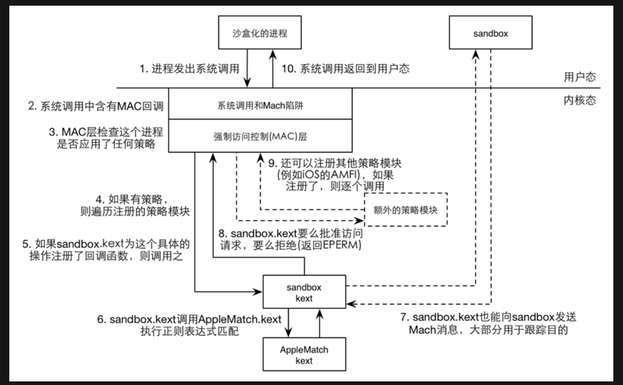
Permission / Entitlements

- 主要解决对系统服务调用的权限控制问题
- Entitilements不完全对应Permission



沙箱机制





图片来源：深入解析Mac OS X & iOS操作系统

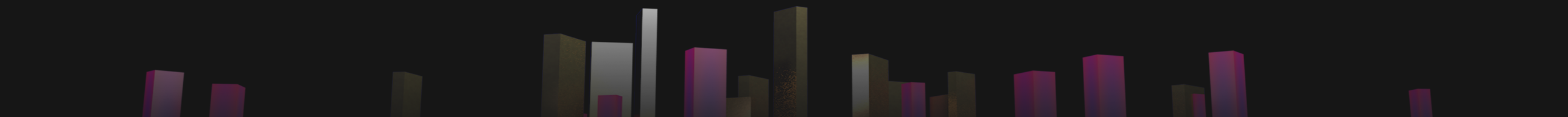
图片来源：http://blog.csdn.net/luoshengyang/article/details/37613135

Mandatory Access Control

- TrustedBSD Mac Framework
- SElinux / SEAndroid

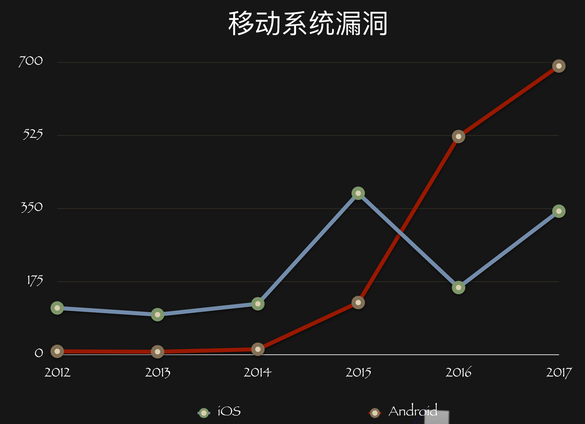
- MAC的使能让Android完善了现代操作系统安全机制的重要一环
- 开启了追赶iOS安全的序幕
- 从攻击者角度的区别

3 现代 / 缓解机制



现代

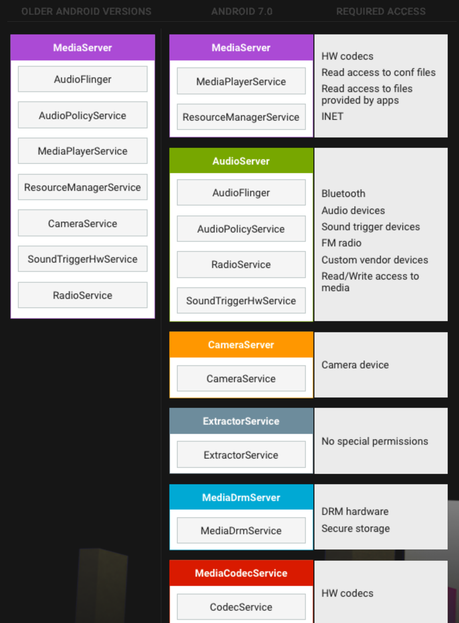
分水岭：Android持续大规模减少攻击面



典型攻击面

- Camera group
  - 2013下半年，Android 4.4
  - 2015上半年，大规模部署
- Stagefright / Media
  - Blackhat USA 2015
  - 成为持续爆出系统漏洞的高危地带
- 数据来源：CVE details

利用缓解



- Android 7.0 Media Framework Hardening
- SECCOMP
  - 最早在Android 5.0引入支持（ARM）
  - Android 7.x引入到特权进程
    - media stack hardening
  - Android 8.0引入到普通App
    - 通过zygote启用
- 图片来源：https://source.android.com/devices/media/framework-hardening

利用缓解

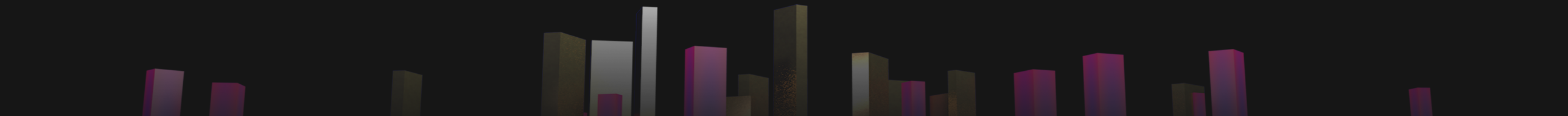
Android:isolatedProcess

- 针对Service，由应用自愿使用；
- Service跑在一个无权限的独立的进程中；

iOS dynamic-code signing entitlement

- JIT使用，其他进程不能分配一块eXecute的内存
- Bulletproof JIT (iOS 10)

```
sailfish:/ $ ps -eZ | grep chrome
u:r:untrusted_app:s0:c512,c768 u0_a70       11585   595 1843376 143136 SyS_epoll_wait      0 S com.android.chrome
u:r:isolated_app:s0:c512,c768  u0_i1        12298  9702 1773344  87920 SyS_epoll_wait      0 S com.android.chrome:sandboxed
u:r:untrusted_app:s0:c512,c768 u0_a70       12336   595 1737704  62340 SyS_epoll_wait      0 S com.android.chrome:privileged_proces
```



利用缓解

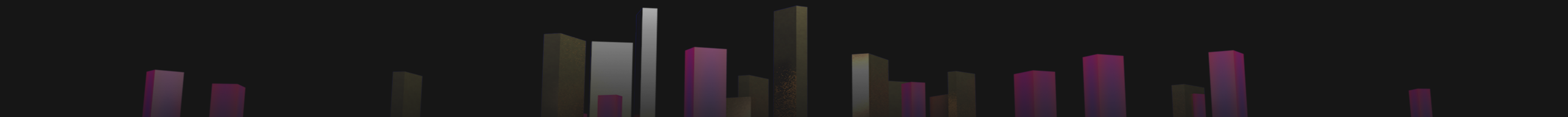
PXN(Privileged eXecute Never)

- ret2user
- ROP/set_fs(KERNEL_DS)/ret2dir/VDSO

PAN(Privileged Access Never)

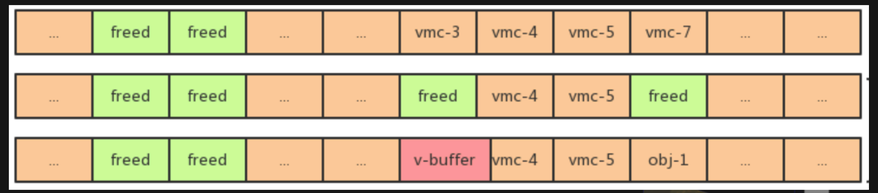
- Stack pivot
- Bypass PAN emulator?

```
"date" : "2017-11-09 17:30:33.80 +0800",
"panicString" : "panic(cpu 0 caller 0xffffffff0269dd9fc): Privileged access never abort.
```



利用缓解

- Heap Free Element Protection
  - 填充poison信息，头和尾填入内核才知道的值，alloc的时候检查
- Random Heap Element Address
  - 在从一个新的page中创建zone的时候，元素随机化
- Free into wrong zone
  - 检查堆元素页对应的zone，如果不符合就panic



- Anti task port abuse
  - 通过一个IPC framework的漏洞，可获得手机任意进程的task port，获得任意进程内存的任意读、写、执行能力。
  - https://bugs.chromium.org/p/project-zero/issues/detail?id=1247&can=1&q=apple&start=200

- Anti Kernel GC
  - CVE-2016-7644：XNU kernel UaF due to lack of locking in set_dp_control_port
  - 无法通过mach_zone_force_gc接口直接触发kernel GC。对抗kernel UAF漏洞，无法通过GC填充新内容，从而无法构造type confuse。

利用缓解

Stack Canaries

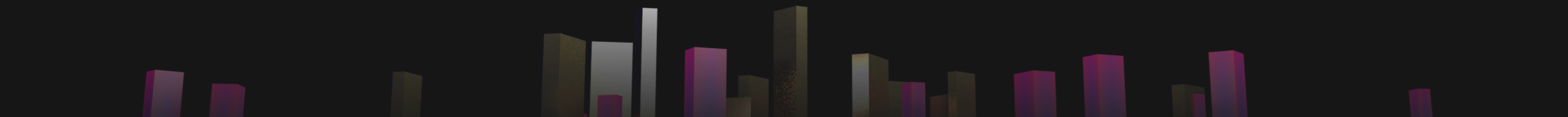
ASLR/KASLR(Android 8.0)

KPP/RKP

Hardened Usercopy

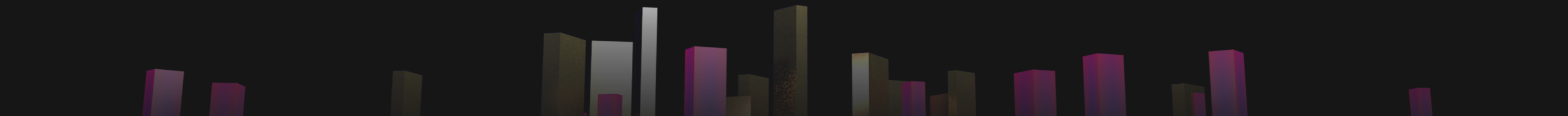
Post-init read-only memory

……



4

未来 / 永恒战争



未来 / 永恒战争

iPhone X / iOS 11.1

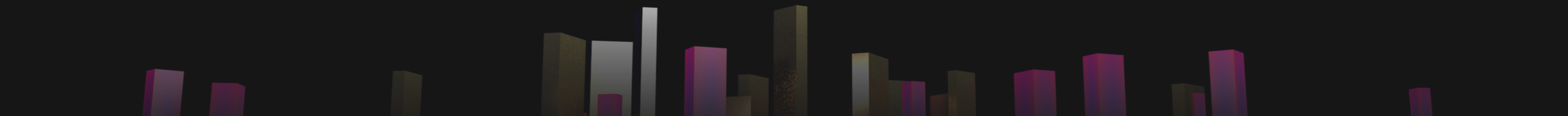
Google Pixel / Android 8.0

随着移动系统的不断发展，系统安全持续提升

漏洞总是会存在的，针对不同的攻击面，针对不同的安全机制

如果一个漏洞搞不定，那就用两个

攻守双方互相促进，这是一场永恒的战争



THANKS

THANKS

THANKS

THANKS

THANKS