Building a Canarytoken to Monitor Windows Process Execution

THINKST CANARY

Who Are We?

We are [the budding] Thinkst Labs:

Research, learning, and teaching
ThinkstScapes (thinkst.com/ts)
Both in CO, enjoying its outdoors

Casey

Jacob

Topics

Canarytokens History / background

Research Approach

Command Execution Alert

Idea to delivery

Canarytokens Background

Canarytokens History / Background

Free / Open Source

Simple Objects

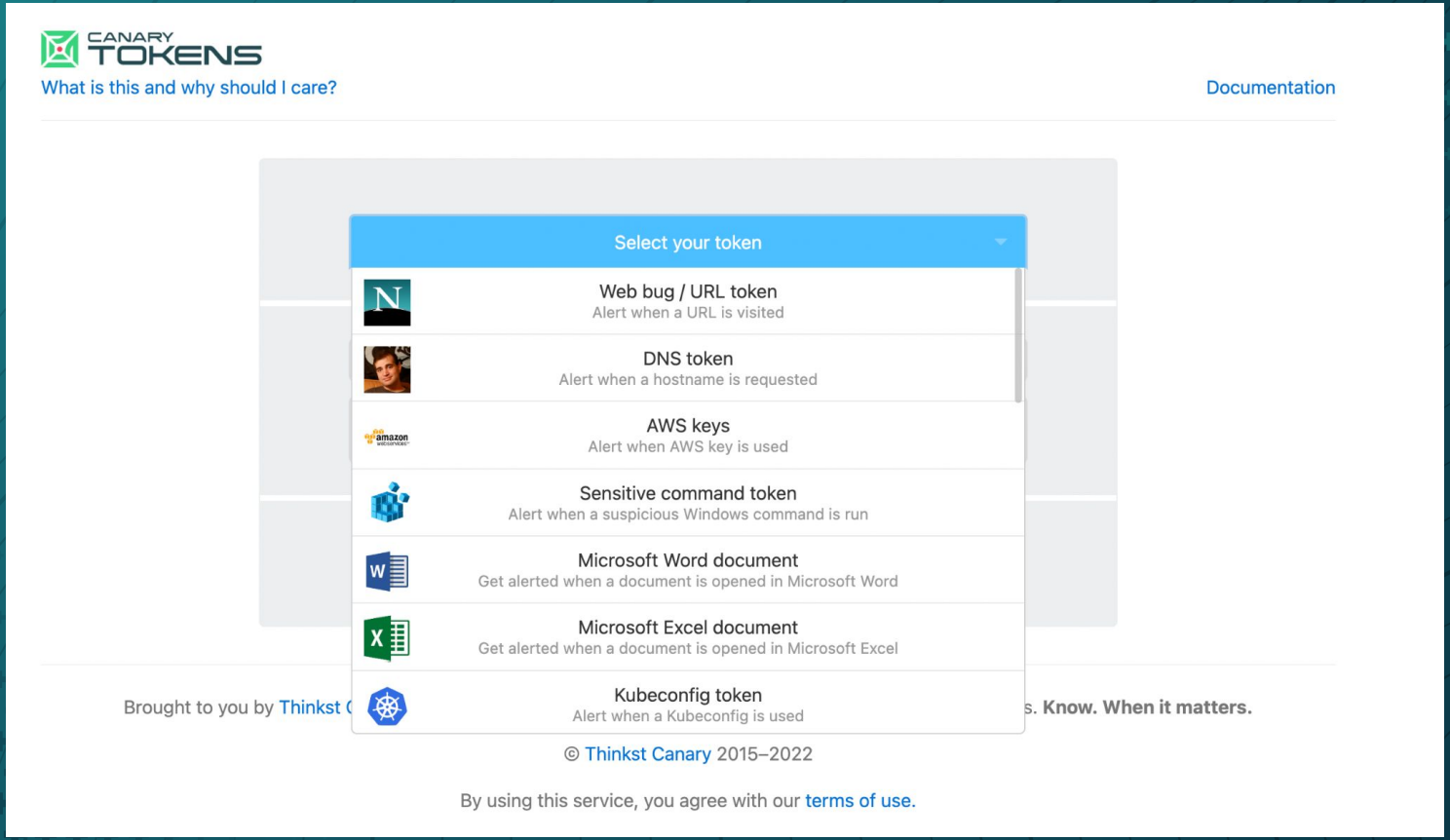
Create alert on access

DNS / HTTP primitives

Surprisingly Effective

https://canarytokens.org/generate

Canarytoken Example



to casey+cactusconDemo

Canarytoken triggered

ALERT

A DNS Canarytoken has been triggered by the Source IP 192.168.32.1. Please note that the source IP refers to a DNS server, rather than the host that triggered the token.

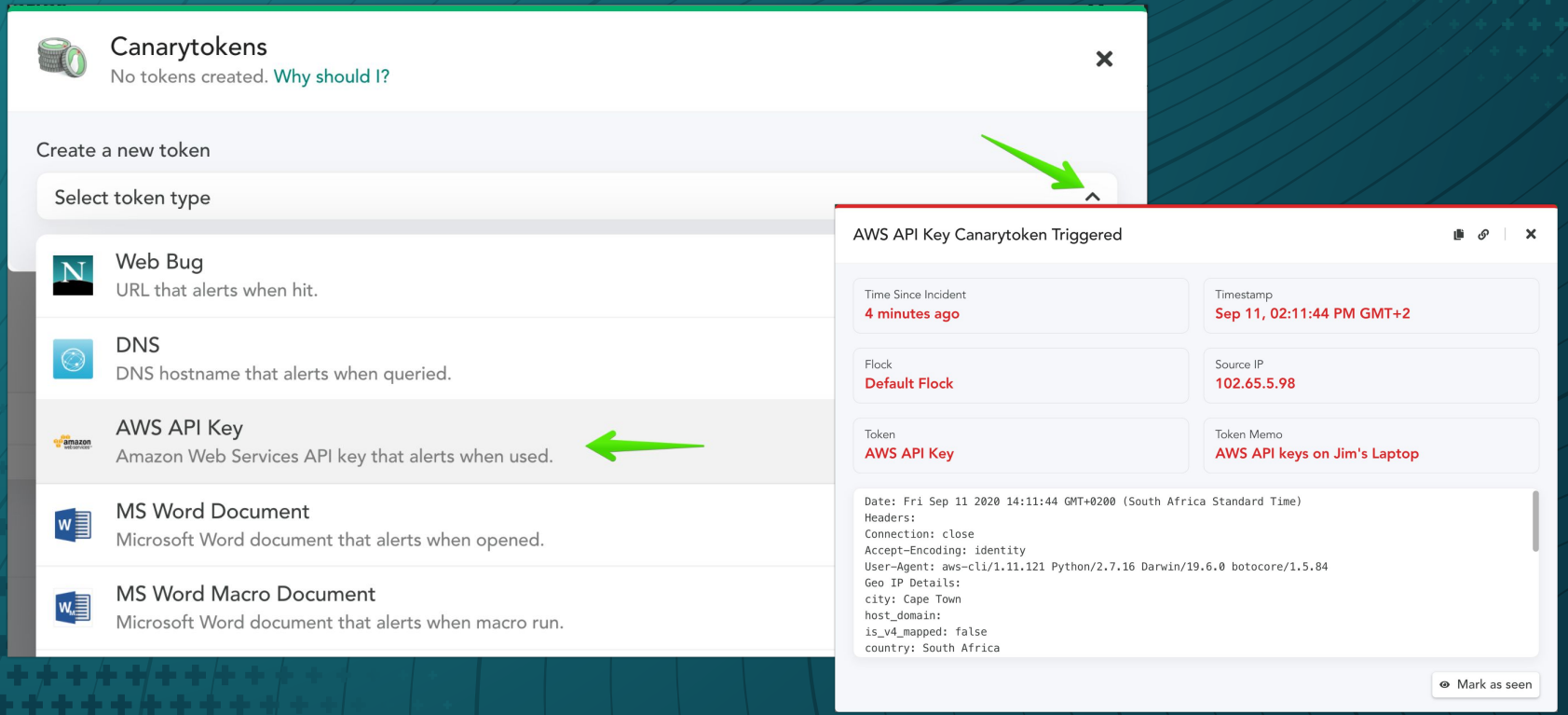
Basic Details:

Channel	DNS
Time	2023-01-04 16:48:32 (UTC)
Canarytoken	pvd6nto02timeqnrpg312345
Token Reminder	nltest.exe example for Cactus Con (This token was created to monitor the execution of: nltest.exe)
Token Type	cmd
Source IP	192.168.32.1
Sensitive Command Information	User casey executed "nltest.exe" on the host

Canarytoken Management Details:

Manage this Canarytoken here
More info on this token here

Attackers Use It / You Are Alerted

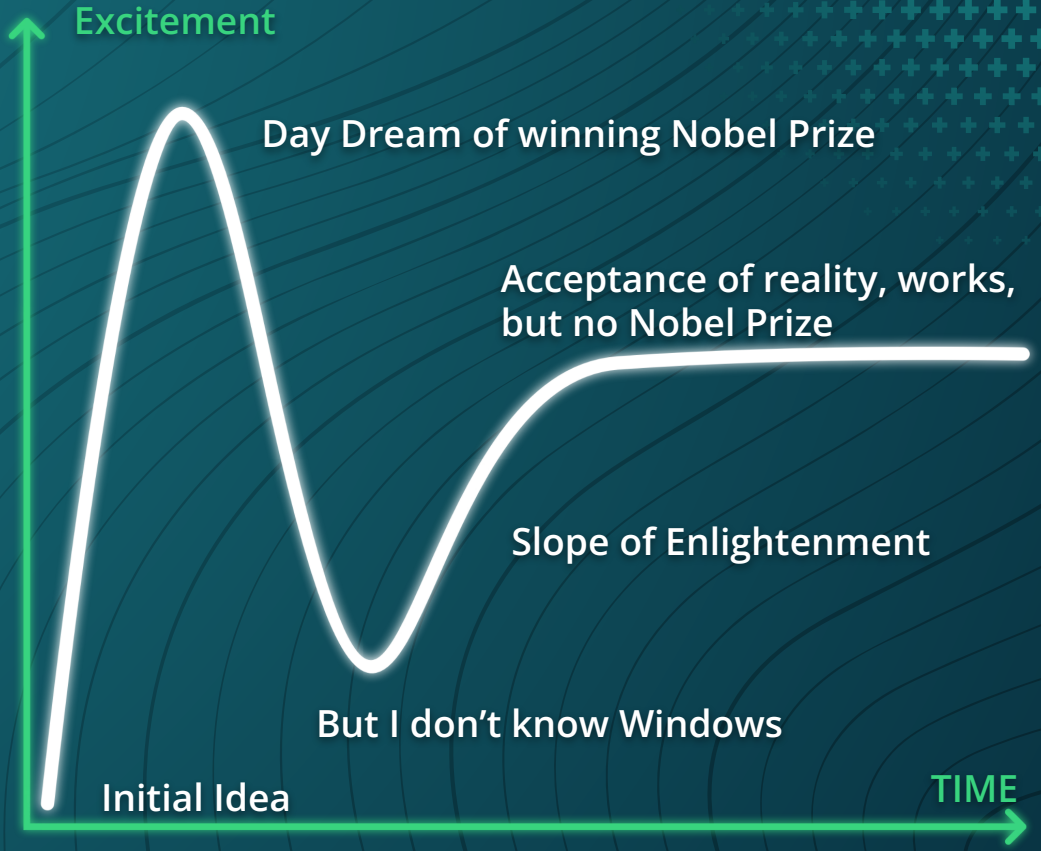


Research Approach

Research Insights - Fragility of Early Ideas

Early ideas:

- DOSKeys
- Event 4688
- Scheduled Task
- WMI
- nslookup



Idea Generation / Collaboration

Thinkst is a remote-first company, always working

Slack WhiteBoards / App

Asynchronous

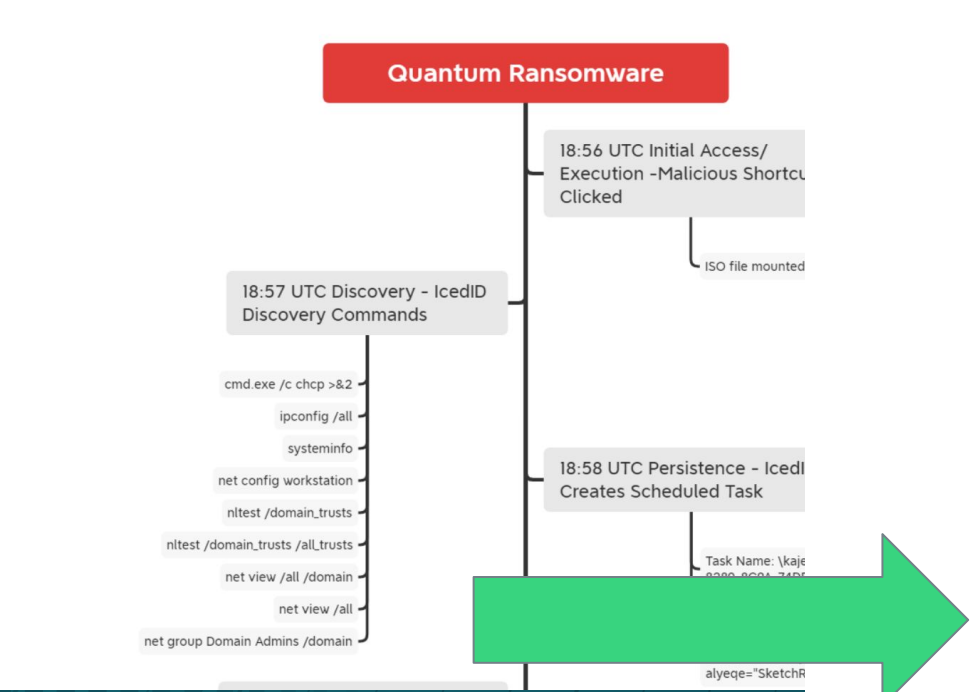
Ok to fail

Command Execution Alert

What do we want to build?

Could we build a SIMPLE token for teams, that will alert on execution of rare commands?

DFIR Report



https://thedfirreport.com/2022/04/25/quantum-ransomware/

Reading DFIR Ransomware Reports

- It is clear there are common exe's attackers run
- "nltest /trusted_domains"
- Who [legitimately] runs this???

- **Could we build a SIMPLE token for teams, that will alert on execution?**

Multiple commands responsible for enumerating Active Directory groups, domain joined computers, and domain trusts, were executed via Cobalt Strike on the beachhead.

```
whoami /groups
net group /domain
net group "domain computers" /domain
net group /domain "Domain controllers"
net group "domain admins" /domain
nltest /trusted_domains
```

The threat actor was observed querying a non-existent group `Domain controller`, followed by a command correcting the mistake that queried the group `Domain controllers` .

```
net group /domain "Domain controller"
net group /domain "Domain controllers"
```

In nearly every report on ransomware we see sequences of commands.

Many of them rarely or never used by ordinary people day to day...

But HOW would we create a simple alert?

Trial and Error - Prototypes

Let’s start experimenting, what command is best?

What is the most simple way to install?

Idea to Delivery

Requirements

Easy To Deploy

This doesn’t replace complex EDR and SIEM alert behavioral detection.

Low False Positive Rate

Some commands are more rare and make better alerting.

First Attempts In Learning - FAIL

doskey alias?

doskey

Article • 03/03/2021 • 8 minutes to read • 8 contributors

Feedback

Calls Doskey.exe, which recalls previously entered command-line commands, edits command lines, and creates macros.

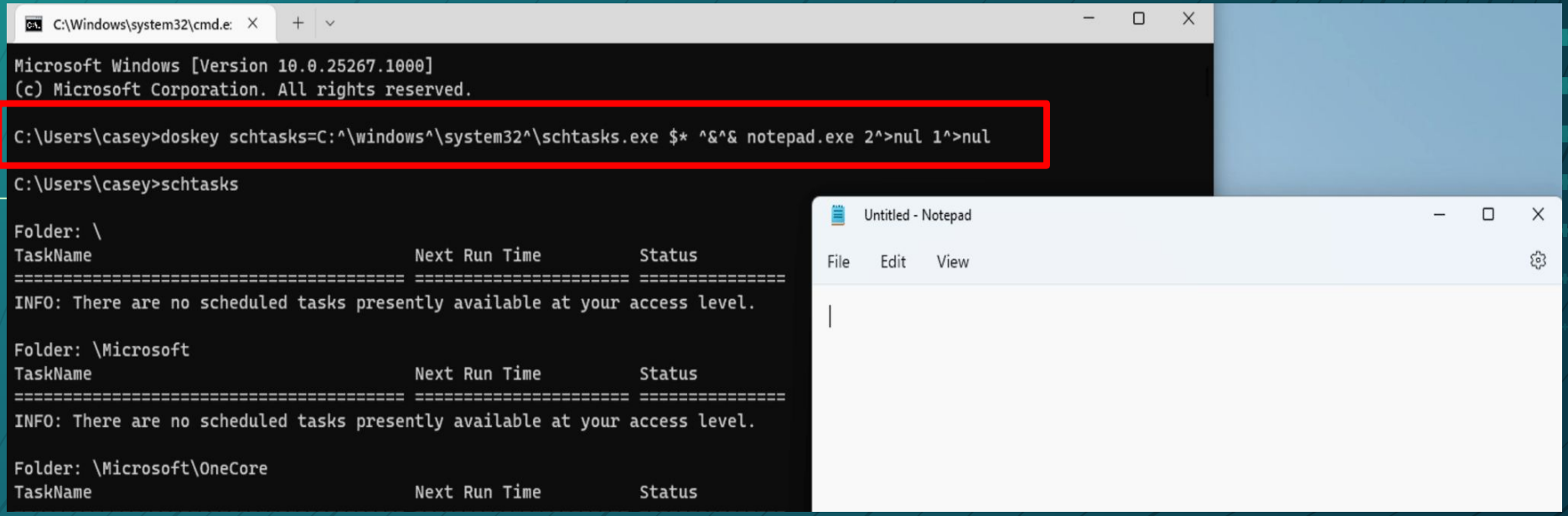
Specifically Macros/Alias

DOSKey examples ^

Some common examples to help you be slightly more productive are as follows:

Alias	What It Does
doskey d=dir $*	Lists a directory
doskey n=notepad $*	Opens a file with Notepad (for example, n text.txt)
doskey ..=cd ..\..	Changes to the parent directory

https://4sysops.com/archives/using-doskey-aliases/
https://devblogs.microsoft.com/oldnewthing/20071121-00/?p=24433



This is CLOSE to what we want
Run Scheduled Task and...
notepad in the background

Why it fails

Not all shells persist or use the doskey, so RDP or remote execution, won't use the doskey alias…

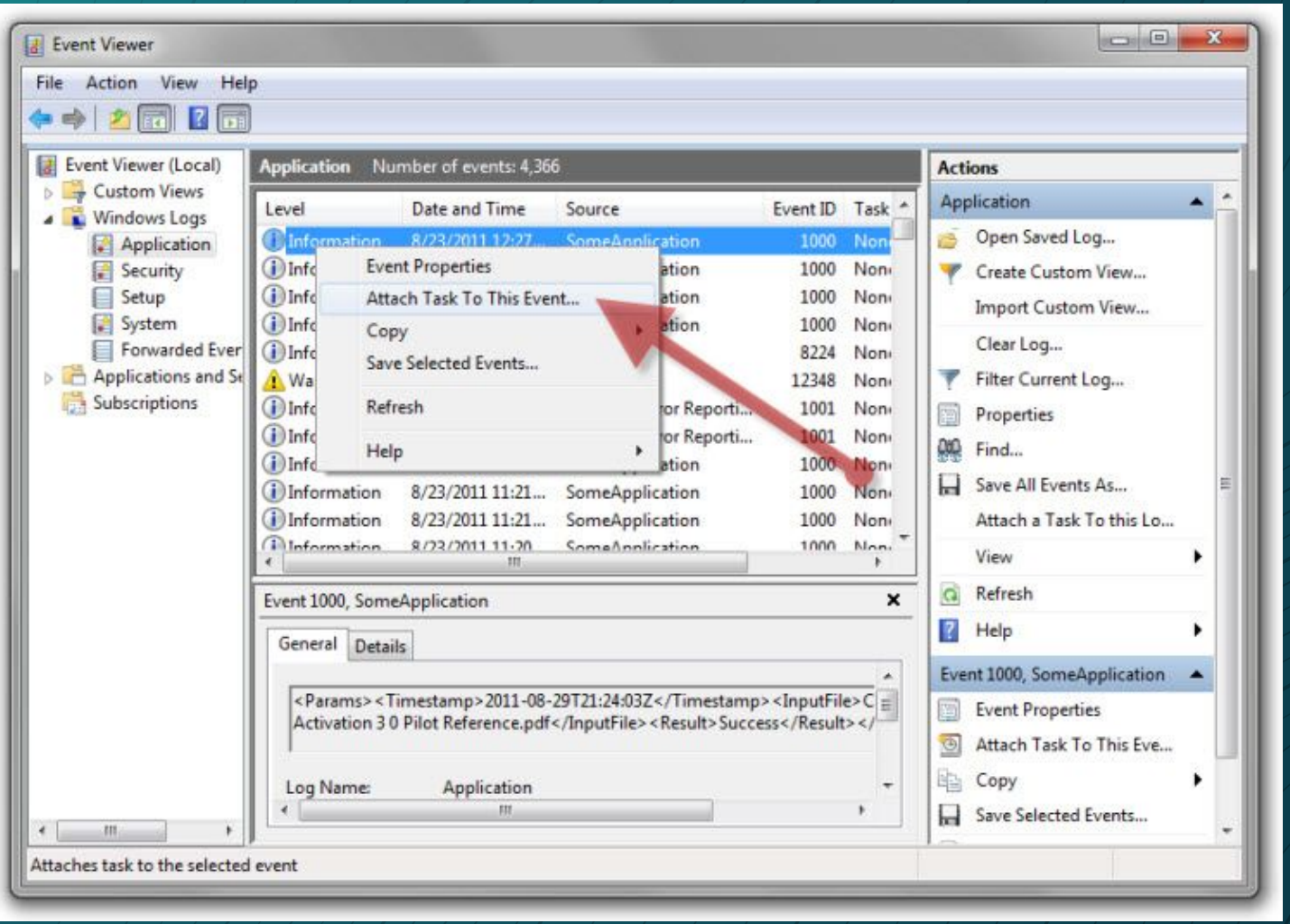
It fails because its not global on the host and risky to rely on.

Event 4688 - Scheduled Task

You can attach a task to an event.

Attempt to monitor Windows Event Logs and alert.

https://learn.microsoft.com/en-us/archive/blogs/wincat/trigger-a-powershell-script-from-a-windows-event


Event Viewer
File Action View Help
Event Viewer (Local)
Custom Views
Windows Logs
Application
Security
Setup
System
Forwarded Ever
Applications and Se
Subscriptions
Application Number of events: 4,366
Level Date and Time Source Event ID Task
Information 8/23/2011 11:21... SomeApplication 1000
Information 8/23/2011 11:21... SomeApplication 1000
Event Properties
Attach Task To This Event...
Copy
Save Selected Events...
Refresh
Help
Event 1000, SomeApplication
General Details
<Params><Timestamp>2011-08-29T21:24:03Z</Timestamp><InputFile>C
Activation 3 0 Pilot Reference.pdf</InputFile><Result>Success</Result></
Log Name: Application
Actions
Application
Open Saved Log...
Create Custom View...
Import Custom View...
Clear Log...
Filter Current Log...
Properties
Find...
Save All Events As...
Attach a Task To this Lo...
View
Refresh
Help
Event 1000, SomeApplication
Event Properties
Attach Task To This Eve...
Copy
Save Selected Events...
Attaches task to the selected event

Why this fails

Not simple

Too Noisy, its per EVENT, then you have to filter by exe

Too Many moving parts

When you get stuck perhaps ask?

What would a developer do?

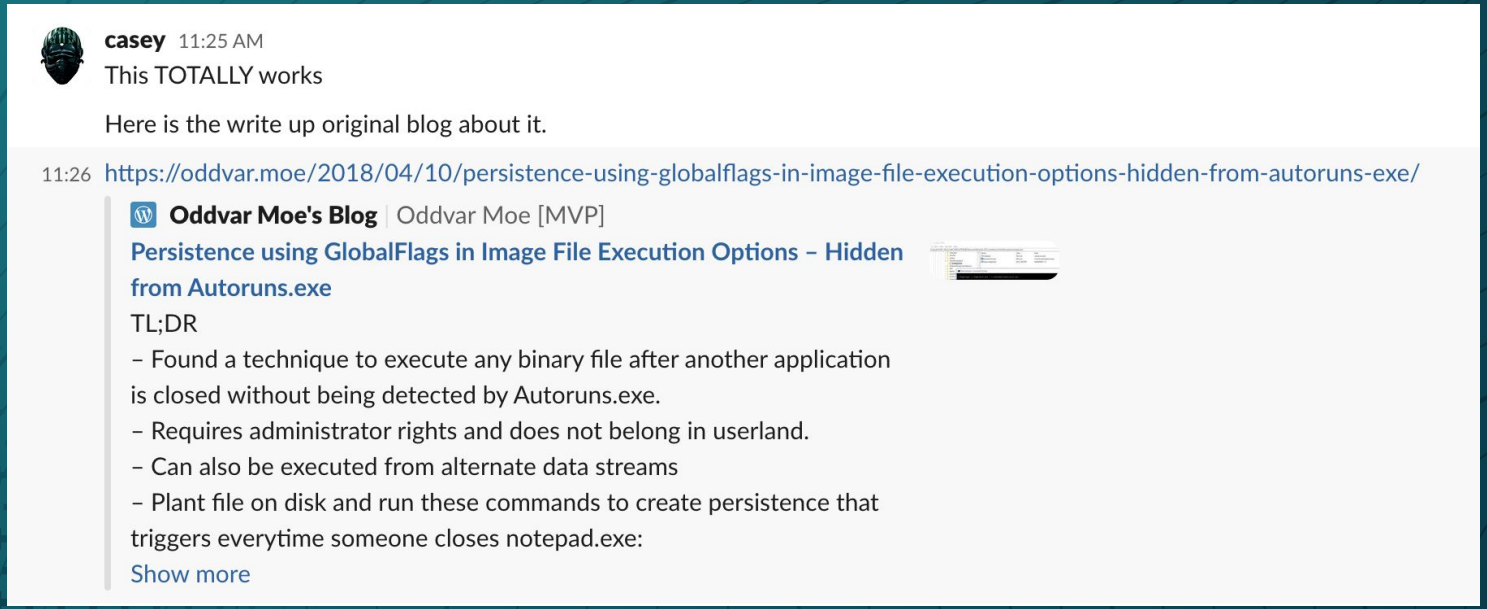
What would an attacker do?

Hmmm...

Image File Execution Options

https://oddvar.moe/2018/04/10/persistence-using-globalflags-in-image-file-execution-options-hidden-from-autoruns-exe/

So it sat for a while… Then "This WORKS!"



Monitoring Silent Process Exit

Monitor Process

You can specify a monitor process by entering a process name, along with command line parameters, in the **Monitor Process** text box. You can use the following variables in your command line.

Varaible	Meaning
%e	ID of the exiting process. This is the monitored process that exited silently.
%i	ID of the initiating process. In the case of self termination, this is the same as the exiting process. In the case of cross-process termination, this is the ID of the process that caused the termination.
%t	ID of the initiating thread. This is the thread that caused the termination.
%c	The status code passed to **ExitThread** or **TerminateThread** .

For example, the following value for **Monitor Process** specifies that on silent exit, WinDbg is launched and attached to the exiting process.

https://learn.microsoft.com/en-us/windows-hardware/drivers/debugger/registry-entries-for-silent-process-exit

Run a command, Run another Command In the Background!

Examples:

nltest.exe => nslookup.exe <canarytoken>

nltest.exe =>

powershell.exe -command Resolve-DNSName <canarytoken>

Why is this not working?

%computername%.%username%.hjgwzjkvh8ak8x168675.canarytokens.com

“This works, **but I don't see the leading parameters** in the alert, just noting???” Why?

Oh Wait? nslookup Won't work 😢

nltest.exe => nslookup.exe <canarytoken>

Why does this fail?

%C - Status Code

collision

%computername

Monitor Process

You can specify a monitor process by entering a process name, along with command line parameters, in the **Monitor Process** text box. You can use the following variables in your command line.

Varaible	Meaning
%e	ID of the exiting process. This is the monitored process that exited silently.
%i	ID of the initiating process. In the case of self termination, this is the same as the exiting process. In the case of cross-process termination, this is the ID of the process that caused the termination.
%t	ID of the initiating thread. This is the thread that caused the termination.
%c	The status code passed to **ExitThread** or **TerminateThread** .

Encoding over DNS with PowerShell

We want to collect some artifacts in the alert

Username

Computername

DNS Encoding

Encoding additional information in your token

Your DNS token can carry a small amount of additional custom data when it's triggered. This can be used for adding incident-specific data to your alert with custom DNS based tokens. Use the following encoding rules to place generic data into your DNS token:

- Base32 encode your data, and remove any padding '=' characters
- Insert periods (.) after every 63-bytes
- Append the magic string '.G'+<2-random-digits>+'.' (e.g. '.G12.' or '.G83.')
- Append your DNS token This creates a new hostname of the form:

```
<base32-string>.<base32-string>.G<2-random-digits>.<dns-token>
```

Bear in mind the total length of the hostname still cannot exceed 253-bytes, so the amount of raw bytes that can be encoded is ~125.

https://docs.canarytokens.org/guide/dns-token.html#encoding-additional-information-in-your-token

Remove Spaces and Non-Unicode

A process run as LOCAL SERVICE. Or any other username.

Needs to be condensed/stripped of spaces

So we can use it in a DNS name.
localservice

Operational Caveats / Impact

Not a replacement for EDR or SIEM

Alerting on cmd / PowerShell cause infinite loops

Some Anti-Malware/Endpoint Protection will detect/alert on token installation / reg setup.

Some realities we had deal with

DNS Duplicate Lookup Suppression / You just won't believe it

AV / EDR Alerts - It looks bad, so be sure you know that

Commands aren't are rare as you might think/suspect.

What may be exposed?

Username

Hostname

Executable

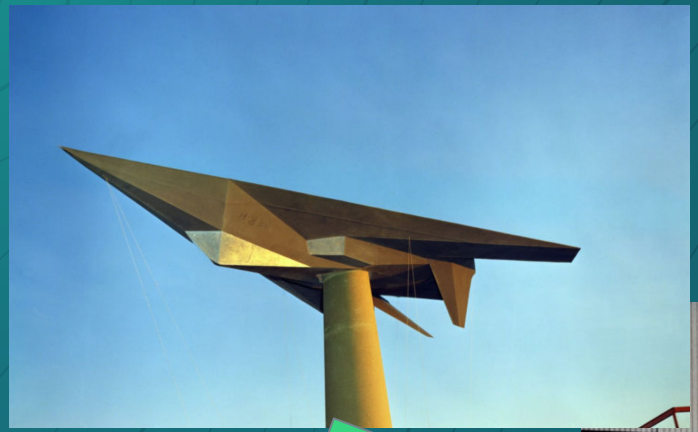
These may appear in public databases, and DNS

Keep that in mind.

Research is Done

Research (what don't we know) and the engineering that depends on it.

Once we had the single exec we were "done" with research, then it was engineering/finesse to get it over the line



Final Product!

The Sensitive Command Token

Choose A Command To Monitor

Candidates are:
Low frequency/rare
Short lived (run then exit)

BAD
cmd, powershell, outlook

BETTER
nltest, systeminfo, whoami

Create Token

Canarytokens.org

Sensitive Command

Input Email, Note and Process
To monitor.

Import Registry

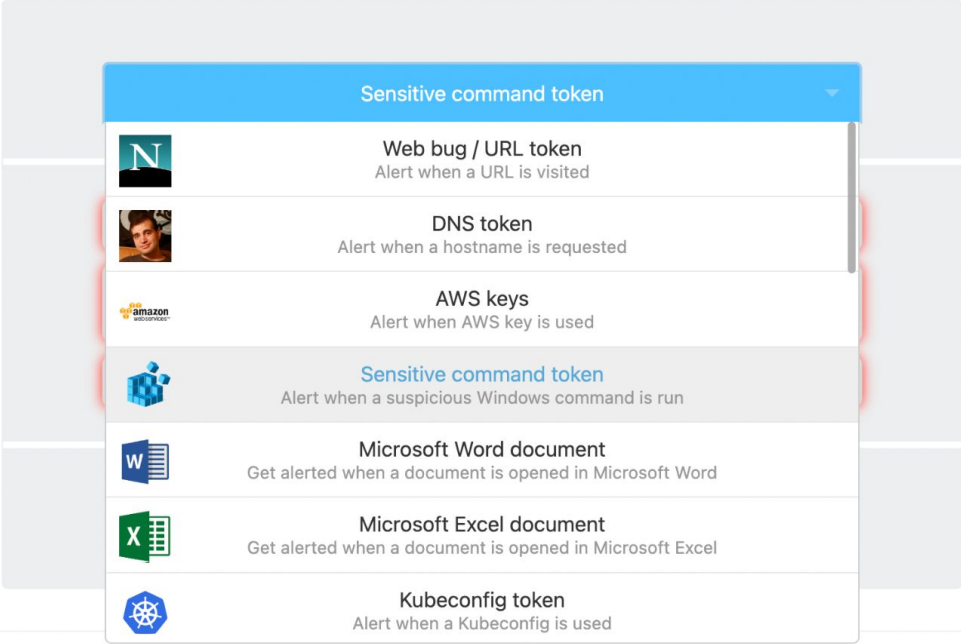
Requires Administrator
reg import canary.reg

Can be deployed to multiple machines

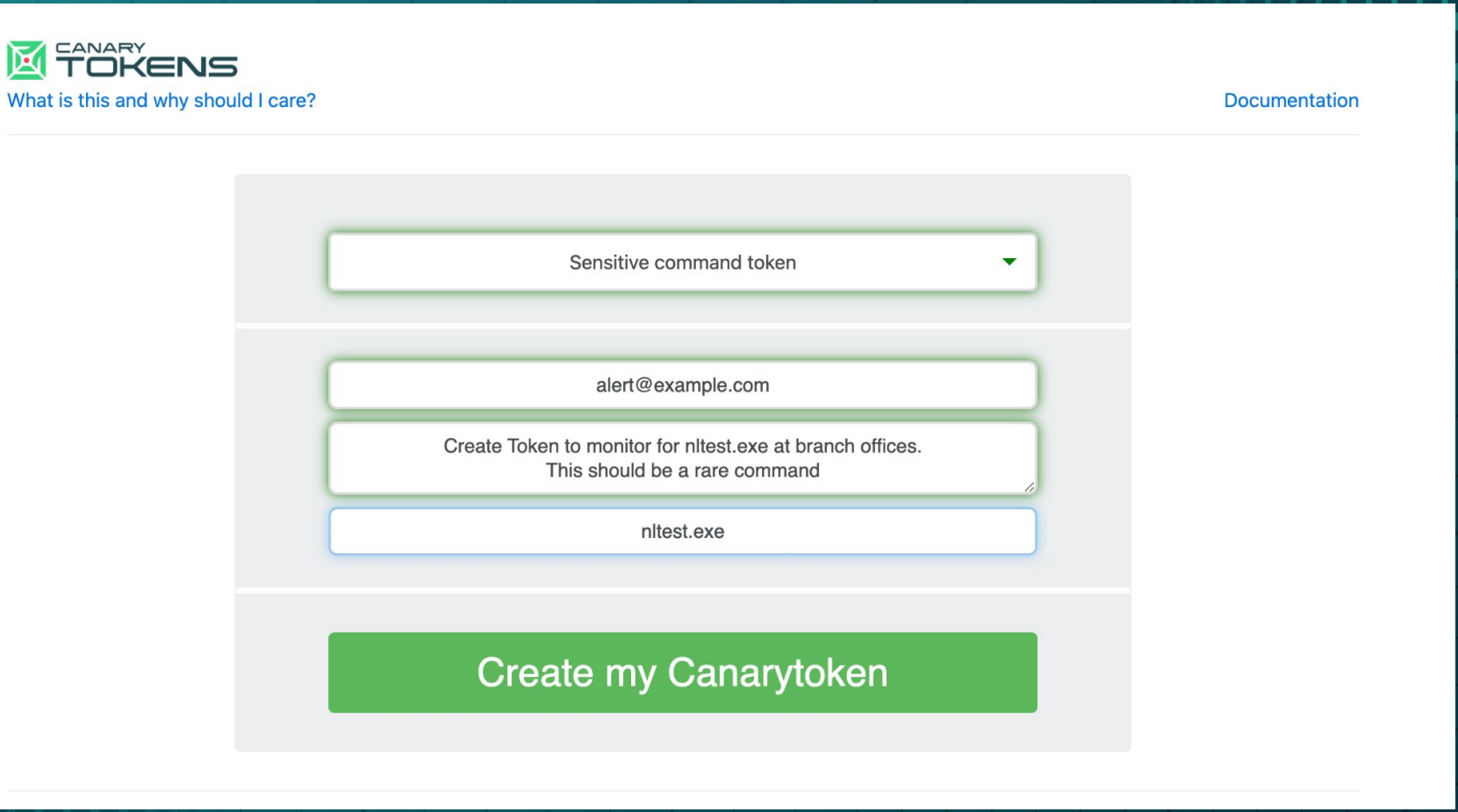
Works even if file is not present.

CANARY TOKENS

What is this and why should I care? Documentation



Brought to you by Thinkst Canary, our insanely easy-to-use honeypot solution that deploys in just 3 minutes. **Know. When it matters.**


CANARY TOKENS
What is this and why should I care?
Documentation
Sensitive command token
alert@example.com
Create Token to monitor for nltest.exe at branch offices.
This should be a rare command
nltest.exe
Create my Canarytoken

Your sensitive process execution token is active!

Download your MS registry file

Once installed (with admin permissions) you'll get an alert whenever someone (or someone's code) runs your sensitive process. It will automatically provide the command used, computer the command ran on, and the user invoking the command.

Ideas for use:

- Ideal candidates are executables often used by attackers but seldom used by regular users (e.g., whoami.exe, net.exe, wmic.exe, etc.).
- You can use this for attacker tools that are not present on your system (e.g., mimikatz.exe), and if they are ever downloaded and run you'll get an alert!
- Use a network management tool to deploy across your organization.

Canarytokens delivers a .reg

```
Windows Registry Editor Version 5.00
; Sensitive command token generated by Thinkst Canary
; Run with admin privs on Windows machine as: reg import FILENAME

; command that will be watched for
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File
Execution Options\nltest.exe]
"GlobalFlag"=dword:00000200

; magic unique canarytoken that will be fired when this command is executed
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SilentProcessExit
\nltest.exe]
"ReportingMode"=dword:00000001
"MonitorProcess"="cmd.exe /c start /min powershell.exe -windowstyle hidden -command
\"$($u=$(\\\"u$env:username\\\" -replace('[^\\x00-\\x7f]|\\s', ''))[0..63] -join '';
$c=$(\\\"c$env:computername\\\" -replace('[^\\x00-\\x7f]|\\s', ''));Resolve-
DnsName -Name \\\"$c.UN.$u.CMD.pvd6nto02timeqnrpg8675309.canarytokens.com\\\")\""
```

We will break this down shortly

Enable silent process exit monitoring

Article • 12/14/2021 • 2 minutes to read • 1 contributor

Feedback

The **Enable silent process exit monitoring** flag enables silent exit monitoring for a process.

Hexadecimal value	0x200
Symbolic Name	FLG_MONITOR_SILENT_PROCESS_EXIT
Destination	Image file registry entry

Reporting Mode

The **Reporting Mode** setting is available as an application setting, but not as a global setting. You can use the following check boxes to set the reporting mode.

Launch monitor process Enable dump collection Enable notification The **ReportingMode** registry entry is a bitwise OR of the following flags.

Flag	Value	Meaning
LAUNCH_MONITORPROCESS	0x1	When silent exit is detected, the monitor process (specified in the **Monitor Process** box) is launched.
LOCAL_DUMP	0x2	When silent exit is detected, a dump file is created for the monitored process. In the case of cross-process termination, a dump file is also created for the process that caused the termination.
NOTIFICATION	0x4	When silent exit is detected, a pop-up notification is displayed.

Monitor Process

You can specify a monitor process by entering a process name, along with command line parameters, in the **Monitor Process** text box. You can use the following variables in your command line.

Varaible	Meaning
%e	ID of the exiting process. This is the monitored process that exited silently.
%i	ID of the initiating process. In the case of self termination, this is the same as the exiting process. In the case of cross-process termination, this is the ID of the process that caused the termination.
%t	ID of the initiating thread. This is the thread that caused the termination.
%c	The status code passed to **ExitThread** or **TerminateThread** .

Using a debug / attacker persistence technique.

We can easily use this to silently alert us upon execution of a specified exe.

Break it down

cmd.exe start /min powershell.exe /hidden

- Start a Hidden PS Window
- Some redundancy here, we do not want popup to show

Powershell.exe constructs a DNS Name Lookup

- Remove non supported chars
- DNS only supports ascii.

End Goal - DNS Alert, user/computer name

```
cmd.exe /c start /min powershell.exe -windowstyle hidden -command
"$($u=$(\"u$env:username\" -replace('[^\x00-\x7f]|\s', ''))[0..63] -join '';
$c=$(\"c$env:computername\" -replace('[^\x00-\x7f]|\s', ''));
Resolve-DnsName -Name \"$c.UN.$u.CMD.pvd6nto02timeqnrp8675309.canarytokens.com\")"
```

```
PS C:\Users\casey> $u = $env:USERNAME

PS C:\Users\casey> $c = $env:COMPUTERNAME

PS C:\Users\casey> Write-host -ForegroundColor Green "$c.UN.$u.CMD.pvd6nto02timeqnrp8675309.canarytokens.com"
CASEYBCD8.UN.casey.CMD.pvd6nto02timeqnrp8675309.canarytokens.com

PS C:\Users\casey>
```

```
PS C:\Users\casey\Desktop> reg import .\canarytoken.reg
The operation completed successfully.
PS C:\Users\casey\Desktop>
```

```
PS C:\Users\casey\Desktop> nltest.exe /?
Usage: nltest [/OPTIONS]


    /SERVER:<ServerName> - Specify <ServerName>

    /QUERY - Query <ServerName> netlogon service
    /REPL - Force partial sync on <ServerName> BDC
    /SYNC - Force full sync on <ServerName> BDC
    /PDC_REPL - Force UAS change message from <ServerName> PDC

    /SC_QUERY:<DomainName> - Query secure channel for <Domain> on <ServerName>
    /SC_RESET:<DomainName>[\<DcName>] - Reset secure channel for <Domain> on <ServerName> to <DcName>
    /SC_VERIFY:<DomainName> - Verify secure channel for <Domain> on <ServerName>
    /SC_CHANGE_PWD:<DomainName> - Change a secure channel  password for <Domain> on <ServerName>
    /DCLIST:<DomainName> - Get list of DC's for <DomainName>
    /DCNAME:<DomainName> - Get the PDC name for <DomainName>
    /DSGETDC:<DomainName> - Call DsGetDcName /PDC /DS /DSP /GC /KDC
        /TIMESERV /GTIMESERV /WS /NETBIOS /DNS /IP /FORCE /WRITABLE /AVOIDSELF /LDAPONLY /BACKG /DS_6 /DS_8 /DS_9 /DS_10
        /KEYLIST /TRY_NEXT_CLOSEST_SITE /SITE:<SiteName> /ACCOUNT:<AccountName> /RET_DNS /RET_NETBIOS
    /DNSGETDC:<DomainName> - Call DsGetDcOpen/Next/Close /PDC /GC
        /KDC /WRITABLE /LDAPONLY /FORCE /SITESPEC
    /DSGETFTI:<DomainName> - Call DsGetForestTrustInformation
        /UPDATE_TDO
```

Run nltest & alert us behind the scene, quick and easy :)

```
CASEYBCD8.UN.casey.CMD.pvd6nto02timeqnrp8675309.canarytokens.com
```

Canarytoken triggered

ALERT

A DNS Canarytoken has been triggered by the Source IP 192.168.16.1. Please note that the source IP refers to a DNS server, rather than the host that triggered the token.

Basic Details:

Channel	DNS
Time	2022-12-30 16:37:53 (UTC)
Canarytoken	0mpmy260rq80sil1uue12345
Token Reminder	Create Token to monitor for nltest.exe at branch offices. This should be a rare command (This token was created to monitor the execution of: nltest.exe)
Token Type	cmd
Source IP	192.168.16.1
Sensitive Command Information	User casey executed "nltest.exe" on the host caseyce69

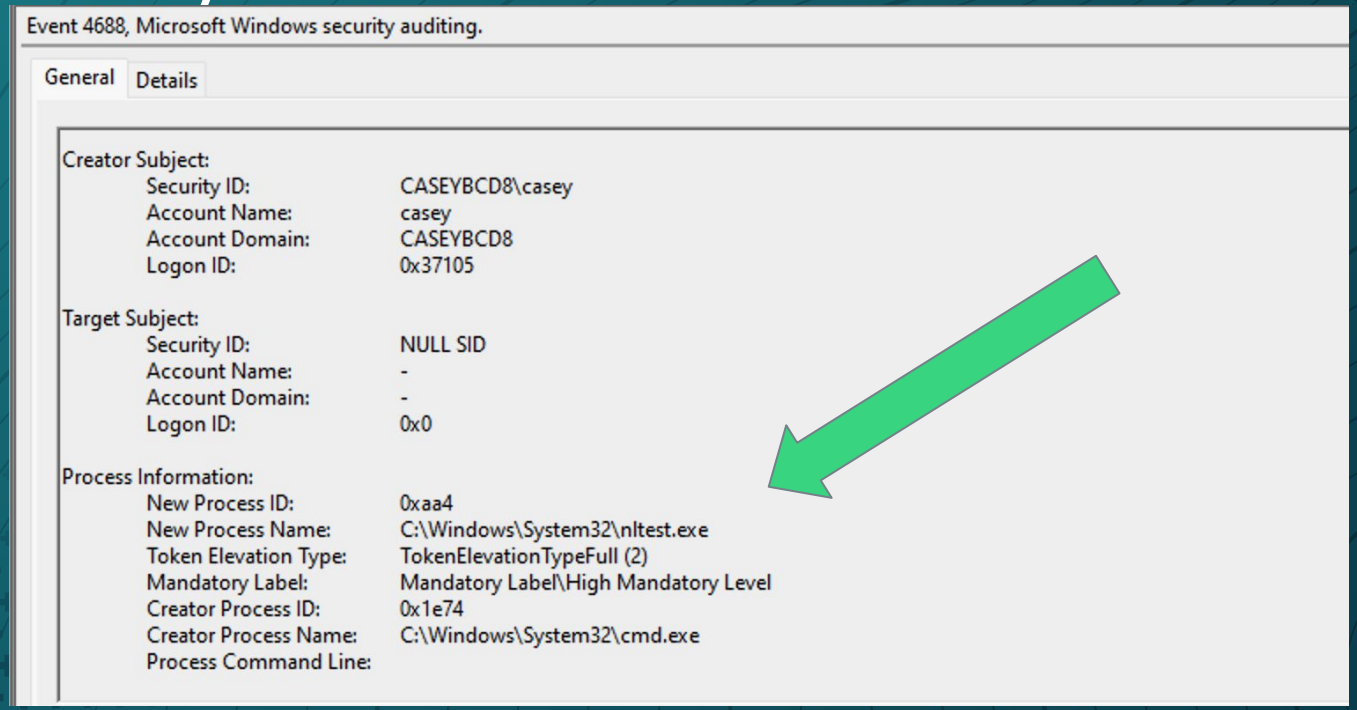
Canarytoken Management Details:

Date: 2023 Jan 04 16:50:10.019649 (UTC) **IP:** 192.168.32.1 **Channel:** DNS

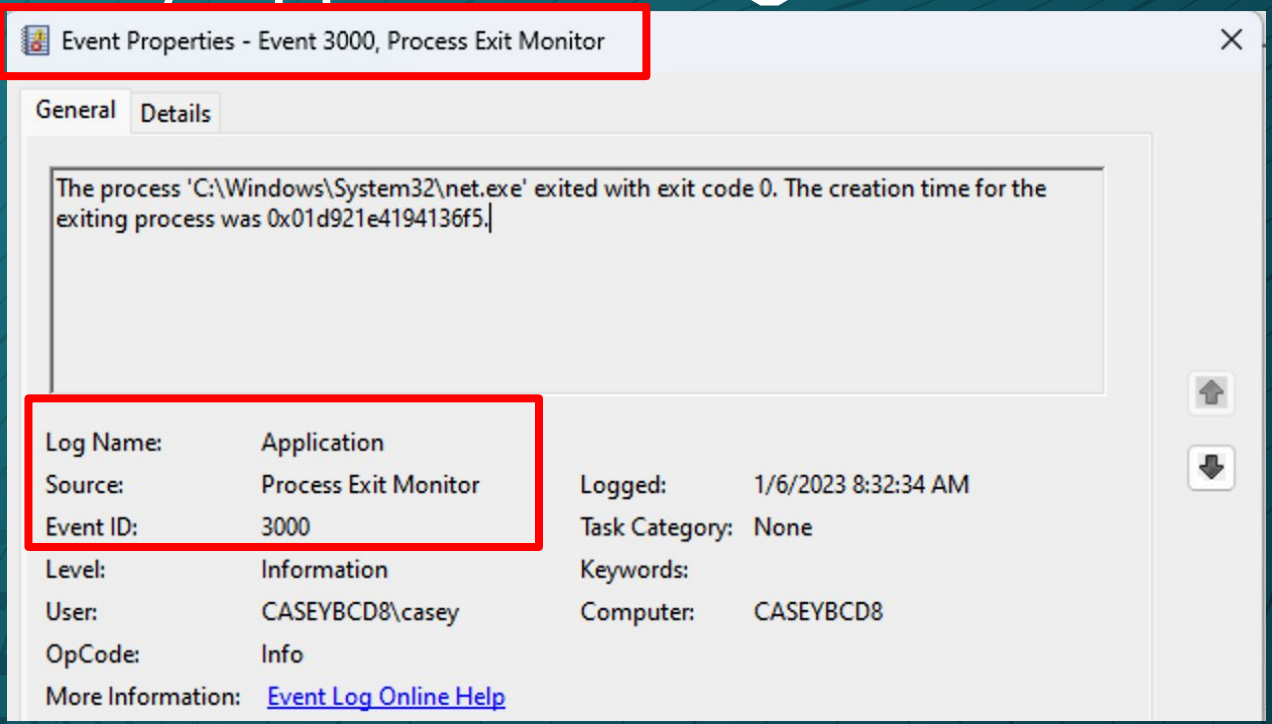
Tor	
Known Exit Node	False

Basic Info	
Memo	nltest.exe example for Cactus Con (This token was created to monitor the execution of: nltest.exe)
User executing command	casey
Computer executing command	caseybcd8

Verify - Windows 4688 Events



Verify Application Log Event 3000



Windows Error Reporting (WER)

This token is dependent on WER being enabled.
Confirm status

WindowsErrorReporting

Disable-WindowsErrorReporting	Disables Windows Error Reporting.
Enable-WindowsErrorReporting	Enables Windows Error Reporting.
Get-WindowsErrorReporting	Retrieves the Windows Error Reporting status.

Requirements

Administrator to install

Any user can run the command.

You can reuse the .reg file on multiple machines.

Any execution of the command creates an alert and sends username / hostname over DNS.

Use Cases - Candidate EXE's

Short run commands

Rare

Binary doesn't need to be present.

- Watch for psexec.exe or example if your organization never uses it.
- System you cannot install software on, but **can** change a registry key. 3rd Party systems.

Use Cases - meta

- Same reg file will work across entire organization (DNS will provide computer and user info)
  - Push out to org for simple visibility into e.g., mimikatz.exe invocation
- Lightweight/built-in option for legacy/non-standard systems that don't run baseline EDR, etc.

Enhancements

Filter Decisions - whoami.exe /all

```
Windows Registry Editor Version 5.00
; Sensitive command token generated by Thinkst Canary
; Run with admin privs on Windows machine as: reg import FILENAME

; command that will be watched for
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\ipconfig.exe]
"GlobalFlag"=dword:00000200

; magic unique canarytoken that will be fired when this command is executed
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SilentProcessExit\ipconfig.exe]
"ReportingMode"=dword:00000001
"MonitorProcess"="cmd.exe /c powershell.exe -Command \"$a=(Get-WmiObject win32_process -Filter 'ProcessID = %e' | select CommandLine); if ($a.CommandLine -match '/all') { $($u=$(\\\'
```

Live Demo!

Closing Thoughts

We can repurpose a malware/offensive technique to create a simple way for teams to get an alert when a executable fires.

One registry key , deployed to all machines, capture user and computer dynamically at run time.

Give it a test? Give us feedback?

Thanks to the Thinkst Team! 💚 ✊

We landed this as a team

Idea to Prototype, to Production, to Support







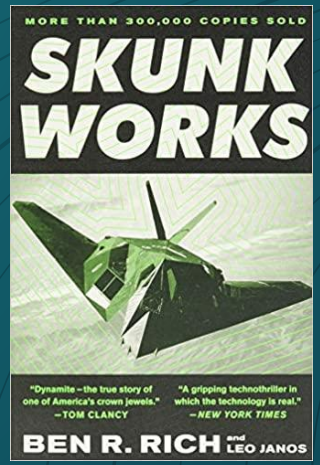

Closing Thoughts and Resources

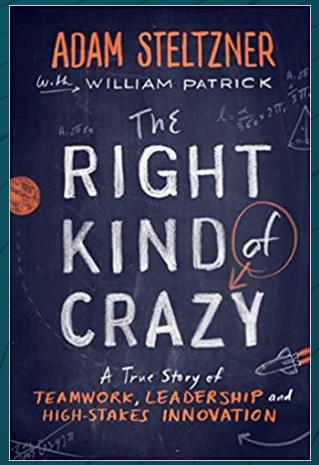
https://blog.thinkst.com/2022/09/sensitive-command-token-so-much-offense.html

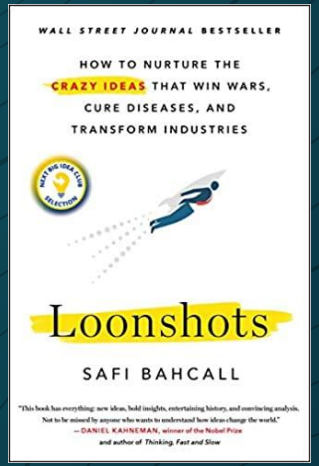
https://canarytokens.org

https://thinkst.com/

Books we love on Research & Engineering







Check it out, try it!
https://canarytokens.org/generate