The Domain Generation Algorithms of SharkBot

bin.re/blog/the-dgas-of-sharkbot/

Table of Contents

Disclaimer

These are just unpolished notes. The content likely lacks clarity and structure; and the results might not be adequately verified and/or incomplete.

Changes

2022-07-14 17:37:50: fixed the regex for version 1.63

Malpedia

For more information about the malware in this blog post see the Malpedia entry on SharkBot.

Cover Image Photo by Jonas Allert on Unsplash

SharkBot is an Android banking malware that steals credentials and banking details [1234]. Apart from one or two hard-coded domains, it relies on a fallback domain generation algorithm (DGA) for communication — which is rather rare for Android malware. The DGA was changed several times during the further development of SharkBot. This blogpost shows four versions of this DGA, and how they differ.

Both jadx and jd-gui were able to decompile almost the entire SharkBot code to great precision, but failed when it came to the DGA function. I therefore used `dex2jar` to translate the compiled Android application code to a jar file, then ran the unmaintained `jad` decompiler on the class that contains the DGA. You can find reimplementations of all four DGAs in Python on my GitHub repo.

Version 0.0.0

The first sample that I found has the version set to `0.0.0`:

MD5
0356f17f28778da7c97dc8b661c0aeb0

SHA1
2c4828f926471ec4f3522fc28dd4d8fdec692c35

SHA256
76b4ee2da4e39677038ea033f25652fb02ed9e84ab829ce212fa1dfbc941df2c

Size
3 MB (4116240 Bytes)

Compile Timestamp
n/a

Links
VirusTotal

Package Name
`com.btfezxwhygk2dw0gaj.eguafyojiqcw7a`

Detections
Virustotal: 21/73 as of 2022-04-25 06:01:16 - Android.BankBot.904.origin (DrWeb), Android.Sharkbot.GEN46543 (CAT-QuickHeal), Android.PUA.DebugKey (Trustlook), Trojan (0058a5511) (K7GW), AndroidOS/SpyAgent.F.gen!Eldorado (Cyren), AppRisk:Generisk (SymantecMobileInsight), a variant of Android/Spy.Agent.BWR (ESET-NOD32), HEUR:Trojan-Banker.AndroidOS.Sharkbot.d (Kaspersky), a.privacy.BankSharkBot (Tencent), Trojan.Agent.Android.340641 (Zillya), Artemis!Trojan (McAfee-GW-Edition), Andr/Banker-HAQ (Sophos), Android:Evo-gen [Trj] (Avast-Mobile), ANDROID/SpyAgent.FKIX.Gen (Avira), HEUR:Trojan-Banker.AndroidOS.Sharkbot.d (ZoneAlarm), Android.Trojan.SharkBot.B

(BitDefenderFalx), Trojan/Android.Agent.1071402 (AhnLab-V3), Artemis!0356F17F2877 (McAfee), Trojan-Spy.AndroidOS.SharkBot (Ikarus), Android/Agent.BWR!tr (Fortinet)

After some mild deobfuscation of the strings and renaming the variables, the DGA looks as follows:

```
private String dga()
{
    StringBuilder urls = new StringBuilder();
    Calendar calendar = Calendar.getInstance();
    String tlds[] = ".top,.xyz,.cc,.info,.com,.ru,.info,.net".split(",");
    int nrTlds = tlds.length;
    int i = 0;
    while(i < nrTlds)
    {
        String tld = tlds[i];
        try
        {
            urls.append(",http://");
            StringBuilder seed = new StringBuilder();
            urls.append(
                Base64.encodeToString(
                    seed.append(
                        calendar.get(Calendar.WEEK_OF_YEAR)
                    )
                    .append("pojBI9LHGFdfgegjjsJ99hvVGHVOjhksdf")
                    .toString()
                    .getBytes()
                , Base64.NO_WRAP)
                .substring(0, 19)
            )
            .append(tld);
        }
        catch(Exception exception) { }
        i++;
    }
    return urls.toString().toLowerCase();
}
```

The DGA performs these steps to generate domains:

1. Take the current week number (e.g. 12)
2. Append the hardcoded string `pojBI9LHGFdfgegjjsJ99hvVGHVOjhksdf`
3. Base64 encode the result
4. Take the first 19 characters
5. Append a TLD from the list `.top`, `.xyz`, `.cc`, `.info`, `.com`, `.ru`, `.info`, `.net` in order.
6. Convert the domain to lower-case

This results in domains like these:

```
mnbvakjjouxir0zkzmd.xyz
mnbvakjjouxir0zkzmd.live
mnbvakjjouxir0zkzmd.com
mnbvakjjouxir0zkzmd.store
mnbvakjjouxir0zkzmd.info
mnbvakjjouxir0zkzmd.top
mnbvakjjouxir0zkzmd.net
```

The second level domain (SLD) is the same for all generated domains at a given time. And since the time-dependent portion only affects the first two characters (when the week number is ≤ 9), or the first three characters (when the week number is ≥ 10), all domains will follow either of the two following pattern:

```
[n-o][awgq]bvakjjouxir0zkzmd\.(top|xyz|cc|info|com|ru|info|net)
[nm][dtjz][acegikmquy]wb2pcstlmsedgzgz\.(top|xyz|cc|info|com|ru|info|net)
```

Version 1.63.3

The second version of the DGA appears in this sample:

MD5
48ad4e0478e4d742f51848604d06130e

SHA1
a9ca49ef2201707b7bcf57798fc67e69d238c900

SHA256
c14f413d8ed944ba7e4364e6b17585019fd622feeb4b53f7002a742d7389e08a

Size
7 MB (7345268 Bytes)

Compile Timestamp
1970-02-21 06:07:33 UTC

Links
VirusTotal

Filename
(VirusTotal)

Detections
Virustotal: 8/72 as of 2022-05-20 04:22:02 - Android.BankBot.958.origin (DrWeb), a variant of Android/Spy.Agent.BWR (ESET-NOD32), HEUR:Trojan-Banker.AndroidOS.Sharkbot.d (Kaspersky), ANDROID/Bankbot.FKWN.Gen (Avira), HEUR:Trojan-Banker.AndroidOS.Sharkbot.d (ZoneAlarm), Trojan/Android.Agent.1111318 (AhnLab-V3)

This sample features a small change to the DGA: The seed is now also based on the current year, which prevents the DGA from repeating after a year.

```
private String dga()
{
    StringBuilder urls = new StringBuilder();
    Calendar calendar = Calendar.getInstance();
    String tlds[] = ".top,.xyz,.cc,.info,.com,.ru,.info,.net".split(",");
    int nrTlds = tlds.length;
    int i = 0;
    while(i < nrTlds)
    {
        String tld = tlds[i];
        try
        {
            urls.append(",http://");
            StringBuilder seed = new StringBuilder();
            urls.append(
                Base64.encodeToString(
                    seed.append(
                        calendar.get(Calendar.WEEK_OF_YEAR) +
                        calendar.get(Calendar.YEAR)
                    )
                    .append("pojBI9LHGFdfgegjjsJ99hvVGHVOjhksdf")
                    .toString()
                    .getBytes()
                , Base64.NO_WRAP)
                .substring(0, 19)
            )
            .append(tld);
        }
        catch(Exception exception) { }
        i++;
    }
    return urls.toString().toLowerCase();
}
```

By adding the year, the time dependent component always exactly affects the first three characters of the domains, which therefore always follow this pattern:

```
mja[xyz0123][mno][hnx3]bvakjjouxir0z.\.(top|xyz|cc|info|com|ru|net)
```

Version 2.1

The next version of the DGA is used in this sample:

MD5
92011ba743860567b85f46aedf360661

SHA1
506df2fd2e638ab614eeb4cbc416bd46fdbf6a19

SHA256
70b244a03a0eacd00cc52ea8863af2c459eb8dc2e6bf5887e657b401e0477485

Size
4 MB (4252173 Bytes)

Links
VirusTotal

Detections
Virustotal: 19/73 as of 2022-05-18 07:59:42 - Trojan.AndroidOS.Sharkbot.C!c (Lionic), TrojanBanker:Android/Sharkbot.bc773aca (Alibaba), Spyware (00592d791) (K7GW), AppRisk:Generisk (SymantecMobileInsight), a variant of Android/Spy.Agent.BWR (ESET-NOD32), HEUR:Trojan-Banker.AndroidOS.Sharkbot.e (Kaspersky), Android.Trojan-spy.Agent.Hphh (Tencent), Android.BankBot.977.origin (DrWeb), Artemis!Trojan (McAfee-GW-Edition), Andr/ShrkBot-B (Sophos), Trojan.AndroidOS.Agent (Ikarus), ANDROID/Bankbot.FKWN.Gen (Avira), TrojanSpy:AndroidOS/Sharkbot.B!MTB (Microsoft), Android:Evo-gen [Trj] (Avast-Mobile), Android.Trojan.Banker.ZP (BitDefenderFalx), Artemis!92011BA74386 (McAfee), Android/Agent.BWR!tr.spy (Fortinet)

For version 2, SharkBot made some larger changes:

- Switch to md5 instead of base64. This assures that the second level domains (SLDs) will no longer have a substantial portion that never changes.
- Adding the TLD to the seed, which leads to different SLDs for different TLDs.
- Cutting the length of the SLD from 19 to 16 letters.
- Using differnt TLDs

```
private String dga() {
    StringBuilder urls = new StringBuilder();
    Calendar calendar = Calendar.getInstance();
    String tlds[] = ".xyz,.live,.com,.store,.info,.top,.net".split(",");
    int nrTlds = tlds.length;
    int i = 0;
    while(i < nrTlds)
    {
        String tld = tlds[i];
        try
        {
            urls.append(",http://");
            StringBuilder seed = new StringBuilder();
            urls.append(
                calc_md5(
                    seed.insert(0, tld)
                        .append(calendar.get(Calendar.WEEK_OF_YEAR))
                        .toString()
                )
                .append(Calendar.YEAR)
                .toString()
                .substring(0,16)
            )
            .append(tld);
        }
        catch(Exception exception) { }
        i++;
    }
    return urls.toString().toLowerCase();
```

Strangely, the year is concatenated to the MD5 string instead of also being hashed. Because the resulting string is cut to the first 16 characters, this means that the year once more has no influence on the generated domains.

Version 2.8

The most recent sample that I investigated is version 2.8:

MD5
2dfe83d4d7c0b5e0cfc0537efdbbbb01

SHA1
f1a820369f02a696e8bcaecee464eeaef0847c44

SHA256
dae193b7cac6d048dcc37916c92b0d2b11c56aa3f45e9995c16417e3b0587404

Size
7 MB (7735296 Bytes)

Compile Timestamp

1970-02-24 12:54:57 UTC

Links
VirusTotal

Filename
(VirusTotal)

Detections
Virustotal: 4/73 as of 2022-05-12 15:24:09 - a variant of Android/Spy.Agent.CDR (ESET-NOD32), HEUR:Trojan-Banker.AndroidOS.Sharkbot.e (Kaspersky), ANDROID/Bankbot.FKWN.Gen (Avira)

In version 2.8 the DGA was fixed. Now the year is part of the input to MD5, so domains will be different for distinct years.

```
private String dga() {
    StringBuilder urls = new StringBuilder();
    Calendar calendar = Calendar.getInstance();
    String tlds[] = ".xyz,.live,.com,.store,.info,.top,.net".split(",");
    int nrTlds = tlds.length;
    int i = 0;
    while(i < nrTlds)
    {
        String tld = tlds[i];
        try
        {
            urls.append(",http://");
            StringBuilder seed = new StringBuilder();
            urls.append(
                calc_md5(
                    seed.insert(0, tld)
                        .append(calendar.get(Calendar.WEEK_OF_YEAR))
                        .append("")
                        .append(calendar.get(Calendar.YEAR))
                        .toString()
                )
                .substring(0,16)
            )
            .append(tld);
        }
        catch(Exception exception) { }
        i++;
    }
    return urls.toString().toLowerCase();
```

1. Malpedia entry for SharkBot ↩

2. SharkBot: a new generation of Android Trojans is targeting banks in Europe ↩

3. Google is on guard: sharks shall not pass! ↩

4. SharkBot: a “new” generation Android banking Trojan being distributed on Google Play Store ↩