

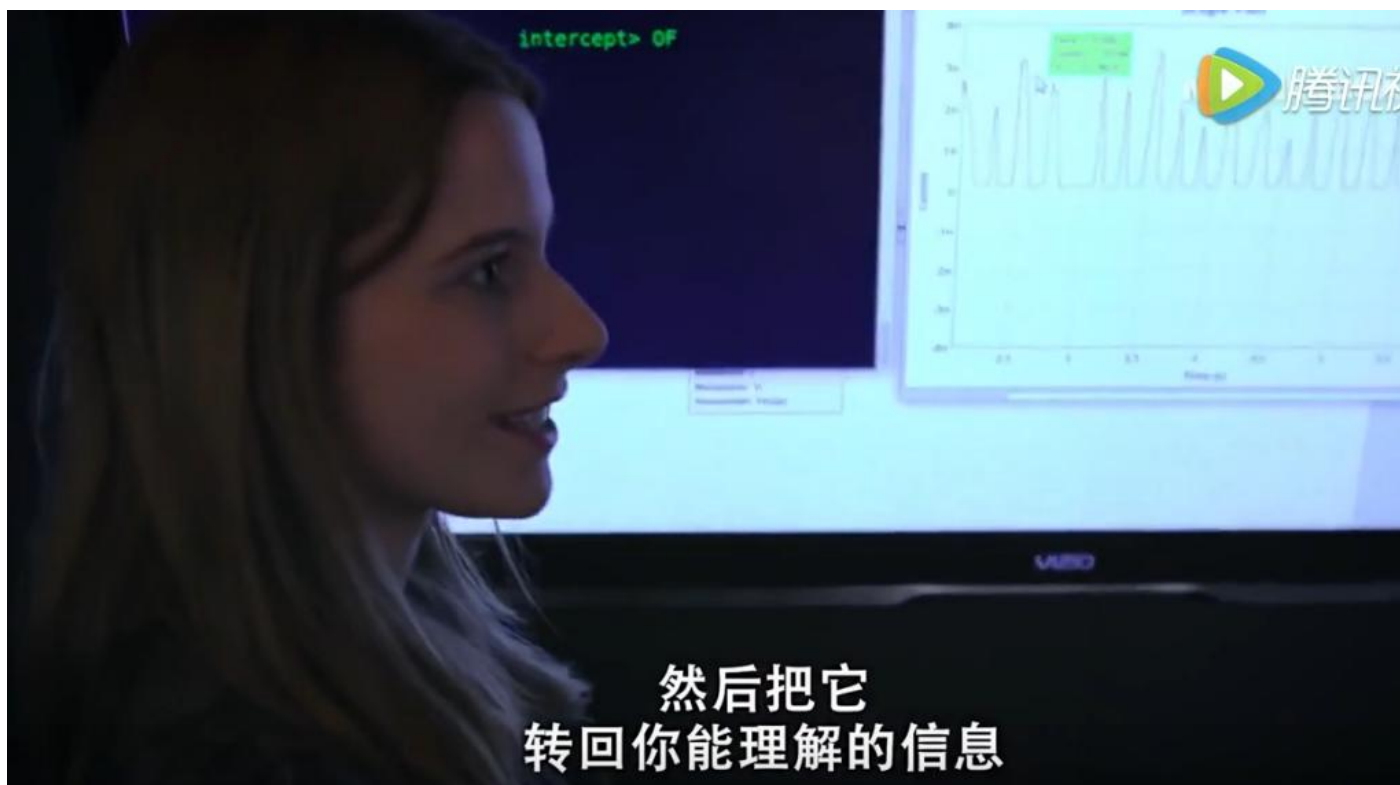
## 《安全团队的商业困境与黑客精神》

by: roysue

首先是群里有大牛发了这样一则从“今日头条”引用过来的视频，半夜发的，没怎么注意看。早上起床之后看了一眼。觉得还是很有意思的。视频地址在这里：  
<https://v.qq.com/x/page/o0357o5n9om.html>



哪里有意思呢？首先当然是美女主播啦！



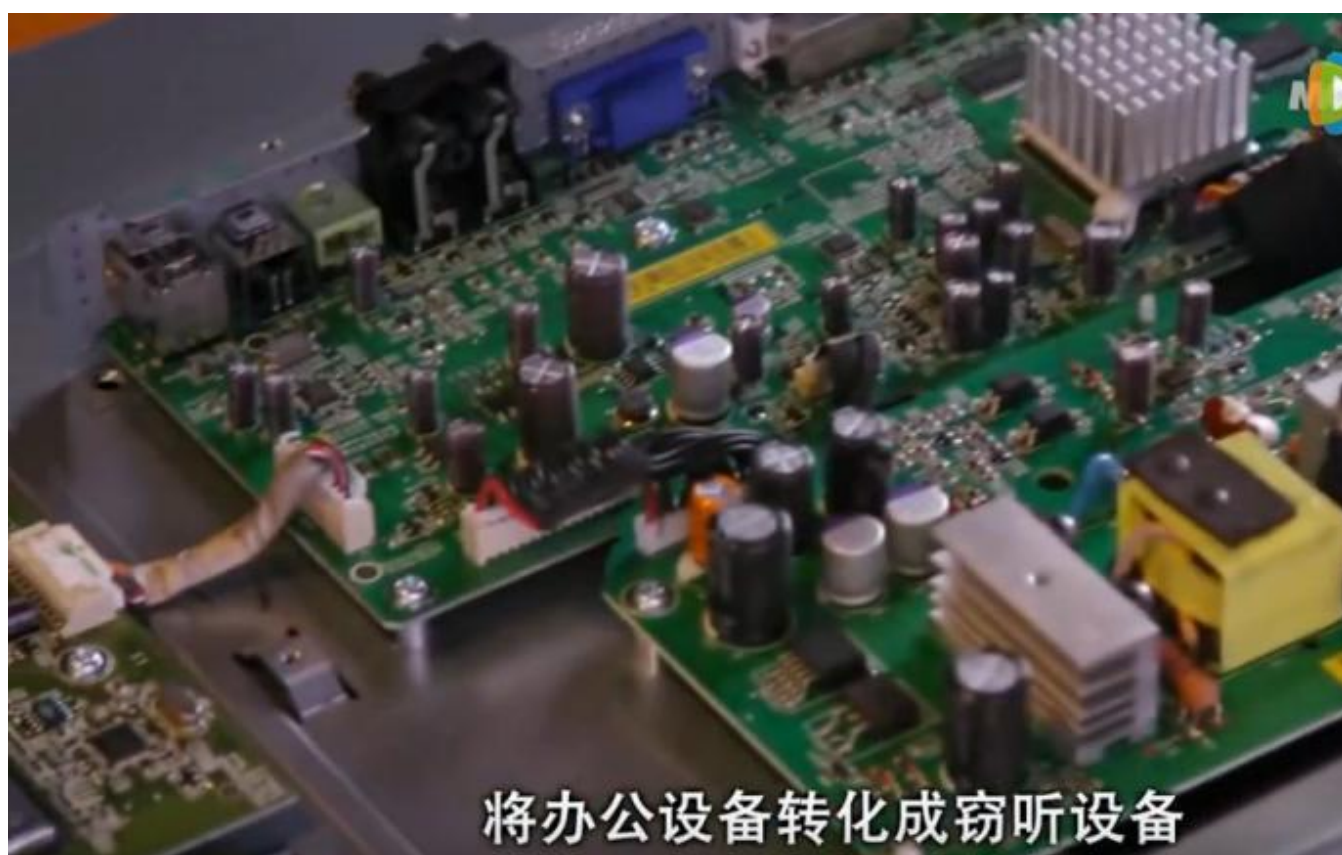
然后，还有各种高大上的设备，让人流口水的 Boss 办公环境，员工办公环境，而且是在 manhattan，曼哈顿主城。不得不说这样的公司我也想去。



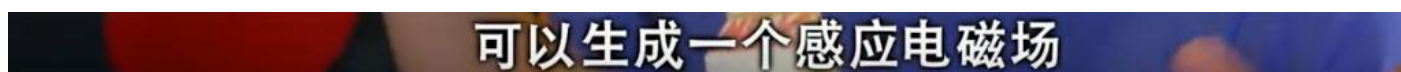
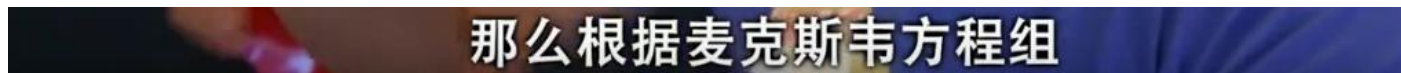
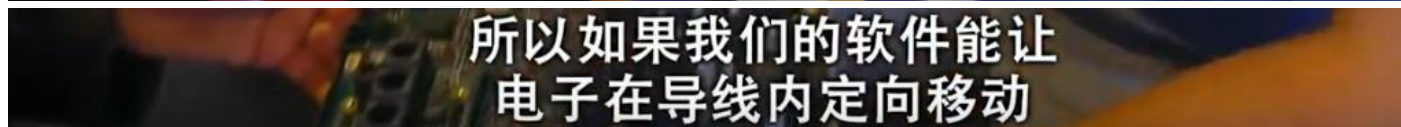
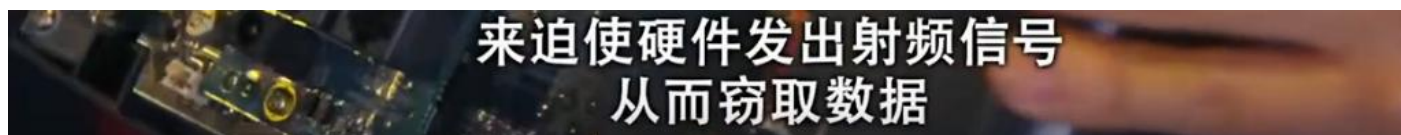


然后就是美女主播介绍这次 interview 的主题——对嵌入式设备的安全研究，Ang Cui（是不是很像是一个中国人的名字？像 An Li，李安），测试过程中的失败很多，大概有好几十部电话机拆散了扔在了“电话机墓”里。崔安的研究是利用无线电技术，将工业电路集成设备变成窃听器。





原理他也说的很清楚：





也就是经典的“特雷门琴”的工作原理，也就是二战末期苏联窃听美国大使馆的“金唇”的工作原理。这款窃听器的厉害之处，在于它不需要电源，“一切窃听器都需要电源。”这个间谍届貌似不可颠覆的真理，被苏联人打破了。苏联人制作了一个结构简单，体积微小，没有电池，也无需外接电流的窃听器，并故意命名为“这玩意”（THE thing）。因为没有电，当时的反窃听设备无法捕捉到任何信号，300 米以内大耗电量振荡器所发出的微波脉冲都能够被“这玩意”捕捉到，更奇特的是它的工作寿命是无限的，因为其不需要电源。而特雷门琴也是至今为止世上唯一一款不需要身体接触就能发音的乐器，所使用的也就是无线电反射的原理。

好了，我们继续：



把输入信号转换成输出信号

就能让座机  
在非通话时间保持通话状态

简单来说就是把窃听器

装入某个会议室  
或者任意一个有座机的地方

没错，靠软件来完成

所以不需要潜入房间  
不需要在电话上做手脚

只要通过网络就能实现

也就是说：

- 1，主要还是依靠一个 0day，改写固件。将 line in 转化成 line out，使之可以持续 24 小时拾音，；
- 2，然后利用特雷门琴射频的原理，将其拾音的频率传送出去；
- 3，这样外面的收音机接受到信号之后做滤波处理，将声音还原出来。

我们继续，短短的七分钟视频，内容还是很多的：



就是给目标发一封简历

用它重写打印机的固件  
以便我们操纵

我们就可以找出所有的座机

一旦找到这些座机  
这部打印机就会去

入侵一切不设防的座机

然后我们就能监控每个房间

在这里又刷新了我们的认知：

1，用简历改写固件？That`s impossible！简历上肯定带有畸形字符，携带某些可供打印机识别的工控信息；

2，打印机识别到指令之后，将会到远程主机更新固件，工控集成设备很难做到 HTTPS 或者证书绑定(Cer Pinning)，只要路由方向的任意一台主机被控制(ISP 劫持)，既可以实施中间人攻击，或者 DNS 投毒也可以；

3，打印机在更新固件的时候，事实上是在下载和更新由攻击者控制的包含木马后门的固件，这样完成了打印机的控制；

4，一般情况下打印机和电话机都是由电话线连接在一起的，这样由打印机横向渗透进入电话机。Trigger the 0day，然后电话机都变成了监听机；

5，这些电话应该是 ip 电话，由 ipv4 地址，然后很像扫描所有内网主机。跟“僵尸病毒”是一样的模式。

我们已经实现了  
语音文字的实时转换  
we were able to do live speech-to-text.

我们可以让座机把内容  
直接转成推文发到Twitter上  
We just had the phone tweet  
everything that was being said.



然后他们还做到了：

- 1，语音到文字的实时转换。
- 2，将办公室的内容实时用文字发布到 twitter。

智能语音识别早就不是什么新鲜的话题，不信你可以去搜索一下李开复博士的简历，从上个世纪八十年代起，他就在从事这个方向的工作。而接近三十年今后的今天，前几天看到的一则新闻中，将专攻智能语音识别的“科大讯飞”放在中国最聪明公司的第一位，排名在腾讯（大数据）、旷视科技（人脸识别）、大疆（无人机飞控）、富士康（智能制造）、阿里巴巴（商业云）等业界巨擘之前。也不知道是该高兴，还是应该感到悲哀。

而且专攻人工智能、自动驾驶的百度，将人工智能和自动驾驶当作公司的“愿景”和“使命”的百度，只排在第五十位。

新闻如下：

6月27日，2017年《MIT 科技评论》评选“全球最聪明50家公司”的榜单在北京全球首发。有9家来自中国，分别是科大讯飞（第6位，002230.SZ）、腾讯（第8位，00700.HK）、旷视科技（第11位）、大疆（第25位）、富士康（第33位，02038.HK）、阿里巴巴（第41位）、HTC（第42位）、蚂蚁金服（第49位）、百度（第50位）。

当然，科大讯飞的成就我们有目共睹，“该公司旗下的语音助理技术是中国版的Siri。其实时翻译技术则是杰出的人工智能应用，克服了方言、俚语和背景杂

音，可将汉语精准地翻译成十几种语言。”在中文识别方向，其成就无出其右，如果有不服的，可以下载他们的 App 试用看看，当然锤子的语音输入，也是用的他们的 API。


我们继续：




先前展示的是Funtenna软件如何  
让打印机通过射频信号



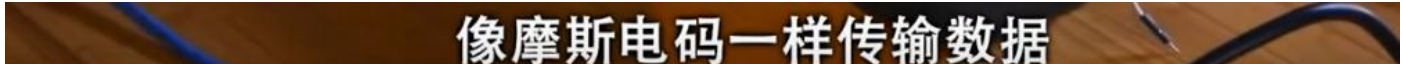
泄露信息致使机密外泄的



他用一根天线就能接收这些情报



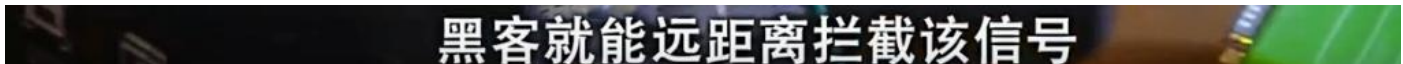
这个漏洞软件  
能让打印机的部件震动



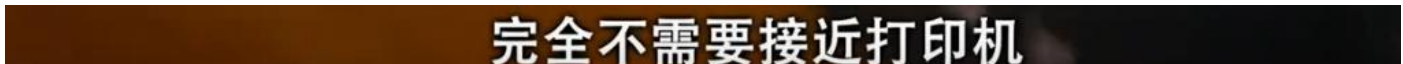
像摩斯电码一样传输数据



只要一根接收能力够强的天线



黑客就能远距离拦截该信号



完全不需要接近打印机

接下来就是有点 low 了，还要给打印机加装一个振动器，让打印机的数据也泄漏出去。这么大的振动器，只要任何办公室人员眼睛没瞎，都看得见。哪怕如果做成内置的，军工采购也不可能通过，也就是说后面的几乎都是 balabalabala 了。更别提军工周围的无线电都是严格管制状态，通过无线电把数据传输出去的思路，还不如通过网络呢。



不同类型的嵌入式设备来  
发送任何想传输的数据

随心所欲

加装振动器和调制器之后，当然任何设备都能把信号发射出去了，这个就跟家里的电视机没什么区别了。把调制解调器当作新的技术，这就有点荒谬了。

总结一下，视频里的技术深度，前面特雷门琴的部分可以打 60 分吧，在专业玩 SDR (Software defined radio, 软件定义无线电) 的 360 团队看来可能还给不到 60 分，后半部分竟然还要加装部件，只能给 20 分。

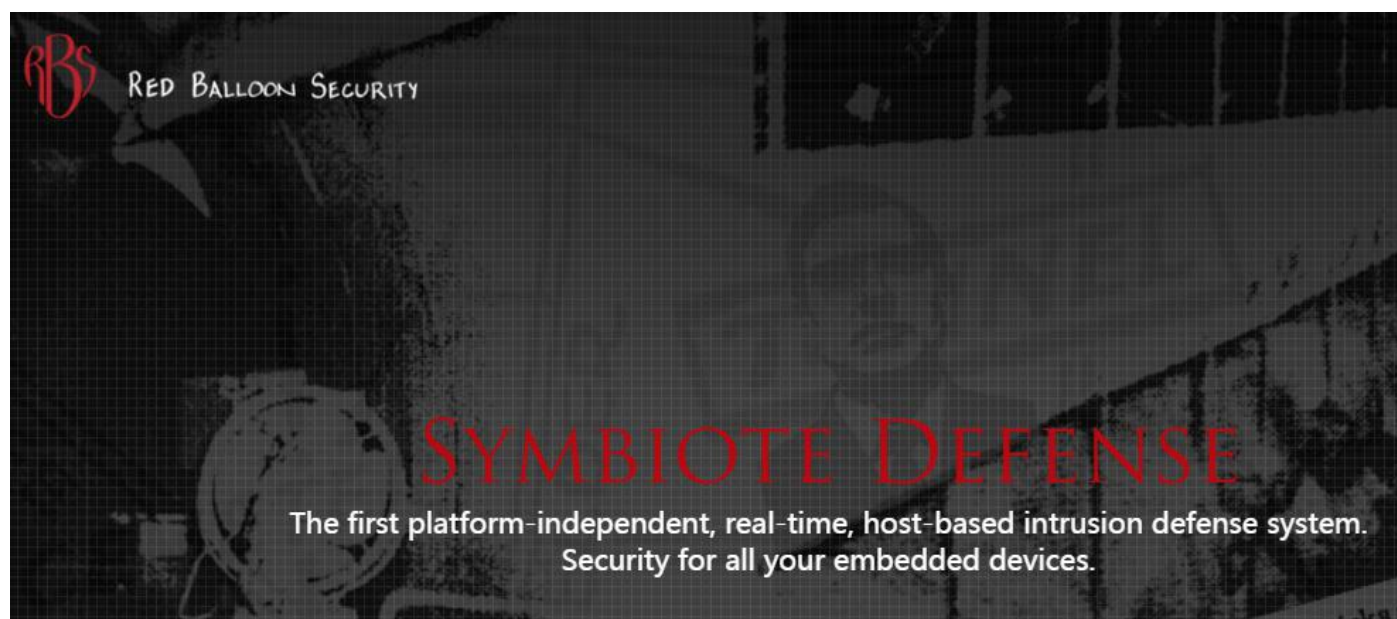


然后就想着，在 Manhattan 曼哈顿主城租这样一个办公室，还带四五个 employee，其实也是一笔不小的开支，大家知道美国的 CS 非常的贵，一般 Intern 都要给到接近八、九万美金一年，如果是全职，那就更加贵了。后来搜索了一下 red balloon security 的官网，大概是连 Boss 在内一共七个人。



按照视频里的曼哈顿主城办公室加上七个人一年的薪水，一年的开销怎么也得接近百万美金，如果放在中国，不管是北京还是上海，深圳还是广州，哪怕是杭州，两、三百万一年也肯定是需要的。

他们的主营业务是什么呢？



# PROJECT SYMBIOTE

The First Universal Embedded Defense for all embedded devices

Cyber-security threat actors today are shifting to the lowest hanging fruit. Most networked devices shipping today are not desktops, laptops or servers and none of them have strong host-based defense. Your automotive, point-of-sale, unified communications, Internet-of-Things, SCADA, home and office equipment are highly vulnerable and are actively being compromised today, whether for corporate espionage, financial fraud, or state-to-state cyber warfare. Red Balloon Security is devoted to hardening all devices against malicious intrusion.

Device manufacturers can now inject Symbiote Defense into any device regardless of CPU type and operating system. No hardware or source code modifications required.



Contact us for more information. Stay tuned for big news.

UNCLASSIFIED//INFOSEC//REDBALLOONSECURITY//REL TO NYC, USA

恕我直言，没看懂。

Device manufacturers can now inject Symbiote Defense into any device regardless of CPU type and operating system. No hardware or source code modifications required.

设备制造商可以将 Symbiote Defense 注入到任何 Iot 设备中去，不管是哪种 CPU 或者什么系统都可以，软硬件都无需更改。

这...是一个硬件？还是一个软件呢？还是...什么呢？

刚刚视频里的小蓝盒子么？也不像吧，那不是个 modern 么？

不懂。

大学生想要炫技可以理解，虽然没有任何独创和突破，但是“我把课本学习的很好呀！”“做了这么多实验，我可是完整的掌握了实验技能呢！”，“凭借这些技能就能改变世界了喔！”所以我要开个公司，还拿到了“Microsoft Ventures Accelerator”的奖学金，其实已经算是不错的哥大毕业生了，可能还在 Top10 或者 Top5 里面。

然后继续谷歌崔安这位大帅哥，果然硕果累累，研究方向基本上都是工控集成电路和智能硬件固件篡改的方向，嗯，有为青年！





Ang Cui

Columbia University  
Computer Security

在 cs.columbia.edu 的电子邮件经过验证

关注

Google 学术搜索

创建我的个人资料

标题 1-16	引用次数	发表年份
<b>When Firmware Modifications Attack: A Case Study of Embedded Exploitation.</b> A Cui, M Costello, SJ Stolfo NDSS	82	2013
<b>A quantitative analysis of the insecurity of embedded network devices: results of a wide-area scan</b> A Cui, SJ Stolfo Proceedings of the 26th Annual Computer Security Applications Conference, 97-106	61	2010
<b>Concurrency Attacks.</b> J Yang, A Cui, SJ Stolfo, S Sethumadhavan HotPar 12, 15	29	2012
<b>Defending embedded systems with software symbiotes</b> A Cui, S Stolfo Recent Advances in Intrusion Detection, 358-377	29	2011
<b>Brave new world: Pervasive insecurity of embedded network devices</b> A Cui, Y Song, PV Prabhu, SJ Stolfo International Workshop on Recent Advances in Intrusion Detection, 378-380	27	2009
<b>Usable, secure, private search</b> M Raykova, A Cui, B Vo, B Liu, T Malkin, SM Bellovin, SJ Stolfo IEEE Security & Privacy 10 (5), 53-60	14	2012
<b>Ethics in security vulnerability research</b> AM Matwyshyn, A Cui, AD Keromytis, SJ Stolfo IEEE Security & Privacy 8 (2)	14	2010
<b>Symbiotes and defensive mutualism: Moving target defense</b> A Cui, SJ Stolfo Moving Target Defense, 99-108	11	2011
<b>Killing the myth of Cisco IOS diversity: recent advances in reliable shellcode design</b> A Cui, J Kataria, SJ Stolfo Proceedings of the 5th USENIX conference on Offensive technologies, 3-3	9	2011



点开其第一篇论文,也就是引用数高达82次的论文,果然找到了通过 standard printed document 来篡改打印机固件的。而且,还是用的别人(第三方)的漏洞。

# When Firmware Modifications Attack: A Case Study of Embedded Exploitation

Ang Cui, Michael Costello and Salvatore J. Stolfo  
Department of Computer Science  
Columbia University  
New York, US  
{ang, costello, sal}@cs.columbia.edu

**Abstract**—The ability to update firmware is a feature that is found in nearly all modern embedded systems. We demonstrate how this feature can be exploited to allow attackers to inject malicious firmware modifications into vulnerable embedded devices. We discuss techniques for exploiting such vulnerable functionality and the implementation of a proof of concept printer malware capable of network reconnaissance, data exfiltration and propagation to general purpose computers and other embedded device types. We present a case study of the HP-RFU (Remote Firmware Update) LaserJet printer firmware modification vulnerability, which allows arbitrary injection of malware into the printer's firmware via standard printed documents. We show vulnerable population data gathered by continuously tracking all publicly accessible printers discovered through an exhaustive scan of IPv4 space. To show that firmware update signing is not the panacea of embedded defense, we present an analysis of known vulnerabilities found in third-party libraries in 373 LaserJet firmware images. Prior research has shown that the design flaws and vulnerabilities presented in this paper are found in other modern embedded systems. Thus, the exploitation techniques presented in this paper can be generalized to compromise other embedded systems.

**Keywords**—Embedded system exploitation; Firmware modification attack; Embedded system rootkit; HP-RFU vulnerability.

security of our existing networks, we present the following four contributions:

**General firmware modification attack description:** We present firmware modification attacks, a general strategy that is well-suited to the exploitation of embedded devices. This strategy aims to make arbitrary, persistent changes to victim devices' firmware by leveraging design flaws commonly found within embedded software. Firmware modification attacks can affect entire families of devices adhering to the same system design flaw, transcending operating system versions and instruction set architectures. The HP-RFU vulnerability presented in this paper affects MIPS- and ARM-based printers alike, regardless of their underlying software implementation. We discuss the general preconditions for and the process of leveraging firmware modification attacks against modern embedded devices.

**HP LaserJet firmware modification case study:** We use a firmware modification vulnerability recently discovered by the authors in nearly all HP LaserJet printers [2] to present a real-world case study of the development cycle of such attacks against common embedded devices. We present the threat model characterization, vulnerability analysis and threat assessment of HP-RFU and show a full exploit against the

## I. INTRODUCTION

通读论文，崔安大帅哥将 payload 解释的很清楚，直接将 firmware update 的 PJI 命令发送给 raw printing port 就可以。也可以将 payload 也就是包含木马后门的固件 binary package 伪装在看起来人畜无害的学术论文或者简历里面。只要“简历”被打印一次，攻击者就成功了。看来连前文提到的连接远程主机执行更新都不需要了。当然，笔者也不是专门做这个的，上文只是个人的猜测。



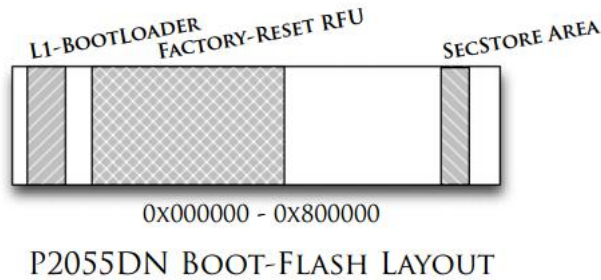


Fig. 5. Boot image layout on the SPI flash chip. The level-1 boot loader contains code that validates, unpacks and decompresses the factory reset RFU allowing us to reverse engineer the binary RFU format and compression algorithm.

way, once the PJI command is sent to the victim printer, it will recognize the print job as containing a valid firmware update and allow the attacker to make arbitrary modifications to the victim's firmware storage area.

The unpacked RFU package for the P2055DN contains over a dozen files. The main file of interest is the binary OS image, a single 14 MB ELF image containing the VxWorks operating system and various other vendor-specific additions.

The creation of the proof of concept malware essentially reduced to creating a VxWorks rootkit capable of:

- Command and control via covert channel
- Print job snooping and exfiltration
- Autonomous and remote-controlled reconnaissance
- Multiple device type infection and propagation to the Windows operating system and other embedded devices
- Reverse IP tunnel to penetrate perimeter firewalls
- Self-destruction

A video discussing the technical mechanics of this rootkit and a demonstration of its capabilities is publicly available [23].

The VxWorks OS image found within the RFU binary contains a complete socket library [24] and direct access to the underlying network transceiver hardware. The creation of the proof of concept code was mainly an exercise in identifying and intercepting the proper pieces of binary within the VxWorks image.

No host-based security mechanism exists within the firmware image. Thus, the attacker is free to make arbitrary changes to the victim device. As long as the functionality and general performance of the device is not altered, detection of firmware modification is not possible without careful removal and inspection of the hardware inside the printer.

Several challenges arose during the construction of the proof of concept code. The VxWorks image extracted from the RFU package contained no symbol information. Locating the appropriate socketlib, print job processing and raw network I/O binary interfaces within the binary proved non-trivial.

We developed a set of IDA-Python scripts to perform standard control-flow analysis of the target binary around code that we manually identified as network-facing. This effort was expedited by a patch made to the VxWorks kernel,

which redirected debug messages destined for the UART to a TCP connection. Using these two mechanisms, a dynamic analysis environment was created to probe network-facing code, which eventually yielded a small set of functions likely to be libraries used by multiple pieces of unrelated code. Function prototype data was taken from available VxWorks documentation and used as a final check to positively identify each library function.

Typically, the malware would be optimized, compressed, packed and broken up to fit within gaps inside the original firmware or placed within dynamically allocated memory. However, since the attacker controls the firmware storage area absolutely, we added a new section within the ELF header marked with *rwX* privileges. This gave us more than sufficient space to implement all the previously mentioned malware functionality. In total, 2,800 lines of assembly were written to create the proof of concept malware.

#### IV. THREAT MODEL AND ASSESSMENT

We present the threat model and assessment analysis for the HP-RFU vulnerability presented in Section III.

##### A. Threat Model Characterization

The HP-RFU vulnerability exploits a design flaw in the firmware update mechanism found in nearly all LaserJet printers. In order to achieve persistent firmware modification on the victim device, the attacker must deliver a malicious PJI command to the raw-printing processing subsystem of the target. This can be done by using the following attack types:

**Active Attacks** require the attacker to directly trigger the firmware update process by actively connecting to the printer and sending it the malicious PJI command over the printer's raw-printing port.

**Reflexive Attacks** are akin to reflexive cross-site scripting attacks where malicious firmware update commands are embedded in passive data that is passed along to the user of the victim device. For example, the final binary package of the HP-RFU attack can be embedded inside innocuous-looking documents and sent to unwitting users, perhaps in the form of an academic paper or resume. In this reflexive attack scenario, the actual attack is launched when the malicious document is printed.

##### B. Threat Assessment

Figure 6 illustrates an advanced persistent attack scenario where a compromised printer is used as a reconnaissance tool and offensive asset. Once the malware package is delivered to the victim printer, it can be used to carry out firmware modification attacks against other embedded devices like other printers, IP phones and video conferencing units. Compromised embedded devices can be used to establish reverse IP tunnels back out to the Internet, giving the attacker direct access to the secured internal network. These devices can also be used to carry out standard network attacks like ARP cache poisoning and act as offensive assets to further compromise



然后文章还详细解释了作者自创的 ABSR 和 Symbiote 的原理,ABSR 类似于 360 安全卫士,可以关闭开启不常用的功能,替厂商/用户做出“正确”的决定,甚至它还有自己的 ABSR configuration interface 配置界面,同时宣城有了 ABSR 的存在,可以更好的提升性能,反之像 ASLR 或者 DynamicRIO 等需要更改二进制或者 hot patching 的机制都是弱鸡。

Symbiote 则是 360 内核加固,在内核中插入钩子,防止自身被卸载或者禁用。就是这么简单和潇洒,玩的都是 360 玩剩下的东西,不过这套系统是给 Iot 设备使用的,不是 Windows。话说连 Windows 我们都防下来了,Iot 设备不知道简单多少倍了好么?这篇 paper 的地址在这里:

<http://ids.cs.columbia.edu/sites/default/files/ndss-2013.pdf>,有兴趣的读者可以继续深入阅读。不过应该也没有什么有用的内容了。

was not trivial and at times impossible, as was the case with the LaserJet P2055DN. We propose a technique, which we call Autotomic<sup>7</sup> Binary Structure Randomization (ABSR), which not only disables unnecessary features, but also removes the unused binary from the firmware image. This technique simultaneously reduces the attack surface of the embedded device as well as the amount of code and data that can be used as part of any shellcode.

Disabling unused features on the embedded device is helpful, but does not guard against exploitation via attack vectors within necessary features that cannot be removed. For example, vulnerable third-party libraries like ones identified in Section VI may be pivotal to the functionality of the embedded device. We believe techniques like ABSR should be used in conjunction with other host-based defenses to detect and mitigate the consequences of successful exploitation. Software Symbiotes have been demonstrated as a viable dynamic firmware integrity attestation technique on embedded systems such as enterprise routers.

Despite proper software and security engineering practices by vendors, firmwares will continue to be released with bugs and vulnerabilities. ABSR and Symbiotes are aimed at securing devices that run such firmware.

#### A. Autotomic Binary Structure Randomization (ABSR)

ABSR is a fortification technique currently being developed by the authors. This approach accepts arbitrary executables or firmware images as input and outputs a hardened, functionally equivalent variant of the original. The exploitability of the input binary is reduced by two primary operations: autotomic binary reduction and binary structure randomization. First, unused code, as determined by the particular configuration state of the target device, is autotomically removed in order to reduce the potential vulnerable attack surface of the overall system. For example, if a network printer is not configured to support LDAP authentication and UPnP, code sections corresponding to these feature sets are programmatically stripped from the resulting binary.

Furthermore, the autotomic operation can remove the binaries of features that are enabled by default but can not be disabled via configuration, which was precisely the case of the HP-RFU vulnerability. The RFU firmware update feature is rarely used but is enabled by default on all systems, some of which had no method of administratively disabling this code path; ABSR would remove the binary executables associated with the feature. Using the free space generated by the autotomic reduction phase, the binary structure randomization phase restructures the remaining executable binary blobs. We propose disabling and removing all unused features in general. However, the firmware update feature is a special case in which the code path should be disabled but not removed from the binary since this feature is necessary for future firmware updates. In this case, we propose an alternative

method of enabling this code path, potentially through an ABSR configuration interface.

This approach differs from most existing techniques in that no attempt is made to remap coherent blocks of code into randomized locations in memory. Instead, ABSR decomposes all remaining basic blocks of the binary in order to transform them into a randomized, functionally equivalent program while intentionally breaking control-flow isomorphism.

Like software Guards and Symbiotes, ABSR is not a stop-the-world defense mechanism. ABSR does not halt the original functionality of the protected device while it is engaged. Like other randomization techniques such as Address Space Layout Randomization (ASLR) and Instruction-Set Randomization (ISR), ABSR is built into the architectural design of the protected device and does not require dynamic patching or binary rewriting like DynamoRIO. ABSR is an topic of ongoing research.

#### B. Software Symbiotes

Software Symbiotes [48] are a host-based defense mechanism that are specifically designed to inject intrusion detection functionality into the binary firmware of existing embedded devices. A Symbiote is a code structure embedded in situ into the firmware of an embedded system. The Symbiote tightly co-exists with its host executable in a mutually defensive arrangement, sharing computational resources with its host while simultaneously protecting the host against exploitation and unauthorized modification. The Symbiote is stealthily embedded in a randomized fashion within an arbitrary body of firmware to protect itself from removal and unauthorized deactivation. Unlike remote software attestation techniques, Guards and Symbiotes do not require the disabling of interrupts or a full system halt while the security mechanisms are engaged.

### IX. CONCLUSION

We presented a general discussion of firmware modification attacks against embedded systems as well as a specific case study of such a vulnerability found in nearly all HP LaserJet printers. We discussed the analysis process that led to the discovery of the HP-RFU vulnerability as well as the implementation of a proof of concept printer malware. The printer malware presented in this paper can be delivered through standard PDL commands and can be embedded in innocuous document formats such as PostScript. It is capable of stealthy network reconnaissance, data exfiltration and propagation by autonomously compromising general purpose computers and other embedded device types. The HP-RFU vulnerability exploits a fundamental design flaw found not only in nearly all LaserJet printers, but in other modern embedded systems as well. Thus, the process presented in this paper can be generalized and applied to the exploitation of similarly vulnerable embedded systems.

We presented the results of exhaustive scans of IPv4 to track the size and distribution of all publicly accessible vulnerable LaserJet printers over time. Out of over 90,000 vulnerable units, only 1.08% of the vulnerable population has been

<sup>7</sup>Autotomy - The spontaneous casting off of parts is a (biologically) viable security mechanism.



继续挖崔安的家底，还挖到了催帅哥作为嘉宾在 BlackHat2015 发表过演讲，将的是如何将打印机和电话机的 serial port 串口上 dump 下来的数据，通过调制器 Funtenna 发射出去，然后坐在家里用“电视机”来接受这些数据。并且祈祷者 FBI 不要跟自己在同一个频道喔！

安全 | <https://www.blackhat.com/us-15/briefings.html#emanate-like-a-boss-generalized-covert-data-exfiltration-with-funtenna> ☆

## EMANATE LIKE A BOSS: GENERALIZED COVERT DATA EXFILTRATION WITH FUNTENNA


PRESENTED BY  
Ang Cui

Funtenna is a software-only technique which causes intentional compromising emanation in a wide spectrum of modern computing hardware for the purpose of covert, reliable data exfiltration through secured and air-gapped networks. We present a generalized Funtenna technique that reliably encodes and emanates arbitrary data across wide portions of the electromagnetic spectrum, ranging from the sub-acoustic to RF and beyond.

The Funtenna technique is hardware agnostic, can operate within nearly all modern computer systems and embedded devices, and is specifically intended to operate within hardware not designed to act as RF transmitters.

We believe that Funtenna is an advancement of current state-of-the-art covert wireless exfiltration technologies. Specifically, Funtenna offers comparable exfiltration capabilities to RF-based retro-reflectors, but can be realized without the need for physical implantation and illumination.

We first present a brief survey of the history of compromising emanation research, followed by a discussion of the theoretical mechanisms of Funtenna and intentionally induced compromising emanation in general. Lastly, we demonstrate implementations of Funtenna as small software implants within several ubiquitous embedded devices, such as VoIP phones and printers, and in common computer peripherals, such as hard disks, console ports, network interface cards and more.



然后把链接的长达 206 页的 PPT 也看了一遍，作者仍然希望有实习生将 Funtenna 带到机密实验室去，将打印机和电话机的 serial port 串口上 dump



下来的数据，通过调制器 Funtenna 发射出去，然后坐在家里用“电视机”来接受这些数据。并且祈祷者 FBI 不要跟自己在同一个频道喔！当然，为了尽可能提高适用性，它还考虑了将载体从无线电换成电磁波的可能性，在加密解密上也做了文章，怎么讲呢？这是一个好思路，笔者差点路转粉了，因为如果不是针对“军工设备”或者“绝密实验室”的话，通过无线电将数据泄漏出去确实是一个非常好的方案，比如在商业谍战渗透的过程中，这一招可以大杀特杀，商业环境下大多数人还是安全小白的。如果将这样的一个设备插到物理隔离的服务器上，可以拿下好多血，当然前提是防范不力，当然没有谁会对服务器防范不力的。PPT 的链接在这里：

<https://www.blackhat.com/docs/us-15/materials/us-15-Cui-Emanate-Like-A-Boss-Generalized-Covert-Data-Exfiltration-With-Funtenna.pdf>

话题讲到这里，是不是就结束了呢？不是的，其实话才说到一半。前文讲到他们这样的一个团队，每年可能需要至少百万美金，才能活下来，哪怕在中国，也需要至少两三百万，才能活下来，这里没有提他们有多少利润，只是完全在说需要这么多真金白银，才能不至于倒闭。

也就是说，他们得有收入，这种情况下，就有两种可能。

- 他们有收入，能赚钱；
- 他们没有收入，赚不到钱。

我们先讨论第一种可能，因为这一种可能其实是一种可能性比较渺然的可能。首先，卖产品，卖授权，给自己 Iot 设备安装由几个大学生开发出来的“安全软件”？一般的设备厂商肯定是信不过外人的，一般选择自建团队或者哪怕外包给

值得信赖的国内重点实验室，都可以。毕竟原理就在这里，剩下的就是写代码的工作，如何竞争过专业的“码畜”，这并不是他们最擅长的地方。

其次，卖服务，卖方案，也就是“乙方”。在国内，“乙方”任何时候都不是一份好干的差事，更不是一份“体面”的差事，正所谓客户虐我千百遍，我待客户如初恋，“钱多事少离家近，位高权重责任轻。睡觉睡到自然醒，数钱数到手抽筋”这里的每一个词的反义词，就是“乙方”的工作特性。一般都是赚着卖白菜的钱，操着卖白粉的心，鉴于安全行业的尿性，不出事谁也没有功劳，一出事全是你的不对，乙方也是人，驻场也是人，也是有爹有妈有人疼的宝宝，有时候为了拿到明年的合同，为了可能是六位数甚至是五位数的少得可怜的预算，腿都跑断了，点头哈腰、毕恭毕敬，鞍前马后、肝肠寸断的，对于做市场的售前业务，再多溢美之辞都不嫌多。

然而，还是赚不到钱。不管你是做方案也好，卖服务也好，还是卖软件卖售后也好，安全就是个人人都不待见的行业。不仅是 2B (toB) 赚不到钱，2C (toC) 也是赚不到钱的，360 把免费杀毒的概念普及开来之后，谁还买杀毒软件，江民瑞星金山卡巴，不免费、都玩完。

崔安也好，Red balloon security 也好，他虽然用的是别人的 0day，但是把这一套玩转了，也是需要不少的水平的，而且还很努力地出 paper 推介，上 Blackhat 做宣传，还参加这样的节目拍摄来炫技做宣传，提高团队知名度，这些都是非常值得肯定的，就冲着他积极向上的态度，也可以给他数个大拇指。

但是他并不是 Top 选手，屡战屡败的 Geohot 才是真正的最好的例子，Geohot 真的很厉害，破解 iPhone 第一人，PS3 第一人，ChromeOS 等等都不在话下，这些可都是自己的洞，自己原创的漏洞！甚至研发了自己的自动驾驶系统 Comma，这才是真正的天才，然而，天才赚到钱了么？

是他没有技术么？不是！要是说他没有技术，全世界 99.99% 的程序员都没有技术。是他不够勤奋么？不是！马不停蹄地破解这么多设备，一会儿想要加入 Facebook，一会儿想要加入 Google，一会儿研发自己的自动驾驶设备，让那些巨头脸上蒙羞！他以一己之力，提升着整个社区团队的技术和道德水准，要说这样的人不努力，那全世界 99.99% 的程序员都是不努力的了。像崔安同学，也是非常努力的！他也以自己力所能及的方式，向着目标前进中！

然而，天才 Geohot，还是没能赚到钱。崔安团队我不知道，难说。

回到国内，国内某个知名团队主力多次诉苦，在找到“打比赛”这条正确的道路之前，曾经拮据得住地下室，几个月发不出工资，连吃顿烧烤都犹豫不决，为了拿个五千块的外包跑吃了多少次闭门羹，这才是安全团队创业的主流“姿势”。后来有名的也好，无名的也好，几乎都是挖洞打比赛。有名的实力雄厚的打国际比赛，奖金最多也就几万美金，都不够开一个人一年的工资，说不定挖到这个洞就花了不止一年。无名的打国内的比赛，刷漏洞平台，一个洞就几百块钱，没事，大家一起熬夜刷吧！积少成多，总之也能活下来就行了。几乎找不到比安全团队创业更加卑微的创业方式了。放在日本，就跟“浪人”是一样的，流浪的武士，空有一身抱负，却得不到赏识。国内的安全团队，创业都不赚钱。



现在开始讨论第二种可能，团队没有收入，赚不到钱，怎么办？

为什么不赚钱？因为商业逻辑和技术逻辑，从来就是两码事。

技术逻辑是数学逻辑，有对有错，对就是对，错就是错。对就继续前进，进入下个分支；错就立即止步，重新进行计算。我们讲究的是严格的论证和推导，是一板一眼的科学精神。

商业逻辑是经济学逻辑，研究的对象是资源的优化与配置。在资源的优化与配置的过程中，没有对与错，只有赢与输。资源总会倾向于本身就拥有巨大资源的人，这是资源的原罪，也是人性的原罪。资源不讲科学，谁更有资源，资源就倾向于谁，资源没有对错。在人类历史上贫富差距从来没有消失过。

很多科学家很喜欢违背资源的本性，故作清高，不愿与资源同流合污，比如说 Geohot，非要跟索尼打官司，自己出 Comma 要把所有科技公司打趴下，这就是非常典型的科学家思维。崔安大帅哥也并没有靠近资本，从谷歌到的结果来看，没有投资，发表的 presentation 也是黑客圈子里的同僚，至少从表面来看，是孤傲的，同时又希望遇到伯乐。千里马常有，伯乐不常有啊。有伯乐前来光顾的时候，还要对伯乐进行筛选，不能只投钱，还要是理解的，有学识的，人脉广的，无条件的等等，伯乐也不敢来了。

资源的优化与配置，在商业的环境下，这里的“资源”指的就是真金白银；如果把资源的定义再放宽一些，就是名与利，名利场，要么名，要么利，都是聚居属性的人类社会中所有男性朝思暮想的无上至宝，李建成甚至为了争夺皇位杀

死了自己的亲哥哥，在资源的配置过程中发挥主观能动性，拿得到了封建社会最大的资源——皇位。

而上文的这些安全团队，在资源的优化和配置过程中，得到了哪些或者什么资源呢？不知道，看不出来。

- ◆ 在中国，政府是最大的雇主，也是拥有巨大资源的行业领导，在世界范围里，政府也是最大的合约提供方，各行各业最大的买方，在经济学的研究对象中，通常是要将政府采购作为单一研究范畴，进行专项研究和深入；
- ◆ 其次是业界巨擘，各行各业都有托拉斯、辛迪加、康采恩这些经济学范畴上独裁、专制的垄断巨头，这些巨头通过各种形式的垄断，攫取了各行各业的绝大多数利润，为这些大金主服务，不管拿下哪一家，一辈子吃喝不用愁了；
- ◆ 再不济，中型企业、有良好发展的中型企业、拥有核心资源的企业，也是良好的目标，但最多算三等公民，风险高，回报低，还不稳定；
- ◆ 如果想着从小微企业或者个人手中得到信赖、得到资源，得到付费用户，在外国或许行得通，在中国还是算了吧。

选择第一梯队的公司很多，而且这些公司大多都上市了，牢牢垄断着互联网信息安全第一梯队阵营，主板上市的卫士通、蓝盾股份、泰豪科技、浪潮信息、东方电子、中国软件、启明星辰，创业板上市的绿盟科技，未上市的华为、中兴、大唐、华三，这些企业生产的软硬件垄断了军工市场、大中小型政府及企业市场，牢牢地占据着第一梯队。任何国家级战略也好，地方政府规划也好，只要是信息

技术安全相关的投资和资源，最大的一口肉、最大一块蛋糕，永远是在这些企业的牢牢掌握之中。

选择第二梯队的公司也不少，KEEN 团队最终选择了被腾讯收购，盘古也依附于 360，长亭科技依附于真格基金、启明星辰，靠打比赛打出名气，建立了绝对的技术壁垒，成为了资本眼中的香饽饽，也树立了成名要趁早、还得靠比赛的准入规则。不打比赛，别人都不认识你，打了比赛，360 也好、腾讯也好，每年都有自己的 CTF 比赛，想要加入他们很简单，用实力赢得他们的尊重即可。跟着这些公司，虽然荣华富贵也很难，毕竟竞争实在是太激烈了；但是温饱肯定是没有问题的，至少不用为生存而担忧，可以把心思专注在科学研究上，虽然只是喝汤，但也能补补身子，养精蓄锐，无丝竹之乱耳，无案牍之劳形。

第三梯队的就比较难办了，时时刻刻都处在崩溃的边缘，必须不断地自我革新、寻找新的利润增长点，才能保证不被淘汰，钱少事多离家远，位低权轻责任重，就是对这个梯队最好的描述。第四梯队的就是地狱模式了，“小而美”的杀毒软件和方案，在中国是没有市场的，在中国只有百团大战。

古代讲究“士农工商”的排名，如今是“士商工农”；第一梯队牢牢掌握着“士”的位置，古代也好现代也好，都是第一选择；第二梯队把“商工”结合在一起，既在商业上形成垄断，又在技术上形成垄断，老虎不在家，猴子当带王，在民主国家，托拉斯的实力早就超过政府，甚至形成跨政府联盟，比如欧佩克石油输出国组织，该组织的力量，远远超过单一国家政府的力量。第三第四梯队就是新时代的“小农经济”，最辛苦，最劳累，靠天吃饭，还没钱赚。



崔安就是要搞小农经济，还有 Geohot，他们有他们的价值观，他们的价值里面，“钱”、business 不是最重要的吧，可能黑客精神，自由的精神和灵魂，才是最重要的吧。众人皆醉我独醒、举世皆浊我独清，这样的境界也不是常人可以达到的，常人的境界，还停留在像笔者这样分析他们赚不赚钱，然后来对他们评头论足，这就是常人，这就是为了五斗米折腰的常人！

而他们，真的是为了黑客精神，在奋斗！