DNSMon

DNSMon: using DNS data to produce threat intelligence (3)

suqitian, Alex.Turing

Feb 9, 2021 • 7 min read

Background

This article is the third in our series of articles introducing DNSMon in the production of threat intelligence (Domain Name IoC).

As a basic core protocol of the Internet, DNS protocol is one of the cornerstones for the normal operation of the Internet.

DNSMon, which was born and raised on the foundation of DNS protocol, naturally has a broad vision and can be involved in security incidents that happen in different industries or different platforms. In our first two blog posts of DNSMon's series, the Skidmap mentioned in the first article was infected with cloud hosts on Linux platform; while the second article mentioned a set of domain names issued after the Windows platform of an Internet cafe was infected; this article is a case involving Android platform. We believe this is the beauty of using DNS data, no matter what platform the bad guys are using, the knowledge points or rules used by DNSMon do not fundamentally change.

Blocking the unknown threat

Recently, we noticed that DMSMon has marked a set of structurally similar domains and automatically blocked them starting from 2021-01-10.

```
BLOCK:
        utionstro.top (from 2021-01-10 to [redacted])
--
BLOCK:
        lesseased.top (from 2021-01-12 to [redacted])
--
BLOCK:
        ssuminat.top (from 2021-01-14 to [redacted])
--
BLOCK:
        holidano.top (from 2021-01-16 to [redacted])
--
BLOCK:
        thinkdisen.top (from 2021-01-17 to [redacted])
```

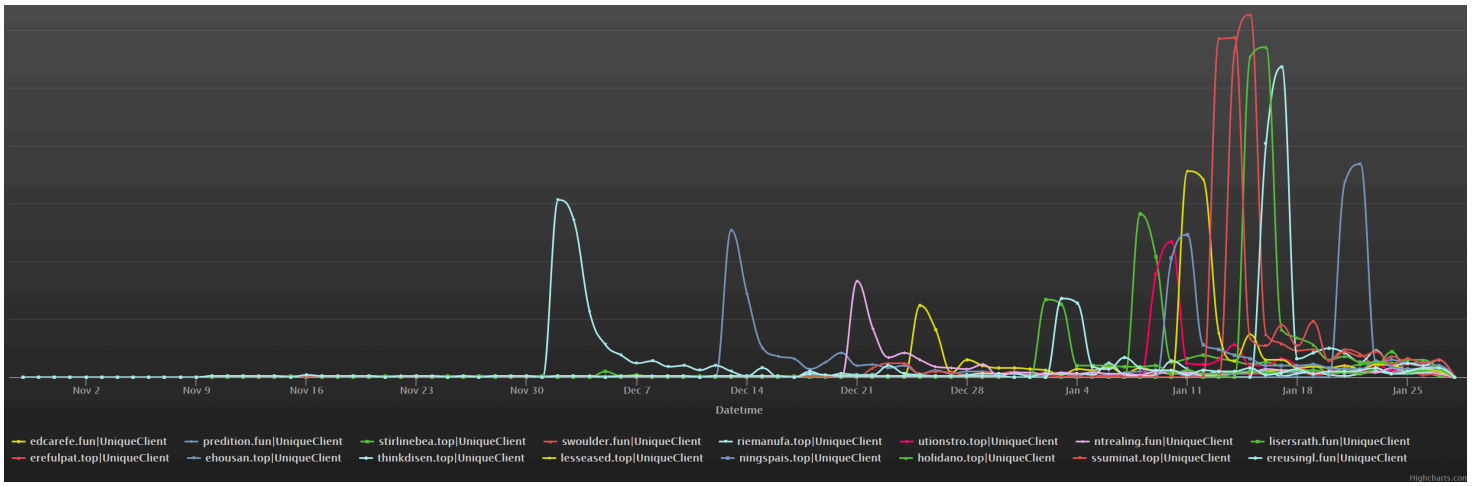
Going into the system to check the collected information, we see that this is a set of malicious domains related to Android platform.

Expanding Similar Domains

Using the rdata and other data of the automatically blocked domains, we found that DNSMon actually warned of more than the 5 domains listed above, there are 16 domains in this category in total.

```
ereusingl[.]fun
lisersrath[.]fun
predition[.]fun
ningspais[.]top
ssuminat[.]top
erefulpat[.]top
stirlinebea[.]top
utionstro[.]top
lesseased[.]top
thinkdisen[.]top
edcarefe[.]fun
holidano[.]top
ehousan[.]top
swoulder[.]fun
riemanufa[.]top
ntrealing[.]fun
```

In terms of active time, these 16 domains are put into use sequentially and have a relatively intensive launch period in January 2021.



In the course of writing this article, we have seen other new similar domains captured by our system such as antdaugh[.] top, ngslalatfin[.] top, etc., but for the sake of consistency, these new domains will be omitted from this article.

Reasons for autoblocking

In the first article, we mentioned that

> *The core of DNSMon is to cross-compare the massive DNS data with the security-related data owned by 360 (including whois, web, sandbox, honeypot, certificate, etc.) and analyze it to derive the threat intelligence IOC.*

In the following, we analyze the data features and annotate from the whois and sandbox obtained by the system. Of course, in addition to these data, DNSMon's autoblocking process also relies on other feature data, but we will not list them here.

1. whois

The 16 domain names were registered in two batches, one on 2020-11-06 and the other on 2020-12-23; the length of validity of the registration: one year, and the

privacy protection was turned on.

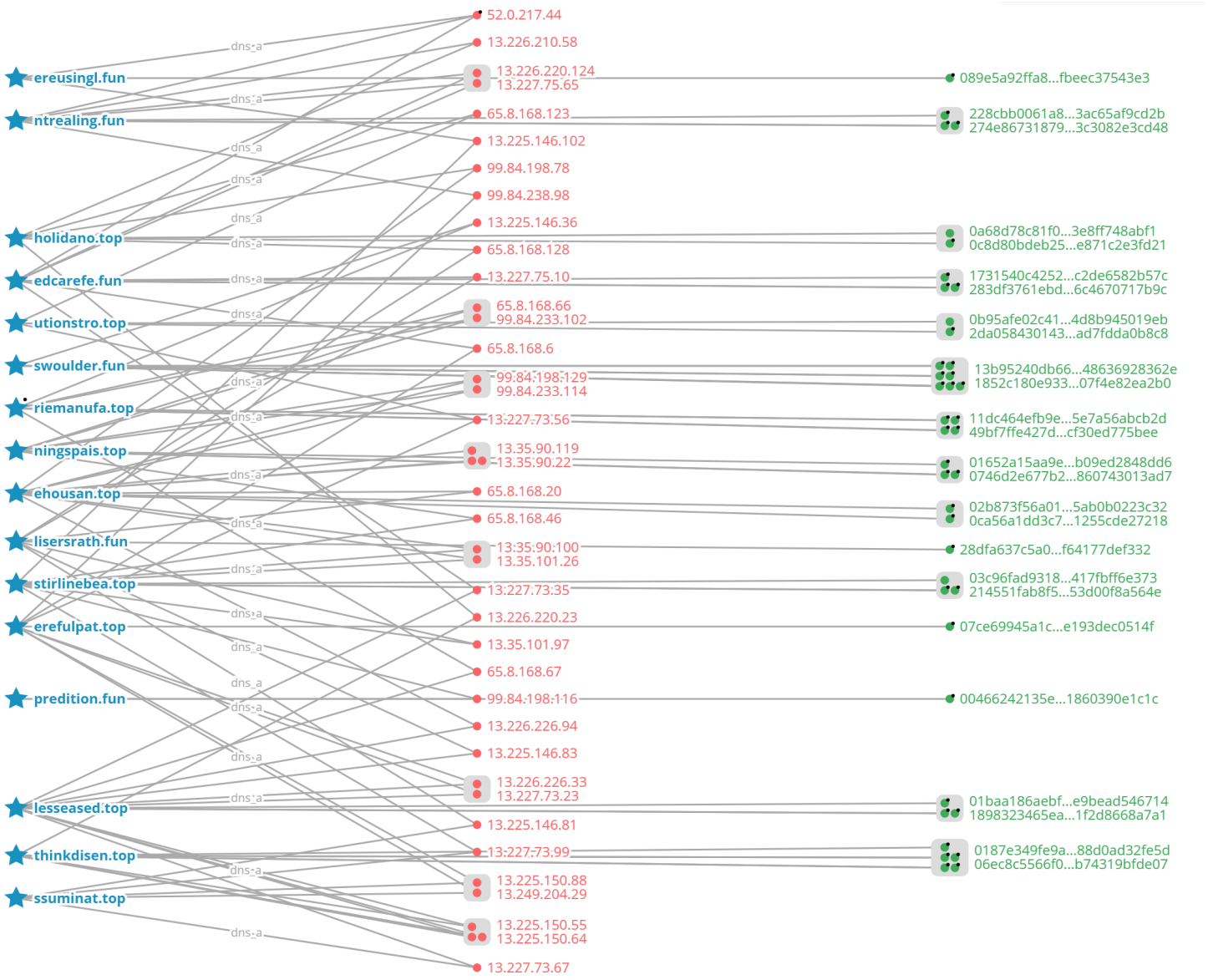
```
ereusingl.fun   createddate                2020-11-06 10:10:12
ereusingl.fun   updateddate                2020-11-06 10:20:27
ereusingl.fun   expiresdate                2021-11-06 23:59:59
ereusingl.fun   status                     addPeriod | clientTransferProhibited | se
lisersrath.fun  createddate                2020-11-06 10:10:12
lisersrath.fun  updateddate                2020-11-06 10:20:20
lisersrath.fun  expiresdate                2021-11-06 23:59:59
lisersrath.fun  status                     addPeriod | clientTransferProhibited | se
predition.fun   createddate                2020-11-06 10:09:51
predition.fun   updateddate                2020-11-06 10:15:26
predition.fun   expiresdate                2021-11-06 23:59:59
predition.fun   status                     addPeriod | clientTransferProhibited | se
...

ningspais.top   createddate                2020-12-23 15:57:59
ningspais.top   updateddate                2021-01-06 10:37:20
ningspais.top   expiresdate                2021-12-23 15:57:59
ningspais.top   status                     clientTransferProhibited
ssuminat.top    createddate                2020-12-23 15:58:00
ssuminat.top    updateddate                2021-01-06 10:37:23
ssuminat.top    expiresdate                2021-12-23 15:58:00
ssuminat.top    status                     clientTransferProhibited
erefulpat.top   createddate                2020-12-23 15:58:01
erefulpat.top   updateddate                2021-01-06 10:37:24
erefulpat.top   expiresdate                2021-12-23 15:58:01
erefulpat.top   status                     clientTransferProhibited
...
```

Batch registration + new domain names + short registration validity period + privacy protection, malicious possibility +1.

2. Sandbox

DNSMon's data on the sandbox focuses on the network behavior of the sample and extracts the relationship between the sample and the domain name visited, and then evaluates the score of the domain name in turn with the help of the sample's score. As can be seen from the correlation chart, the samples visiting the domains have malicious labels in the system (the samples are marked with "black dots" in the upper right corner).



According to the label propagation algorithm, 16 domains were awarded with black possibility plus points.

In order to clarify the purpose of sample propagation, we conducted a simple reverse analysis of the sample.

Reverse Analysis

This article selects a sample named "***Your File Is Ready To Download.apk***" as the object of analysis, and its basic information is shown below.

MD5：230ca35f90c55bf9c46ddfb7980b632d
File type：Android

Magic : Zip archive data, at least v2.0 to extract
File size : 543671 bytes

The sample ***Assets*** have a dat file that contains Base64 encoded configuration information.

And the content is

```
eyJjaWQiOiIxZGZiZTAzNS1kNzViLTQzYzgtYmMwMy1kYTg5YWU5NzU1ZjYiLCJ1cmxzIjpbImh0dHBzOi8vb
```

After Base64 decoding, we get the following configuration information in json format, and we can see that the "urls" field contains the domain name we want to characterize.

```
{
    "cid": "1dfbe035-d75b-43c8-bc03-da89ae9755f6",
    "urls": [
        "https[:]//lisersrath.fun"
    ],
    "fn": "Your File Is Ready To Download",
    "info": "blank_",
    "routes": {
        "x86": "/x86",
        "arm64-v8a": "/arm64",
        "x86_64": "/x64",
        "armeabi-v7a": "/arm"
    },
    "uid": "888098709355331972",
    "sid1": "888098709355331972",
    "tid": "792297",
    "sid2": ""
}
```

Download data from the domain stored in "urls" via the following code snippet.

This process can generate the following URL.

```
lisersrath[.]fun/x86?v=0.0&l=6.9&p=Y29tLmludGVuc2l2ZS5zb3VuZA==
lisersrath[.]fun/x64?v=0.0&l=6.9&p=Y29tLmludGVuc2l2ZS5zb3VuZA==
lisersrath[.]fun/arm?v=0.0&l=6.9&p=Y29tLmludGVuc2l2ZS5zb3VuZA==
```

So what exactly is the download? Taking the x86 parameters as an example, the download gets a json file (f7e8f0aec32ceb27d5e202d4b2b50812), which has an apk field pointing to a new APK file (12098a59b35bcabb16bfeab887eb7f9f).

After analysis, the new APK file belongs to the ***FakeAdsBlocker*** family, a covert adware program, which has a low detection rate of 4/64 in VT at the time of this writing.

In addition, we also found another URL.

```
lisersrath[.]fun:80/?cid=c423cdff-6c1b-4052-859f-11223c65f1ad&tid=827722&sid1=318254
```

Accessing the above domain, for example, will download an APK named "synapse_x_key" (a384b97afd0f9432a71b27fa0ccd9667), which is homologous to the sample analyzed above.

The main function is to download and execute FakeAdsBlocker.

So far, the purpose of the dissemination of this batch of domain names is very clear, they are used to spread FakeAdsBlocker Downloader and FakeAdsBlocker.

New features

In the process of manually checking the URLs, we noticed a patternthat we had not seen before, that is, a string of identical URLs with randomly downloaded samples at different times, with different MD5 values for each sample file.

For example.

```
https[:]//lisersrath.fun/?cid=f0e87e37-1bc6-4073-af71-efcc4a3eca22&tid=792297&sid1=72
```

Randomly downloaded 3 times, the file name is "Your File Is Ready To Download.apk", but the MD5 values are.

```
ab2f4fde57fdcb56bce5ffa48c4d9069
21c99e0a12a7ce9dc0d91df9e29af4ad
1e55fde5540147fbfac1c8060c449c58
```

Again, the author randomly picked another domain to test, same pattern

```
https[:]//holidano.top/?cid=a53fd199-f207-49dc-b5d7-aea0de14eee3&tid=899546&sid1=4009
```

The file name "MobileVPN.apk", MD5 values are different.

```
a3ad2ba6caf05275545d65cb2e8990d4
f4316d300116fa9de72f6f0ed3d29c28
56396eae0bbafd82701399d717ff3708
```

Inspired by this, we extracted a new feature and searched the database with the new feature to find the data with the same feature in the history, such as the URL in the following figure.

Subsequently, all eligible historical data are manually statistically analyzed to determine the size of the contribution value of the new features to the judgment of the black action, and finally decide whether to merge them into the DNSMon system. It can be said that the accumulation of features is extremely important for

a massive data analysis and mining system like DNSMon. It is the system that has precipitated a lot of rules in more than 6 years since 2014 that makes it possible to extract threat intelligence (domain name IOC) from the daily hundreds of billions of DNS traffic and provide security defense to end users.

Conclusion

Thanks to the fundamental nature of DNS, DNSMon has a broad view and can see security events occurring on Linux, Windows and Android platforms without discrimination. The main reason for this capability is that the system has accumulated a large number of security-related feature rules.

Readers should note that they should not easily imitate the process of judging the malicous in the article by themselves. Industrial IoC production includes not only the basic determination process, but also the complex processes of filtering, false alarm prevention, feedback correction, fail-safe, and automation. This article is written to facilitate the reader's understanding of the general principles of how DNSMon works, but the complexity of the various devilish details of the industrialized process is far beyond what can be described in the lines.

IoC

```
ereusingl[.]fun
lisersrath[.]fun
predition[.]fun
ningspais[.]top
ssuminat[.]top
erefulpat[.]top
stirlinebea[.]top
utionstro[.]top
lesseased[.]top
thinkdisen[.]top
edcarefe[.]fun
holidano[.]top
ehousan[.]top
swoulder[.]fun
riemanufa[.]top
ntrealing[.]fun
```

— 360 Netlab Blog - Network Security Research Lab at 360 —

DNSMon

俄乌危机中的数字证书：吊销、影响、缓解

商业数字证书签发和使用情况简介(删减版)

An assessment of Non-Authorized Domain Name Resolution provided by DNS Resolution Service Provider

rinfo

rinfo卷土重来，正在疯狂扫描和挖矿

版权 版权声明：本文为Netlab原创，依据CC BY-SA 4.0 许可证进行授权，转载请附上出处链接及本声明。 概述 2018年我们公开过一个利用ngrok.io传播样本的扫描&挖矿型botnet家族: "利用ngrok传播样本挖矿"，从2020年10月中旬开始，我们的BotMon系统检测到这个家族的新变种再次活跃起来，并且持续至今。相比上一次，这次来势更加凶猛，截至202...

DNSMon

DNSMon: 用DNS数据进行威胁发现(3)

--- Linux，Windows，Android，一个都不能少 背景 本文是介绍DNSMon在生产威胁情报(域名IoC)系列文章的第三篇。 DNS协议作为互联网的一项基础核心协议，是互联网得以正常运行的基石之一。在祖国960万平方公里的土地上，那一张纵横交错的数据网络里，每一秒都有数万亿计的DNS数据包在高速穿梭着，它们或来自于机房的服务器，或...

See all 28 posts →

Feb 10, 2021 · 8 min read

Feb 8, 2021 · 10 min read