Bypassing Defender SmartScreen using DNS Sinkholing

In this short blog, I discuss a simple method to bypass Windows Defender SmartScreen by blocking `checkappexec.microsoft.com`, `nav.smartscreen.microsoft.com`, and `nav-edge.smartscreen.microsoft.com`.

Introduction

SmartScreen performs 2 functions:

1. Protecting against malicious sites.
2. Protecting against malicious downloads (app or app installer).

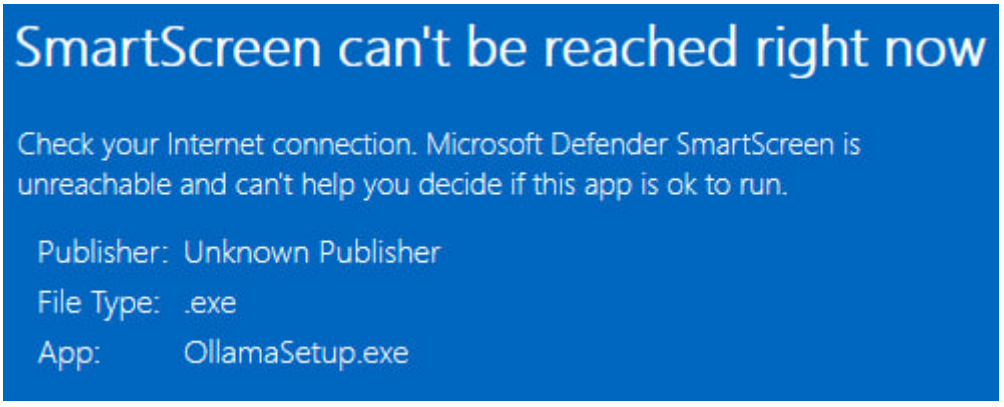
Telemetry Tales

I have a Pi-Hole container running on my old Intel i5-2400 PC, which I use for Telemetry blocking (apart from the usual ad-blocking features of Pi-Hole) using custom regexes I developed while working with a Squid proxy and a system to monitor those logs. This allows me to see egress traffic in real-time and check which domains are being allowed and which ones are blocked.

So, if I see a particular domain that shouldn't be allowed but is 'forwarded', I can quickly modify existing regexes or add a new rule. For e.g., I had unknown requests to `*.azure.cn` which can be addressed using a blocking regex `^(.*)\.cn`:

```
Mar  9 11:33:53 dnsmasq[48]: forwarded chinanorth3-0.in.applicationinsights.azure.cn to 84.2
00.69.80
Mar  9 11:33:54 dnsmasq[48]: forwarded chinanorth3-0.in.applicationinsights.azure.cn to 84.2
00.69.80
Mar  9 11:33:54 dnsmasq[48]: forwarded chinanorth3-0.in.applicationinsights.azure.cn to 84.2
00.69.80
Mar  9 11:33:54 dnsmasq[48]: forwarded chinanorth3-0.in.applicationinsights.azure.cn to 84.2
00.70.40
```

While trying to setup DeepSeek-R1 (which requires Ollama to be installed first), I noted SS wasn't working:



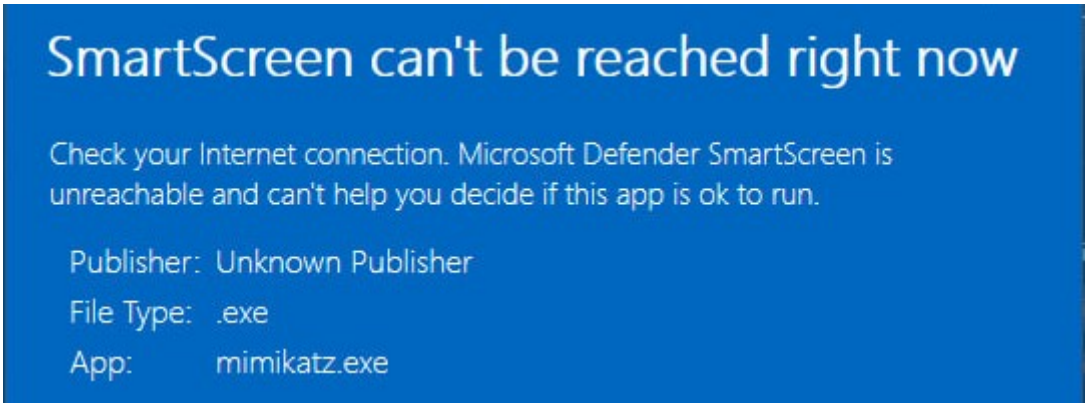
This is not unusual because it also pops up when a machine is not connected to the internet, but requests to which domains were being blocked wasn't clear to me. Checking my Pi-Hole logs I found `checkappexec.microsoft.com` was blocked:

```
regex denied checkappexec.microsoft.com is 0.0.0.0
regex denied checkappexec.microsoft.com is 0.0.0.0
```

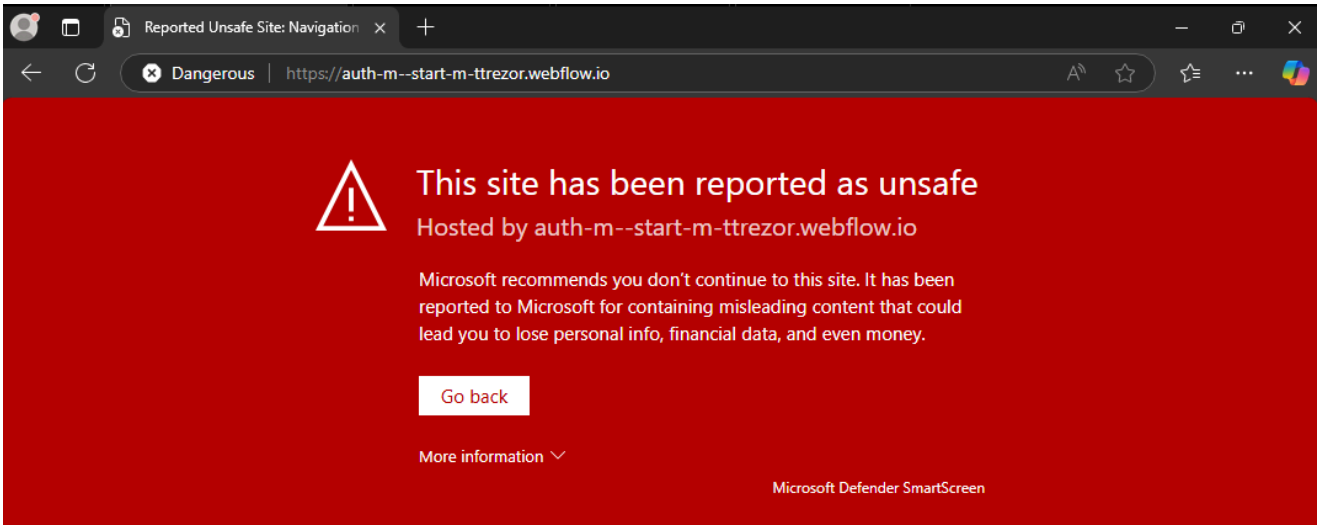
If I allow list this domain, there's no "SmartScreen can't be reached right now" pop-up indicating that SS uses this domain to match the current URL against a larger blocklist of malicious domains.

Outsmarting SmartScreen

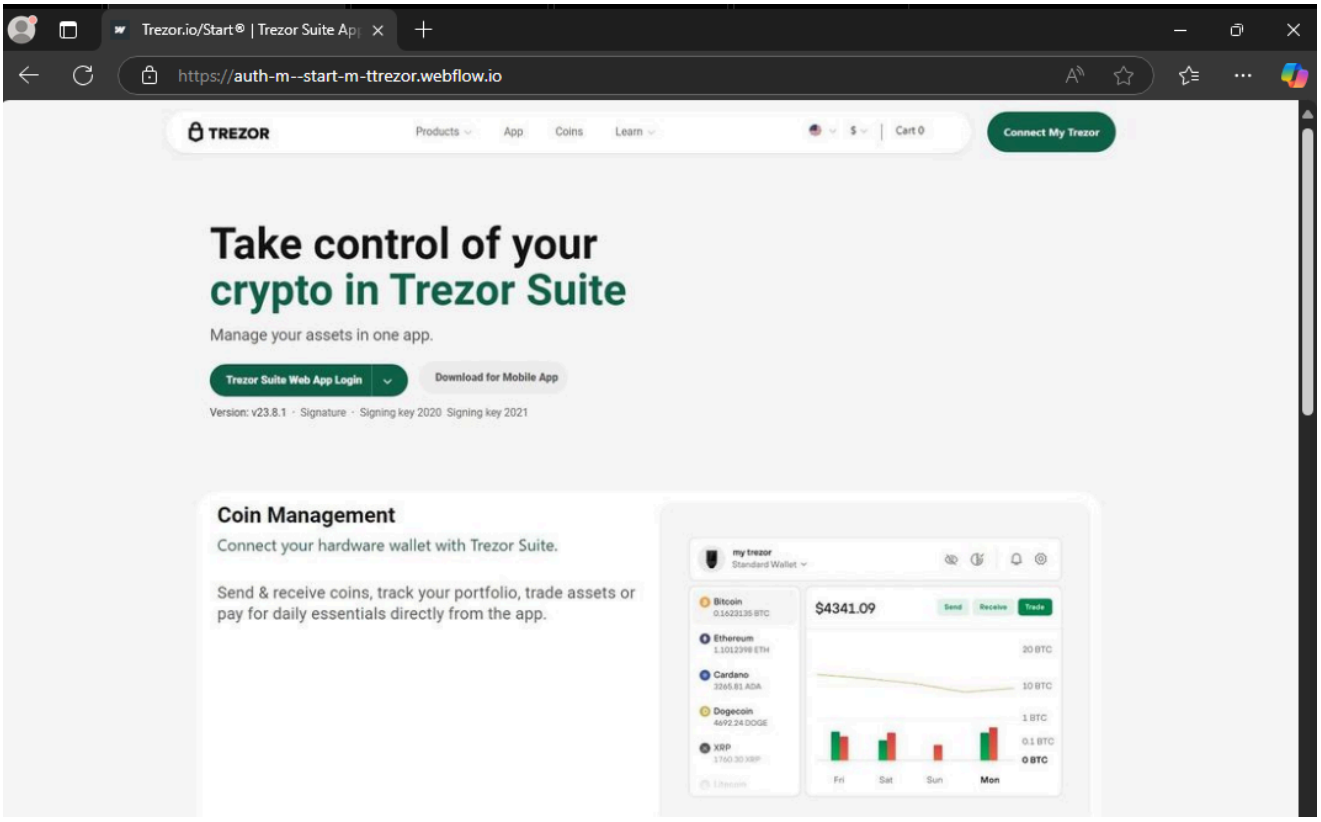
To confirm my findings, I decided to test Mimikatz and the result was similar:



To test the browser component of SmartScreen, I decided to test in Microsoft Edge and picked up a malicious domain. I disabled blocking on my Pi-Hole and got a warning:



After the I enabled blocking and navigated to the phishing site, there was no warning:



> I had to explicitly clear all browsing data after the site had been marked as malicious. Otherwise, even after enabling blocking the detection will still be there. This would mean SS uses caching.

Entries corresponding to this visit:

```
regex denied nav.smartscreen.microsoft.com is 0.0.0.0
regex denied nav.smartscreen.microsoft.com is 0.0.0.0
```

```
regex denied nav-edge.smartscreen.microsoft.com is 0.0.0.0
regex denied nav-edge.smartscreen.microsoft.com is 0.0.0.0
```

Implications for Engagements

If you have local Admin on a machine or more still, local admin on DC (domain admin), at which point you might as well disable SS via settings or GPO. But that would be noisy, instead of doing that, one can add DNS Query Resolution Policies on the DC to sinkhole these domains to **disable SmartScreen for the entire domain** (provided all machines are configured to use the DC as their DNS server which they are by default).

Using `*-DnsServerQueryResolutionPolicy` cmdlets we can configure blocking for `checkappexec.microsoft.com`, `nav.smartscreen.microsoft.com`, and `nav-edge.smartscreen.microsoft.com`.

```
Add-DnsServerQueryResolutionPolicy -Name "BlockCheckAppExec" -Action DENY -FQDN
"EQ,checkappexec.microsoft.com"

Add-DnsServerQueryResolutionPolicy -Name "BlockSS" -Action DENY -FQDN
"EQ,nav.smartscreen.microsoft.com"

Add-DnsServerQueryResolutionPolicy -Name "BlockSSEdge" -Action DENY -FQDN "EQ,nav-
edge.smartscreen.microsoft.com"
```

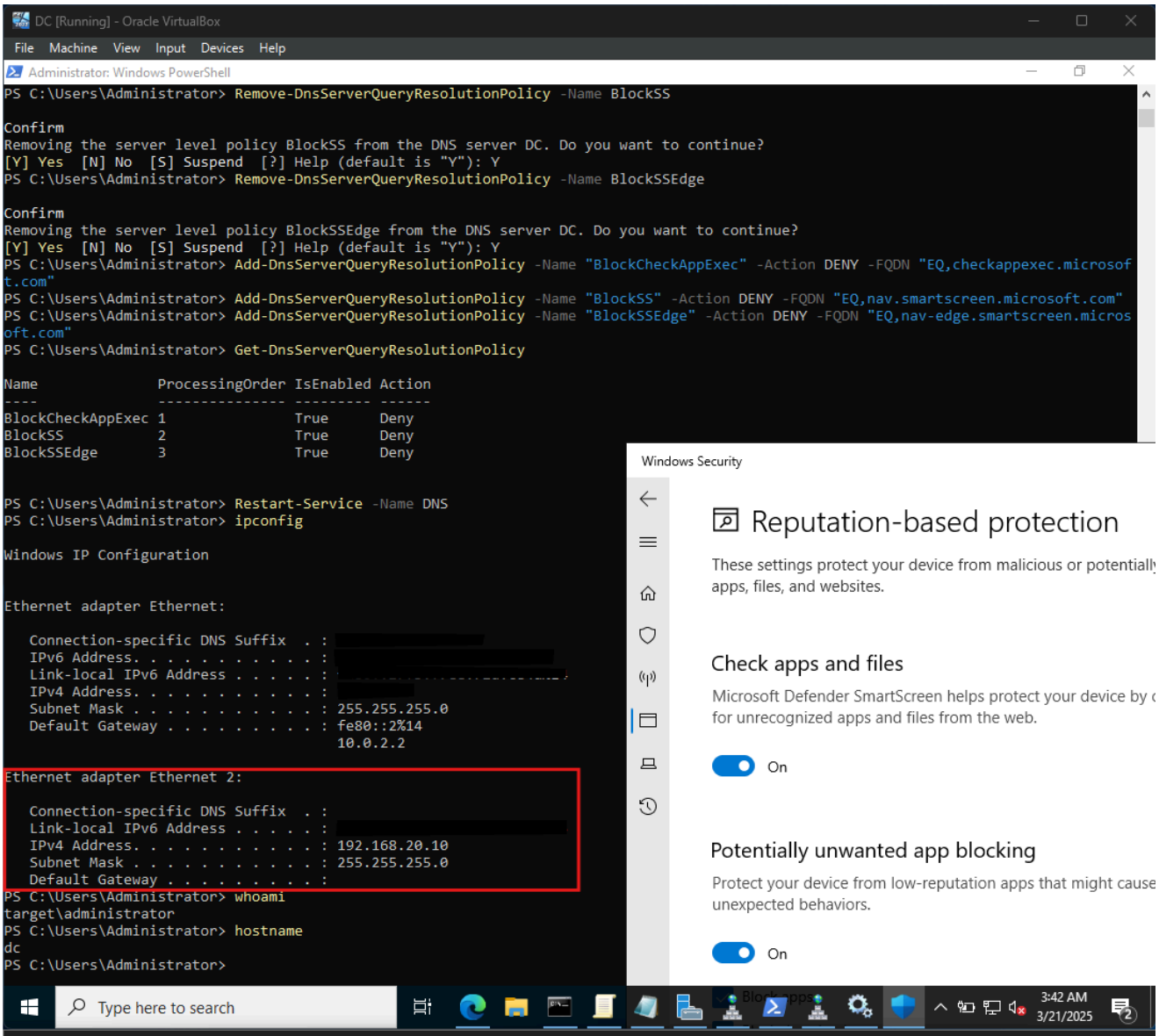
Check applied policies using `Get-DnsServerQueryResolutionPolicy`:

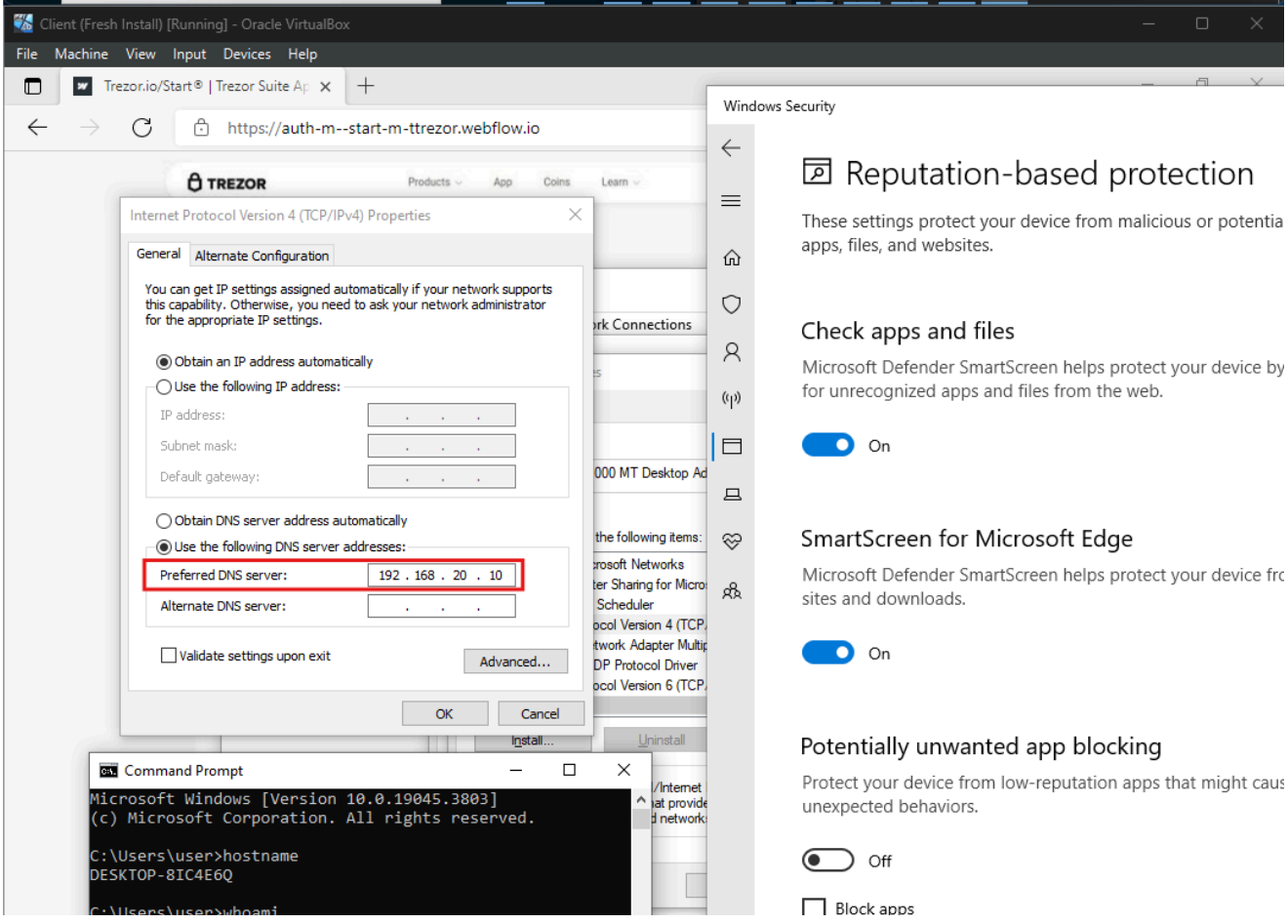
```
Get-DnsServerQueryResolutionPolicy

Name              ProcessingOrder IsEnabled Action
----              --------------- --------- ------
BlockCheckAppExec 1               True      Deny
BlockSS           2               True      Deny
BlockSSEdge       3               True      Deny
```

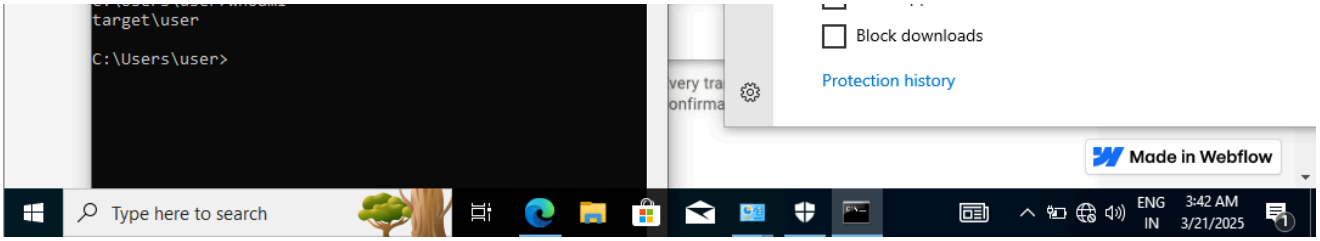
And restart the DNS as a final step:

```
Restart-Service -Name DNS
```

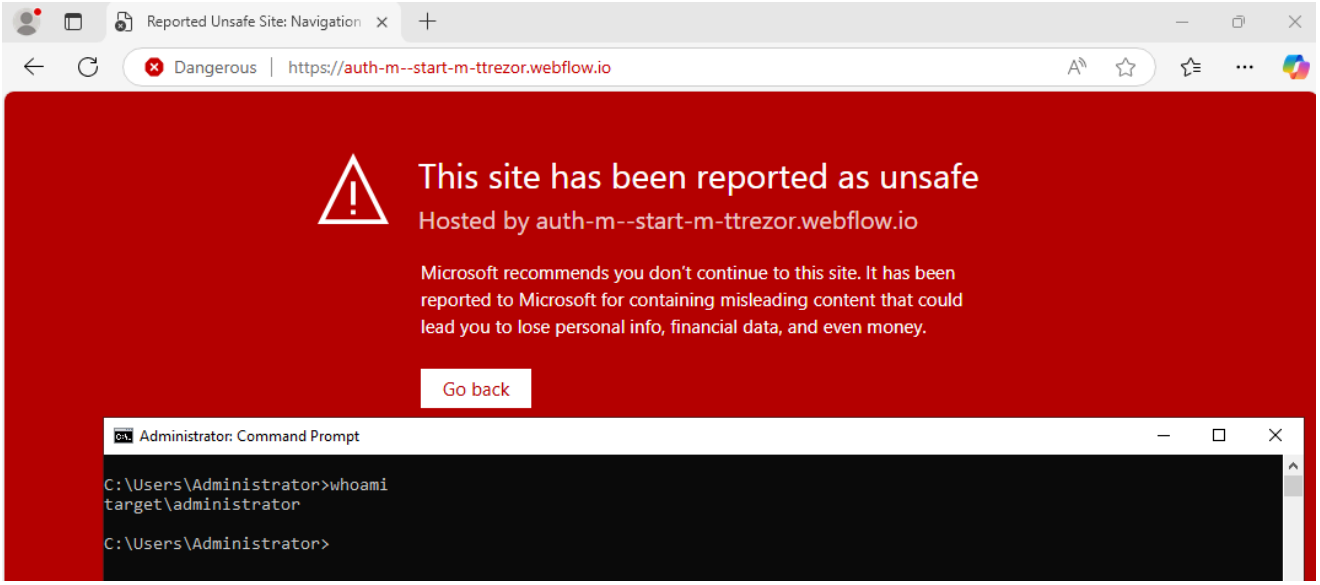
The screenshot below shows the DC at the top and a domain joined machine on the bottom. The client machine's adapter is configured to use DC's DNS as for name resolution. Despite Reputation-based controls being in place, it is evident that for anyone using this DNS SS won't be functional.


DC [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Administrator: Windows PowerShell
PS C:\Users\Administrator> Remove-DnsServerQueryResolutionPolicy -Name BlockSS
Confirm
Removing the server level policy BlockSS from the DNS server DC. Do you want to continue?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
PS C:\Users\Administrator> Remove-DnsServerQueryResolutionPolicy -Name BlockSSEdge
Confirm
Removing the server level policy BlockSSEdge from the DNS server DC. Do you want to continue?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
PS C:\Users\Administrator> Add-DnsServerQueryResolutionPolicy -Name "BlockCheckAppExec" -Action DENY -FQDN "EQ,checkappexec.microsoft.com"
PS C:\Users\Administrator> Add-DnsServerQueryResolutionPolicy -Name "BlockSS" -Action DENY -FQDN "EQ,nav.smartscreen.microsoft.com"
PS C:\Users\Administrator> Add-DnsServerQueryResolutionPolicy -Name "BlockSSEdge" -Action DENY -FQDN "EQ,nav-edge.smartscreen.microsoft.com"
PS C:\Users\Administrator> Get-DnsServerQueryResolutionPolicy
Name ProcessingOrder IsEnabled Action
BlockCheckAppExec 1 True Deny
BlockSS 2 True Deny
BlockSSEdge 3 True Deny
PS C:\Users\Administrator> Restart-Service -Name DNS
PS C:\Users\Administrator> ipconfig
Windows IP Configuration
Ethernet adapter Ethernet:
Connection-specific DNS Suffix . :
IPv6 Address. . . . . :
Link-local IPv6 Address . . . . . :
IPv4 Address. . . . . :
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : fe80::2%14
10.0.2.2
Ethernet adapter Ethernet 2:
Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . :
IPv4 Address. . . . . : 192.168.20.10
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :
PS C:\Users\Administrator> whoami
target\administrator
PS C:\Users\Administrator> hostname
dc
PS C:\Users\Administrator>
Windows Security
Reputation-based protection
These settings protect your device from malicious or potentially
apps, files, and websites.
Check apps and files
Microsoft Defender SmartScreen helps protect your device by
for unrecognized apps and files from the web.
On
Potentially unwanted app blocking
Protect your device from low-reputation apps that might cause
unexpected behaviors.
On
Type here to search
3:42 AM
3/21/2025


Client (Fresh Install) [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Trezor.io/Start® | Trezor Suite
https://auth-m--start-m-ttrezor.webflow.io
TREZOR
Internet Protocol Version 4 (TCP/IPv4) Properties
General
Alternate Configuration
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
Obtain an IP address automatically
Use the following IP address:
IP address:
Subnet mask:
Default gateway:
Obtain DNS server address automatically
Use the following DNS server addresses:
Preferred DNS server: 192 . 168 . 20 . 10
Alternate DNS server:
Validate settings upon exit
Advanced...
OK
Cancel
Command Prompt
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.
C:\Users\user>hostname
DESKTOP-8IC4E6Q
Windows Security
Reputation-based protection
These settings protect your device from malicious or potentia
apps, files, and websites.
Check apps and files
Microsoft Defender SmartScreen helps protect your device by
for unrecognized apps and files from the web.
On
SmartScreen for Microsoft Edge
Microsoft Defender SmartScreen helps protect your device fr
sites and downloads.
On
Potentially unwanted app blocking
Protect your device from low-reputation apps that might caus
unexpected behaviors.
Off
Block apps

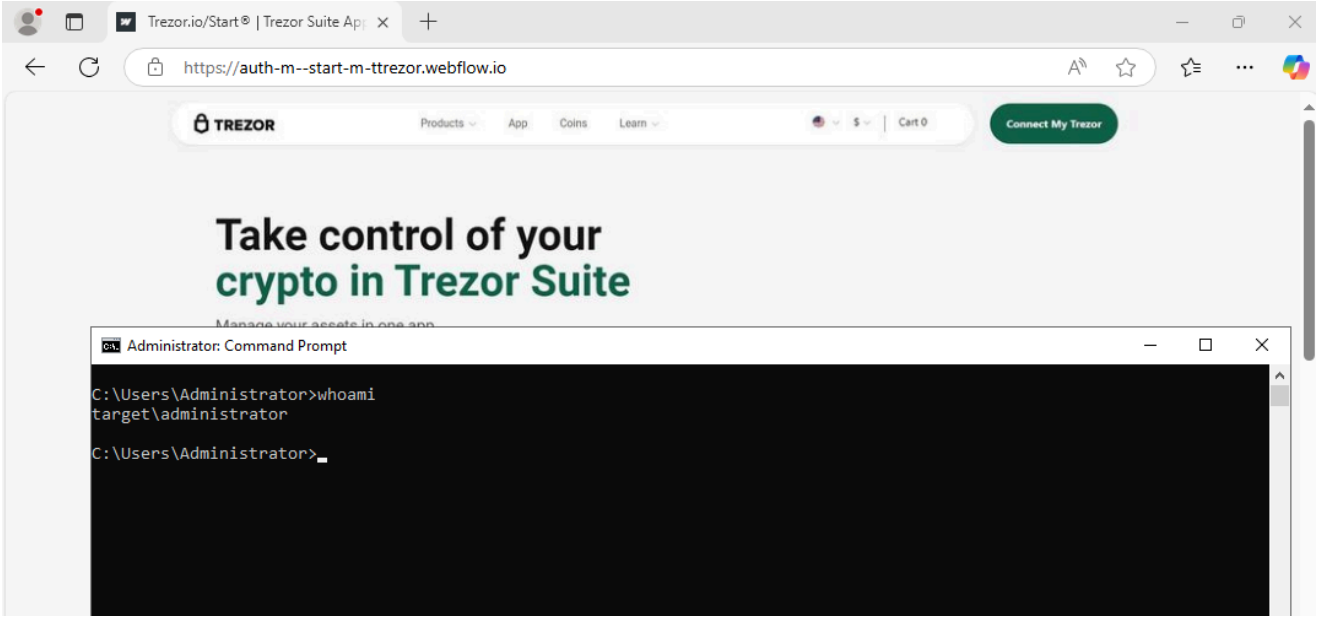


One important thing to note is that the changes in resolution policies doesn't affect the DC as I got a detection when visiting the malicious domain from the DC:



Fortunately, making these changes in the hosts file, clearing browser cache, leads to profit:

```
0.0.0.0 checkappexec.microsoft.com
0.0.0.0 nav.smartscreen.microsoft.com
0.0.0.0 nav-edge.smartscreen.microsoft.com
```



Blocking for Everyone

I had earlier shared my Squid proxy ACL in a gist. I'm sharing those again below as I haven't updated them lately but the ones I list below are the latest I use in conjunction with Pi-Hole.

1. Block analytics: `^(.*)\.browser-intake-datadoghq.com|^(.*)\.scorecardresearch.com|^(.*)\.braze.com|^(.*)\.analytics.yahoo.com`
2. Block Facebook: `^(.*)\.facebook.com|^(.*)\.meta.com|^(.*)\.facebook.net`
3. Block Mozilla telemetry: `incoming.telemetry.mozilla.org`
4. Block Mozilla location services: `location.services.mozilla.com`
5. Block Google: `^(.*\.)(think)?with)google($|((adservices|apis|mail|static|syndication|tagmanager|tagservices|usercontent|zip⊢analytics)($|\..+)))|^(.*\.|^)g(gpht|mail|v(t[12])?)($|\..+)|^(.*\.|^)chrom(e(experiments)?|ium)($|\..+)|^(.*\.|^)doubleclick($|\..+)|^(.*\.|^)firebaseio($|\..+)|^(.*\.|^)waze($|\..+)`
6. Block Microsoft: `^(.*)\.microsoft.net|^(.*)\.msecnd.net|^(.*)\.azure.com|^(.*).msn.com|^(.*)\.windows.net|^(.*)\.microsoft.com|^(.*)\.s-microsoft.com|^(.*)\.skype.com|^(.*)\.a-msedge.net|^(.*)\.msn.net|^(.*)\.doubleclick.net|^(.*)\.adnxs.net|^(.*)\.msads.net|^(.*)\.data\.microsoft.com|^(.*)\.live.com|^(.*)\.outlook.com|^(.*)\.nstac.net|^(.*)\.cloudapp.net|^(.*)\.windows.com|^(.*)\.akadns.net|^(.*)\.msedge.net|^(.*)\.msftncsi.com|^(.*)\.serving-sys.com|^(.*)\.bing.net|^(.*)\.bing.com|^(.*)\.visualstudio.com|^(.*)\.azure.net`

If you don't want to block Microsoft completely (on an engagement for example) you can block specific domains `*.smartscreen.microsoft.com` and `checkappexec.microsoft.com` (hosts file doesn't support regex or wildcards so domains have to be mapped directly). Or if you have a Pi-Hole on your network you can simply do: `^(.*)\.smartscreen.microsoft.com` and `checkappexec.microsoft.com` although there's little reason you might want to do this because it might be dangerous if you happen to click here and there on a malicious site or simply exclude these domains if you happen to be using the above regex.

Detection

This method to suppress SS relies on mapping specific domains to `0.0.0.0`. Enhanced monitoring of the hosts file is required to detect any attempts at sinkholing. I'm not sure how `C:\Windows\System32\drivers\etc\hosts` is monitored even though I found a deprecated detection from Splunk: Detection: Windows hosts file modification | Splunk Security Content

In future, I might look into WD telemetry because despite disabling "Automatic Sample Submission" there are detections, which is not suspicious at all ;)

That's it! Thanks for reading.

Tx0actical. Out.