OPSEC Fundamentals for Remote Red Teams

BLACK HILLS
Information Security
• 2008 •

Who am I?

- Michael Allen
    - Go by Wh1t3Rh1n0 online
    - Professional penetration tester/red teamer since 2014
    - Security Analyst at Black Hills Information Security since 2019
    - Certifications: OSCE, MLSE, CISSP, ...
- No formal training in OPSEC.
- Still make lots of mistakes. Just sharing some lessons I've learned.

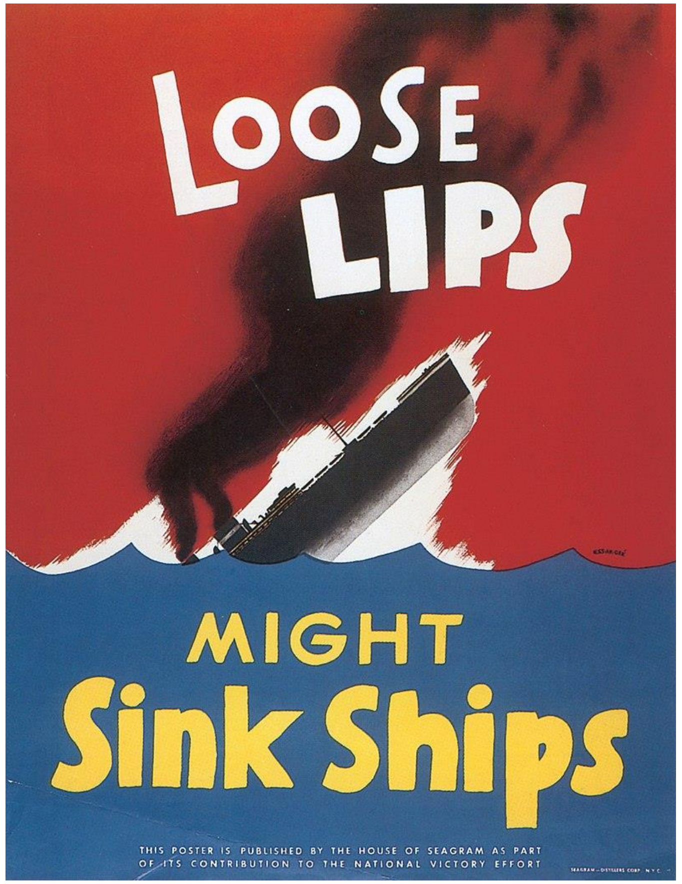
What is this talk about?

- OPSEC Fundamentals for Remote Red Teams

What is this talk about?

- **OPSEC** Fundamentals for Remote Red Teams
- "Operations security (OPSEC) is a process that identifies critical information to determine if friendly actions can be observed by enemy intelligence, determines if information obtained by adversaries could be interpreted to be useful to them, and then executes selected measures that eliminate or reduce adversary exploitation of friendly critical information."

  -- Wikipedia (highlighting added)


LOOSE
LIPS
MIGHT
Sink Ships
THIS POSTER IS PUBLISHED BY THE HOUSE OF SEAGRAM AS PART OF ITS CONTRIBUTION TO THE NATIONAL VICTORY EFFORT
SEAGRAM-DISTILLERS CORP. N.Y.C.

What is this talk about?

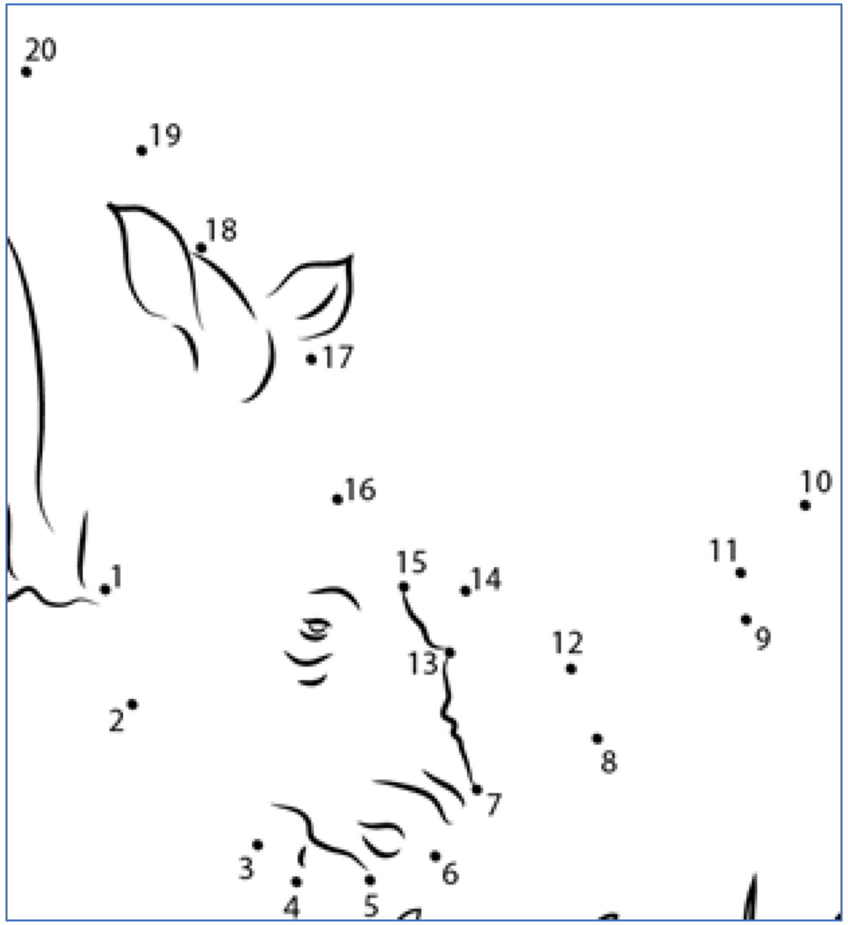
- OPSEC Fundamentals for **Remote Red Teams**
- Remote Red Team Exercises
  - Perform a cyberattack against the target organization.
  - Attack success demonstrates business impact.
  - Unannounced - The security team (Blue Team) of the target organization **actively** identifies and responds to suspicious activity - just like a real attack.
  - Physical and wireless security not in scope.
- This talks focuses on red team OPSEC **up to the initial breach**.
  - Spin-off of content in the "*Red Team: Getting Access*" class.
  - Additional OPSEC considerations introduced after gaining access.

Why is OPSEC important for red teams?

- Indicators of suspicious activity are well known
  - Open sharing of information is how InfoSec works
- Modern blue teams have many sources to inform them of suspicious activity:
  - Logs on Internet-facing systems/services
  - Logging on systems and connections that make egress from the internal network (e.g. workstations)
  - Various of third-party "threat intelligence" services
  - Analysis and correllation of information collected from those sources

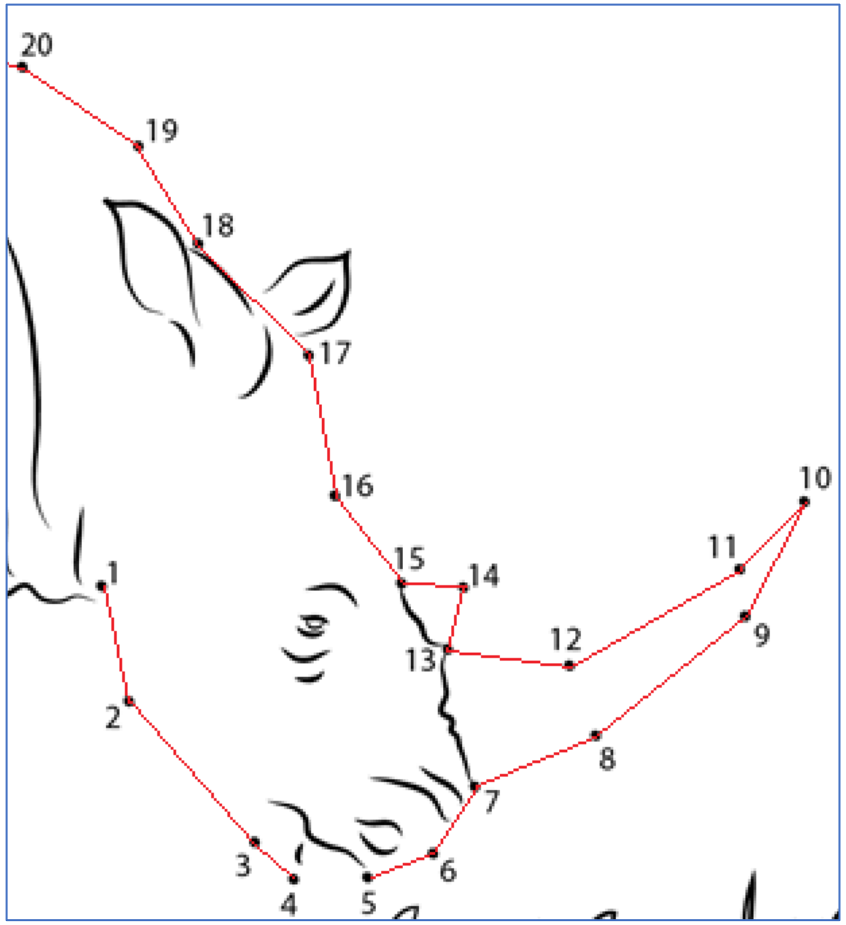
Red doesn't want blue to connect the dots

- Each time the red team interacts with the target organization, it leaves a "dot".



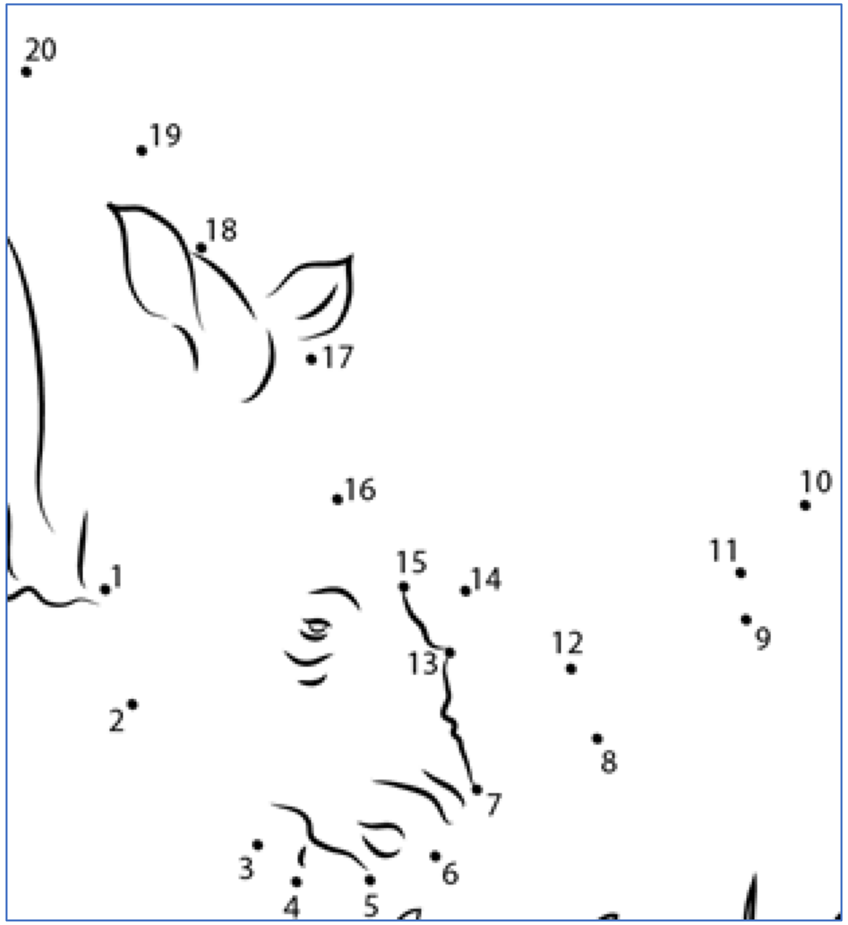
Red doesn't want blue to connect the dots

- Each time the red team interacts with the target organization, it leaves a "dot".
- With enough dots, the blue team can identify red team tools and infrastructure, and prevent attacks before they occur.
- This is especially frustrating if the blue team connects the dots before the red team knows it.



Possible countermeasures

How can the red team keep the blue team from connecting the dots?



Possible countermeasures

How can the red team keep the blue team from connecting the dots?

1. Don't leave any dots.

Possible countermeasures

How can the red team keep the blue team from connecting the dots?

1. Don't leave any dots.
2. When leaving a dot is unavoidable, don't leave clues that associate it with other dots.

Possible countermeasures

How can the red team keep the blue team from connecting the dots?

1. Don't leave any dots.
2. When leaving a dot is unavoidable, don't leave clues that associate it with other dots.
3. Create dots that associate other dots with unrelated dots.

Possible countermeasures

How can the red team keep the blue team from connecting the dots?

1. ~~Don't leave any dots.~~
2. When leaving a dot is unavoidable, don't leave clues that associate it with other dots.
3. ~~Create dots that associate other dots with unrelated dots.~~

Other threats to the red team

- Data leaks
    - Potentially damaging to the company reputation
    - May reveal the identity of the red team organization
    - May reveal the identity of the red team's customers
- Real-world threat actors
    - Network and application security vulnerabilities

Steps for assessing red team actions

1. Plan likely actions
2. Brainstorm information disclosed by each action
3. Assess whether the disclosed information can be used against the red team
   - What information is disclosed?
   - Who can observe the disclosed information?
   - Is it likely for the information be used against the red team?
4. Adjust plans to mitigate the risks of information disclosure

Build OPSEC into your standard procedures

- Document and apply OPSEC in advance to common actions (Standard Operating Procedures)
  - Setup (Local VMs, C2 servers, domains, ...)
  - Reconnaissance (Search queries, port scans, browsing target's web site, ...)
  - Attacks (Password guessing, phishing, ...)
- Apply the OPSEC assessment process before using new tools/techniques
  - Setup a test environment to mimic the target
  - Observe the action from the target's point of view

My Standard Operating Procedures for Red Team OPSEC

1. Local workstation setup
2. Source IP addresses
3. Other third-party services
4. Network services
5. Testing new tools

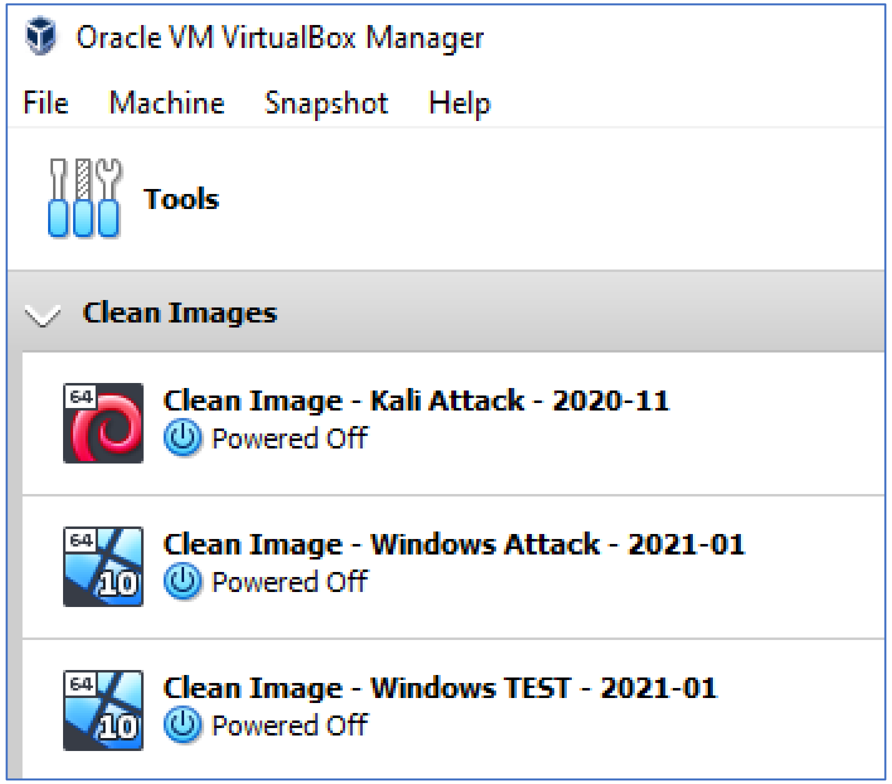
1. Local workstation setup

Use Virtual Machines

- Quick deployment:
  - System configured for OPSEC best practices
  - Tools - installed, customized, and configured
  - Clean environment - no artifacts left over from other customers
  - Additional VMs when necessary
- Images can be updated/modified without a full rebuild
- Keep a checklist for your build and configuration process for each OS
  - Update it as you learn more
- Some protection against accidental compromise (downloaded hacking tools)

Use Virtual Machines

- Kali Linux Attack VM
- Windows Attack VM
  - All antivirus, firewall, and other defenses disabled.
  - Browser configurations adjusted for OPSEC.
  - Payload development and execution testing.
- Windows Test VM
  - Stock - Latest Windows build. No extra libraries.
  - All defenses left intact. Additional defenses installed to mimic the target.
  - Payload portability and AV testing. (Revert snapshot after.)
  - Used for authenticated connections to the target network.



Clean Image - Kali Attack - 2020-11
Powered Off

Clean Image - Windows Attack - 2021-01
Powered Off

Clean Image - Windows TEST - 2021-01
Powered Off

Operating System modifications

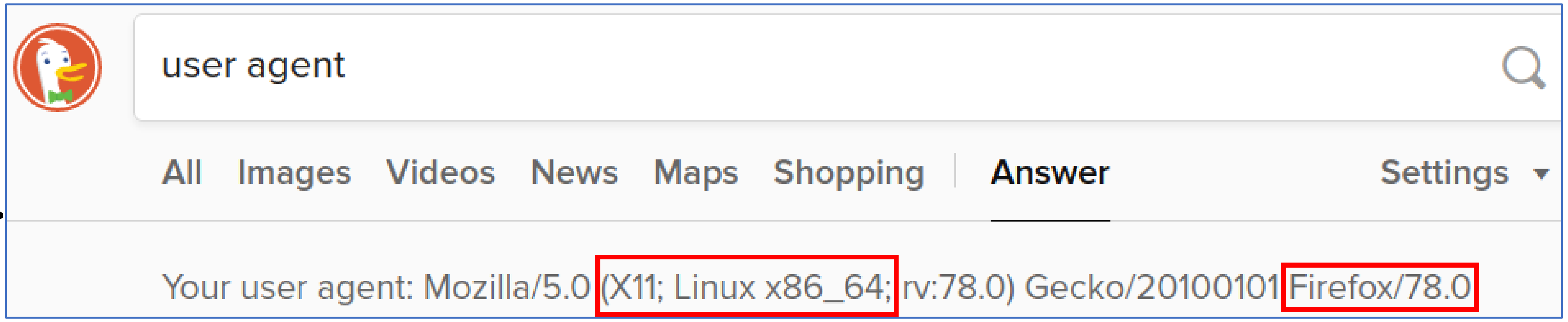
- Set a strong password (no kali:kali)
- Change the hostname & local user name
  - Remote: localhost, DESKTOP, PC / On-site: PRINTER, deskjet, LEXMARK
  - After recon: Match real/similar values used by the target
- Why change the hostname and username?
  - Many opportunities to leak your local host or user name to the target
    - NTLM authentication requests from websites/network services
    - "Honeypot" documents that phone home
    - Metadata in PDF/Office document payloads
  - **Prevention against others that may be unknown**

Why change the domain name?

- VPN clients commonly report criteria about the system to the VPN server that causes the connection to be allowed or denied:
  - Domain name
  - Host name
  - **Domain** user name
  - Operating system version and patch level
  - Whether expected EDR/AV products are installed
- It's trivial for blue teams to alert on VPN connections from systems not joined to the expected domain name.
- Also useful when testing payloads keyed to the target's domain name.

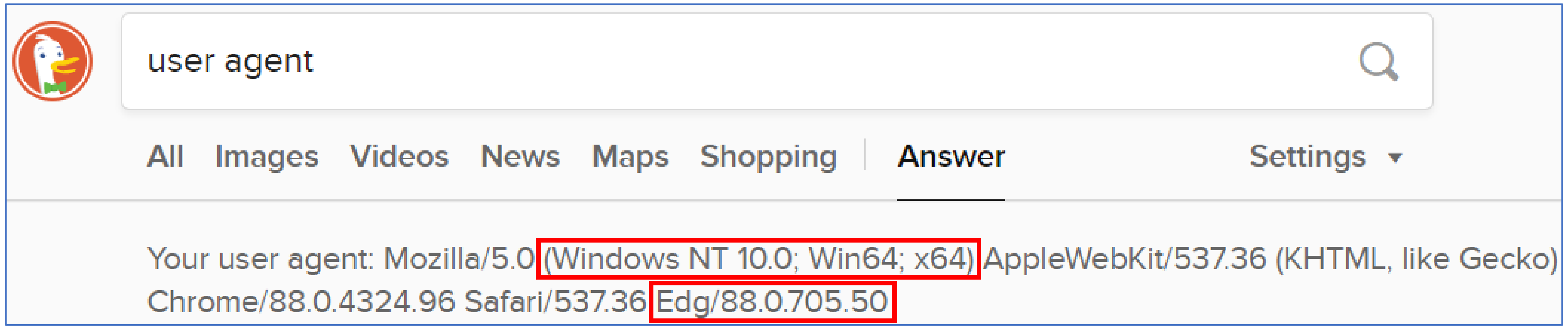
Tool configurations

- "User-Agent" header - Tells web servers:
  - Browser/client software and version number
  - OS version
- Kali Linux ----->



Your user agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0

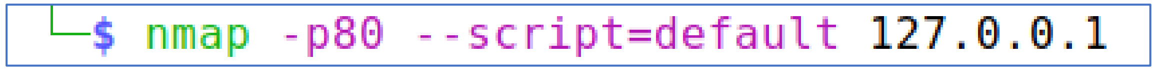
- Edge ------->



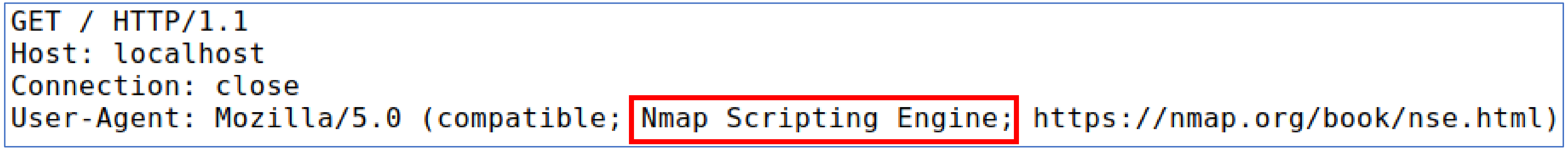
Your user agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.96 Safari/537.36 Edg/88.0.705.50

User-Agent headers

- Nmap default User-Agent:

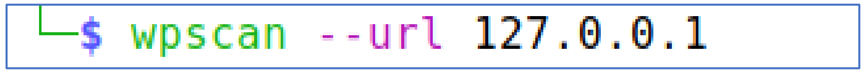


```
└─$ nmap -p80 --script=default 127.0.0.1
```

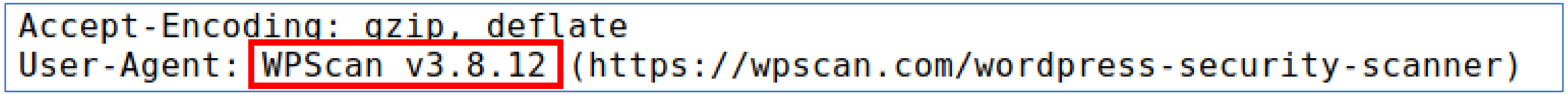


```
GET / HTTP/1.1
Host: localhost
Connection: close
User-Agent: Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)
```

- WPScan default User-Agent:



```
└─$ wpscan --url 127.0.0.1
```



```
Accept-Encoding: gzip, deflate
User-Agent: WPScan v3.8.12 (https://wpscan.com/wordpress-security-scanner)
```

Changing the default User-Agent

- Pick a user agent that isn't the default value for a hacking tool
  - Consider matching with the type and/or amount of traffic the site receives
  - You may not want to use the same user agent string for every tool
- Examples:
  - Google Chrome on Windows 10

```
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/87.0.4280.88 Safari/537.36
```

  - Googlebot - Web crawler of the Google search engine

```
Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
```

Changing the default User-Agent

- Command-line tools - Fail-safe command aliases
  - Add to ~/.bashrc or ~/.zshrc:

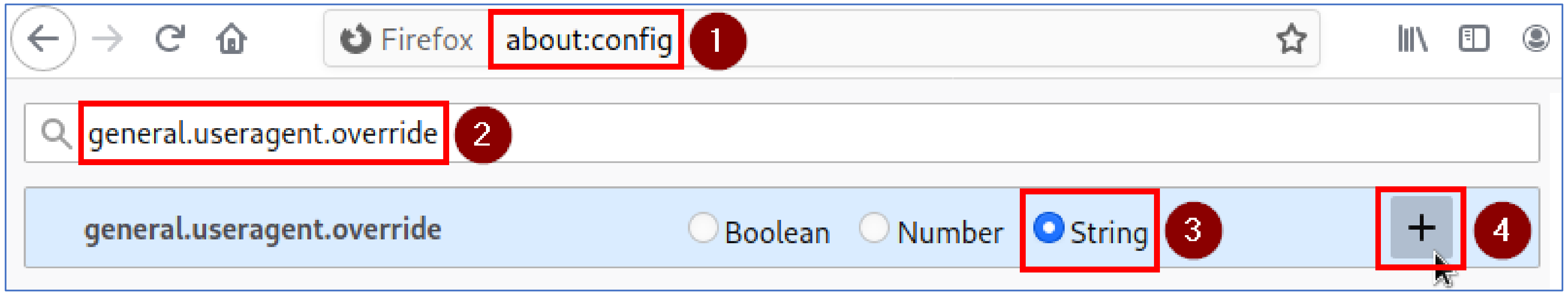
```
export AGENT="Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/87.0.4280.88 Safari/537.36"

alias curl="curl -A '$AGENT'"

alias wget="wget -U '$AGENT'"

alias nmap="nmap --script-args=\"http.useragent='$AGENT'\""
```

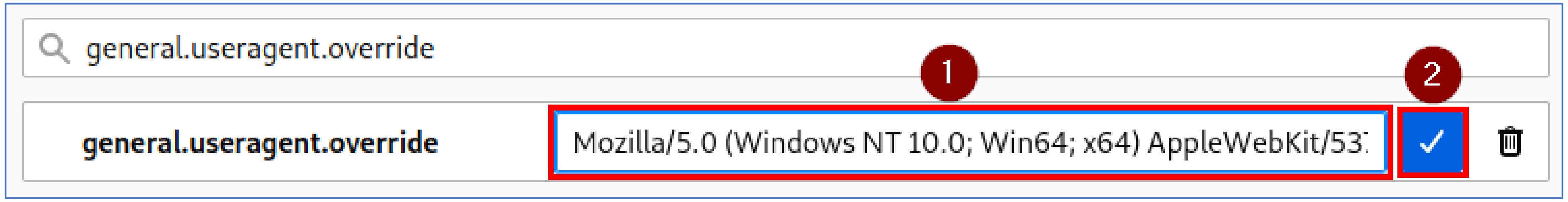
- $AGENT environment variable use with other commands:

```
wpscan --ua "$AGENT" --url 127.0.0.1
```

Changing the default User-Agent

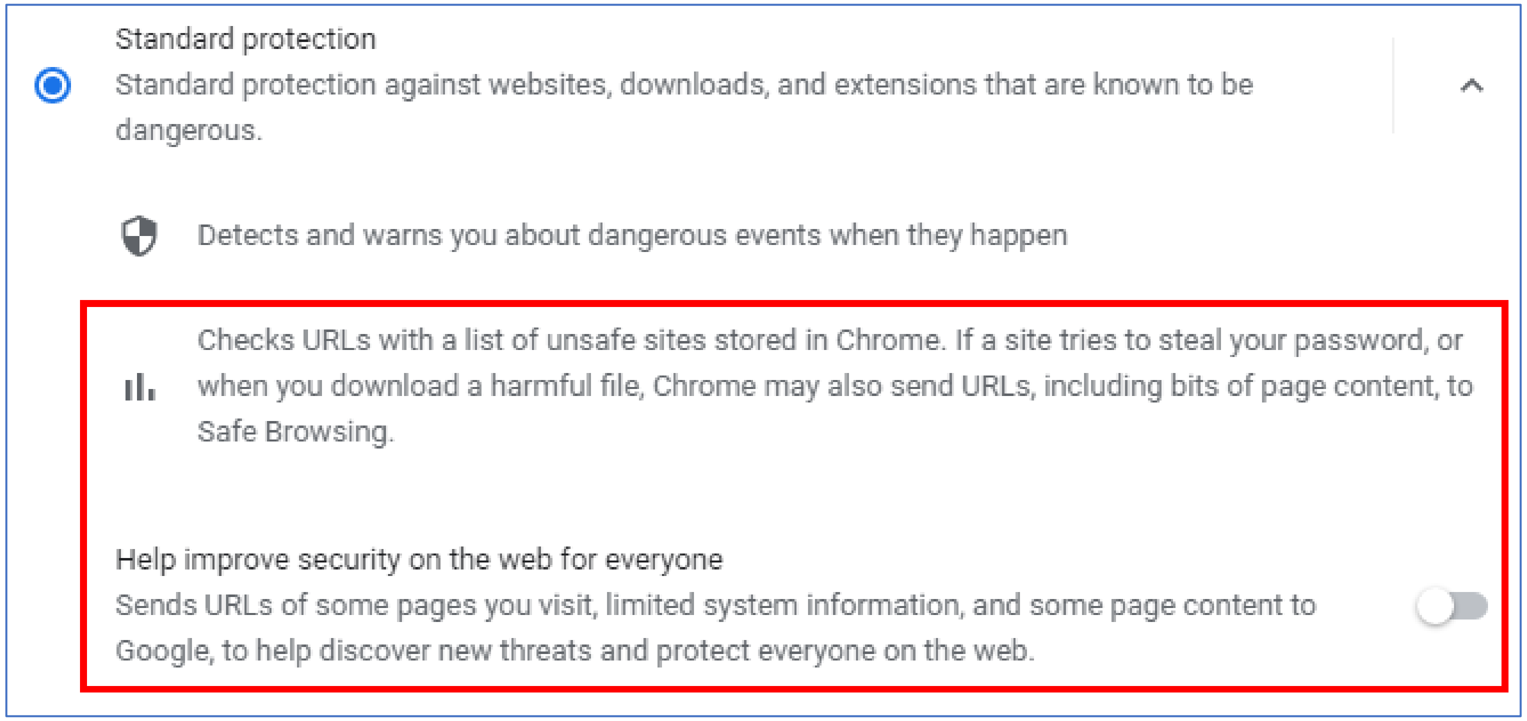
- In Firefox:
  - Use an extension like "User-Agent Switcher and Manager"
  - Or set the "general.useragent.override" string in "about:config"




general.useragent.override
1
2
general.useragent.override
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/53

Watch out for browser leaks

- Examine browser settings that send data to third-parties



Standard protection

Standard protection against websites, downloads, and extensions that are known to be dangerous.

Detects and warns you about dangerous events when they happen

Checks URLs with a list of unsafe sites stored in Chrome. If a site tries to steal your password, or when you download a harmful file, Chrome may also send URLs, including bits of page content, to Safe Browsing.

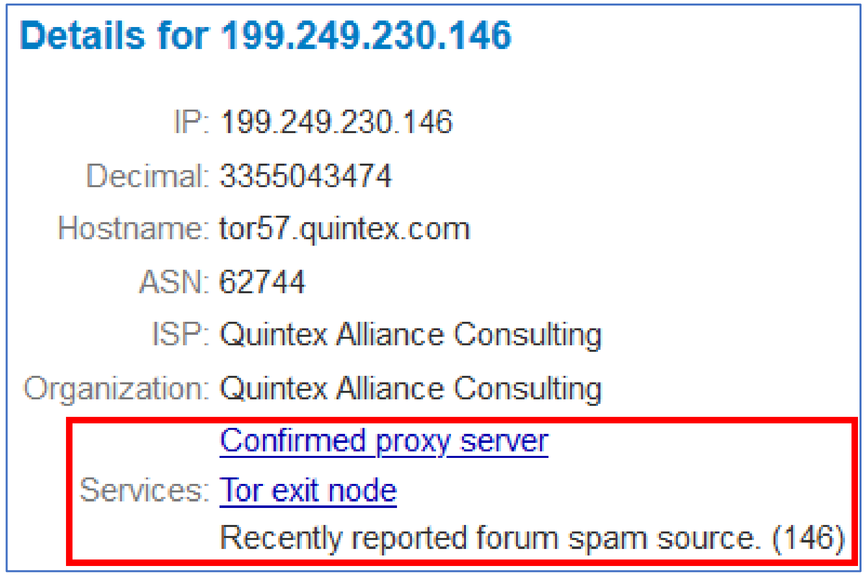
Help improve security on the web for everyone

Sends URLs of some pages you visit, limited system information, and some page content to Google, to help discover new threats and protect everyone on the web.

2. Source IP addresses

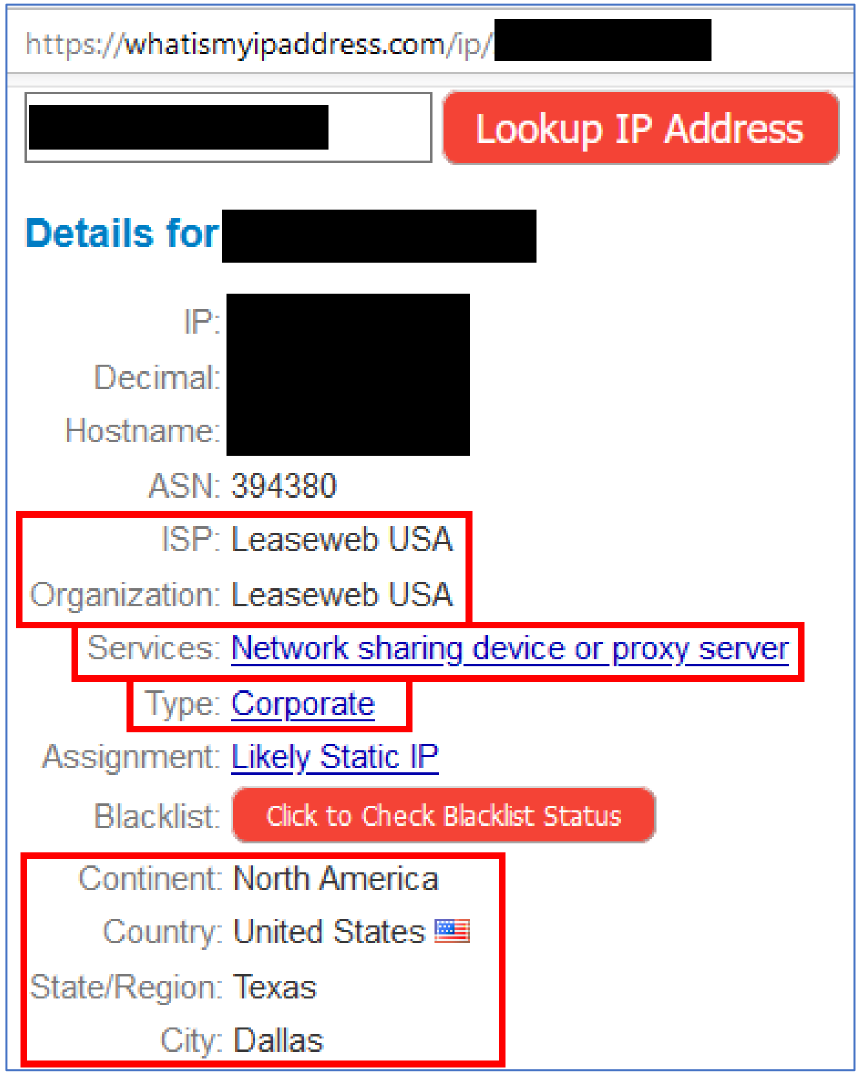
What might make an IP address suspicious?

1. Association with other suspicious traffic
2. Physical location
3. Service provider and Type of connection
   - Residential/Commercial
   - Mobile
   - Cloud
   - VPN
   - Proxies
   - TOR



Details for 199.249.230.146

IP:	199.249.230.146
Decimal:	3355043474
Hostname:	tor57.quintex.com
ASN:	62744
ISP:	Quintex Alliance Consulting
Organization:	Quintex Alliance Consulting
	Confirmed proxy server
Services:	Tor exit node
	Recently reported forum spam source. (146)



https://whatismyipaddress.com/ip/[redacted]

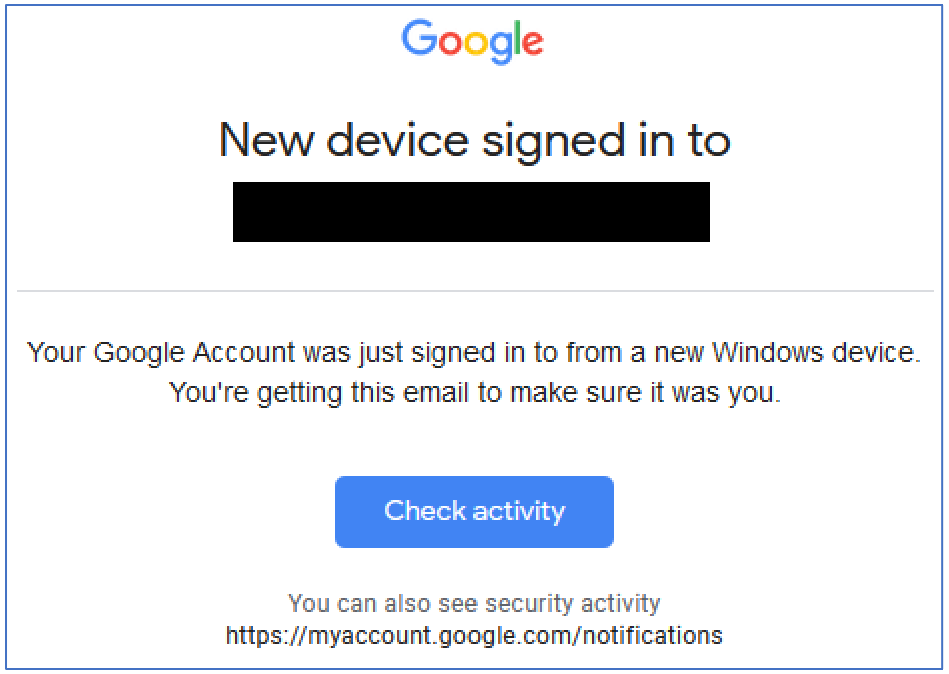
[redacted]

Details for [redacted]

IP:	[redacted]
Decimal:	
Hostname:	
ASN:	394380
ISP:	Leaseweb USA
Organization:	Leaseweb USA
Services:	Network sharing device or proxy server
Type:	Corporate
Assignment:	Likely Static IP
Blacklist:	Click to Check Blacklist Status
Continent:	North America
Country:	United States
State/Region:	Texas
City:	Dallas

Countermeasures for source IP addresses

- Never use the same IP address for two activities that you don't want associated with each other
- IP addresses should make sense relative to your actions from the IP
- Logging into user accounts:
  - One IP address per user account - and keep using the same IP for all subsequent logins
  - Login from an IP address in the same region as the user (common web mail alert)
  - Login from a service provider that makes sense for the target user. Probably not a VPS
  - Avoid known-suspicious IPs: TOR, proxies



New device signed in to

[redacted]

Your Google Account was just signed in to from a new Windows device. You're getting this email to make sure it was you.

You can also see security activity
https://myaccount.google.com/notifications

Make your VPN connection fail safe

- If your VPN connection drops unexpectedly, subsequent connections may come from your real IP address.
- Not ideal in the middle of suspicious activity (e.g. password spraying)

```
openvpn --script-security 2 --down vpn-down.sh --config <OPEN VPN FILE>
```

```
#!/bin/bash
echo 'Disabling network interfaces...'
systemctl stop network-manager
killall -9 dhclient
for i in $(ifconfig | grep -iEo '^[a-z0-9]+:' | grep -v '^lo:$' | cut -d ':' -f 1)
do
    ifconfig $i 0.0.0.0 down
done
```

3. Other third-party services

Third-party services

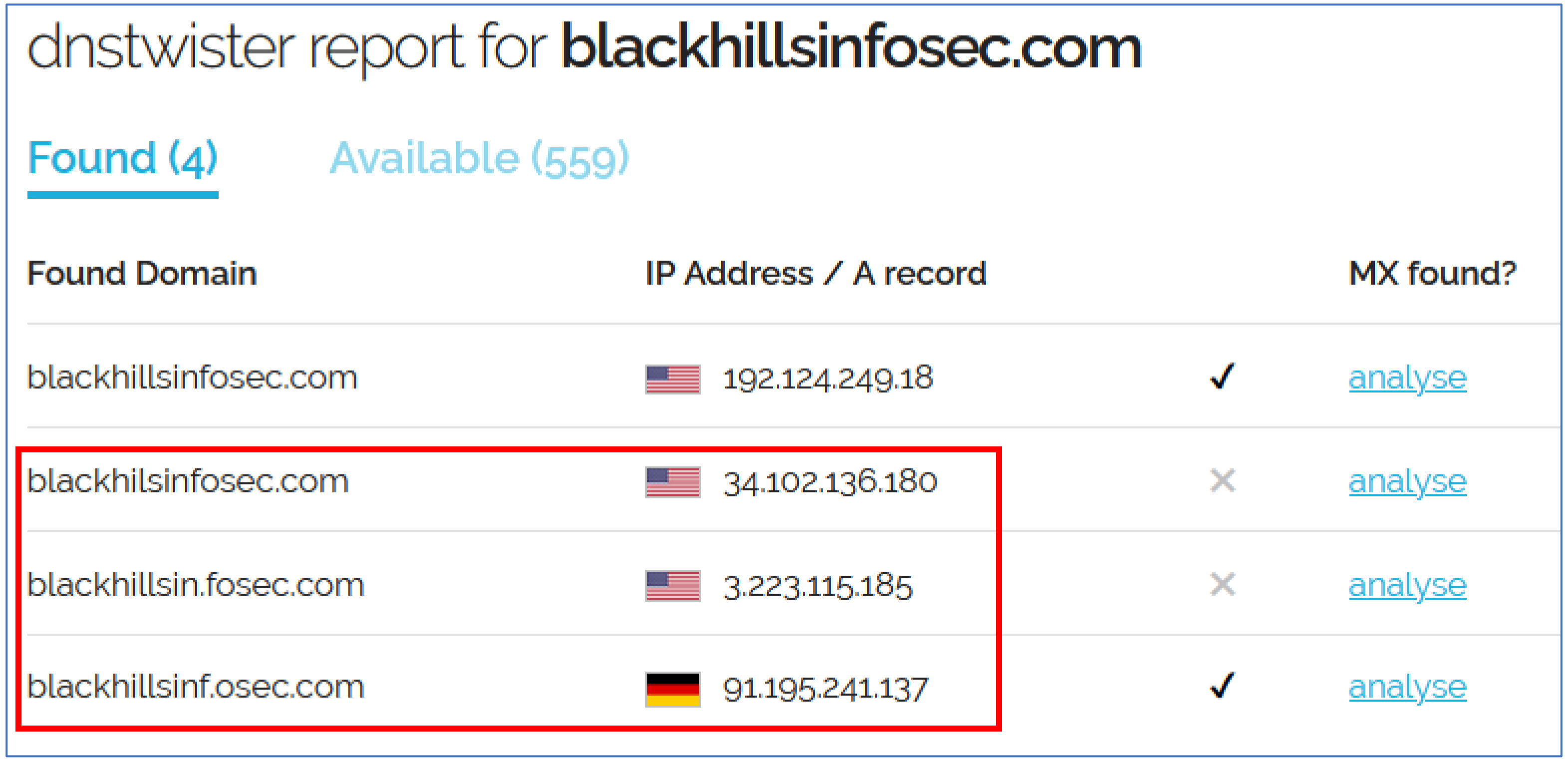
- Assess whether your registration information is likely to be exposed.
- Assess whether use of the same account across multiple projects is likely to leak information about the red team or its customers.
  - Can the resource owner (the red team) be identified by outsiders?
  - Can relationships between multiple resources be associated with each other?
  - Is there a reasonable level of trust with the resource provider?
- Identify additional areas of concern.

Domain names

- Use private registration (WHOIS privacy)
  - Make sure it is enabled by default on your account.
  - Make sure it is enabled at the time of registration.
- One red team action per domain name - same as source IPs
  - Domain A sends email
  - Domain B serves payload files
  - Domain C receives C2 callbacks
  - ...

Domain names

- Avoid typosquatting the target domain name or company name
- Subdomains may be detected too



dnstwister report for **blackhillsinfosec.com**

Found (4)   Available (559)

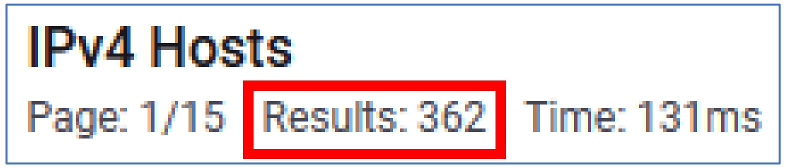
Found Domain	IP Address / A record		MX found?
blackhillsinfosec.com	192.124.249.18	✓	analyse
blackhilsinfosec.com	34.102.136.180	✗	analyse
blackhillsin.fosec.com	3.223.115.185	✗	analyse
blackhillsinf.osec.com	91.195.241.137	✓	analyse

SSL/TLS certificates

- Watch out for data leaks in certificate transparency logs
  - Customer names (e.g. in subdomain names)
  - Email addresses

```
certbot --apache --register-unsafely-without-email
```

- No default or self-signed certificates - Both easily flagged/blocked
  - Censys.io search for Cobalt Strike: 87f2085c32b6a2cc709b365f55873e207a9caa10bffecf2fd16d3cf9d94d390c
- Let's Encrypt certificates *might* be suspicious
  - Default CA in lots of hacking tutorials
  - Free
  - Used by legitimate websites too



IPv4 Hosts

Page: 1/15 Results: 362 Time: 131ms

4. Network services

Network services

- Don't expose services to the Internet that aren't required
  - Use SSH port forwarding for access from the red team
- Change all the default settings on hacking tools that listen on an Internet-facing port
- Use redirectors liberally and spread them across multiple CSPs
  - On the server: Whitelist redirector IP addresses and block all others
- Use common web services (Apache, Nginx) to redirect HTTP/S traffic to servers run by hacking tools

Example: Cobalt Strike Team Server Study*

- **Indicator:** TCP Port 50050 - Default Cobalt Strike Team Server port
  - *Don't expose services to the Internet that aren't required*
  - *Use SSH port forwarding for access from the red team*
- **Indicator:** Cobalt Strike DNS server responds to requests with IP 0.0.0.0
  - *Change all the default settings on hacking tools that listen on an Internet-facing port*
- **Indicator:** Web root: 404 Not Found, no content, Content-Type: text/plain
  - Default response without content explicitly hosted at / or a web redirector.
- **Indicator:** JA3S service fingerprinting
  - *Use common web services (Apache, Nginx) to redirect HTTP/S traffic to servers run by hacking tools*

* https://blog.cobaltstrike.com/2019/02/19/cobalt-strike-team-server-population-study/

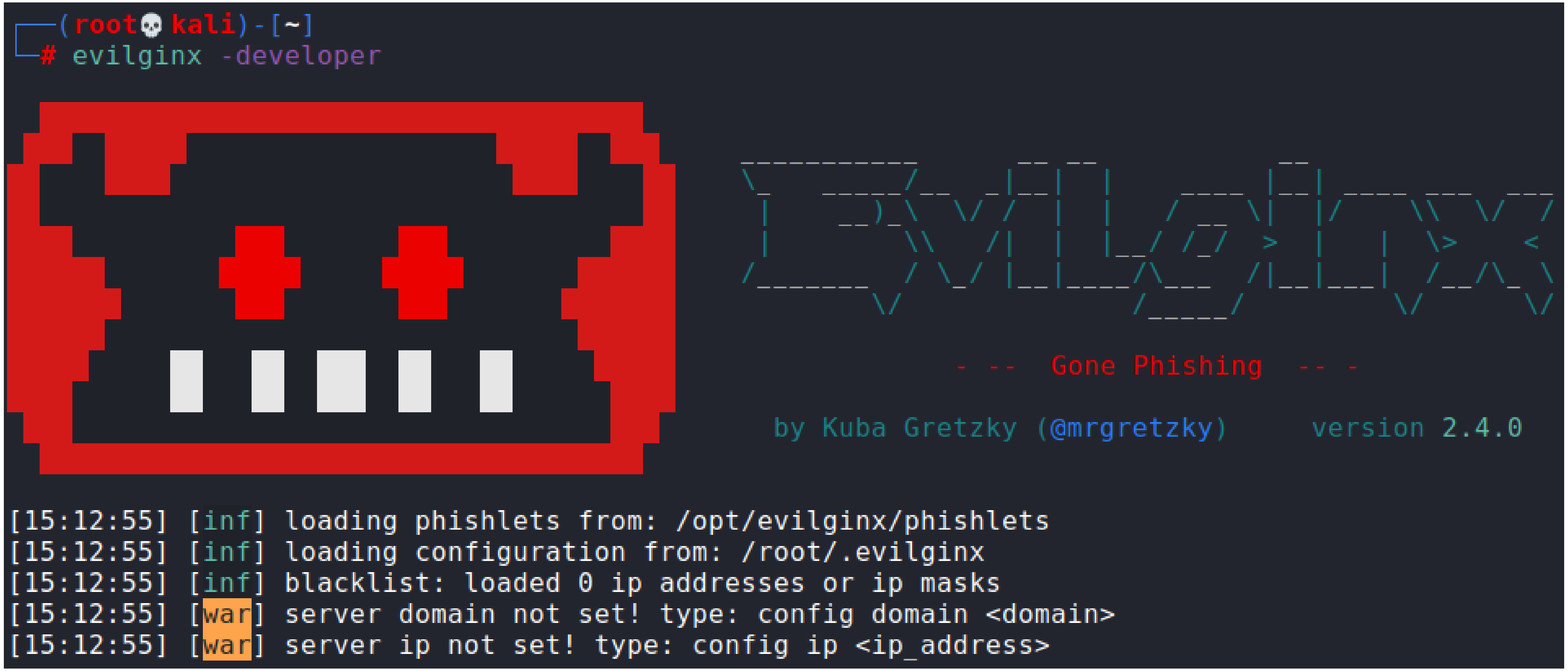
5. Testing new tools

Vetting new tools

- Goals:
  1. Confirm that the tool won't harm you or your customer.
  2. Identify and mitigate any "tells" that could give away your attack to the target.
  3. Confirm that tool does the thing it's supposed to do.

- Steps for vetting the tool:
  - Read the source code
  - Understand the configuration options
  - Run the tool in a VM
  - Observe the traffic the tool generates:
    - Packet capture - WireShark
    - Intercepting proxy - Burp Suite
    - Simulated target - Ncat/Netcat

Example: Evilginx

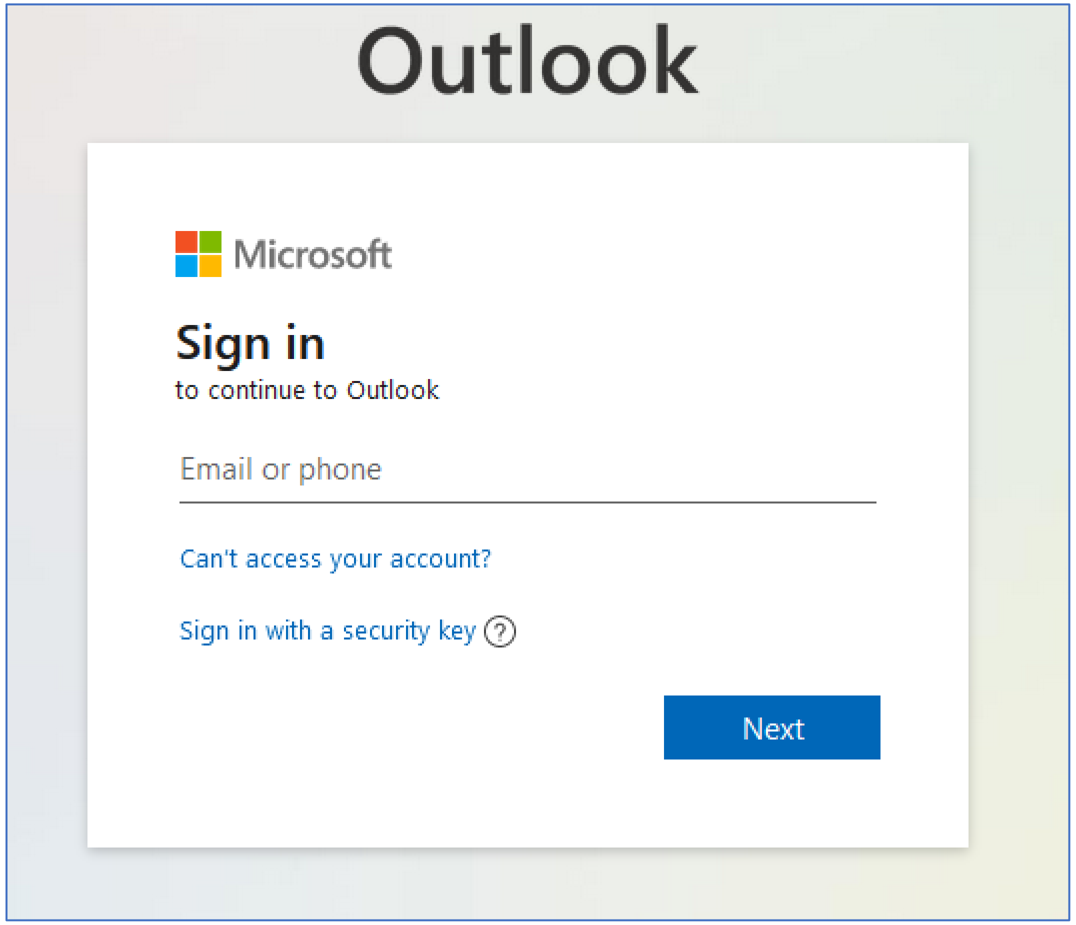


```
┌──(root💀kali)-[~]
└─# evilginx -developer
```

```
[15:12:55] [inf] loading phishlets from: /opt/evilginx/phishlets
[15:12:55] [inf] loading configuration from: /root/.evilginx
[15:12:55] [inf] blacklist: loaded 0 ip addresses or ip masks
[15:12:55] [war] server domain not set! type: config domain <domain>
[15:12:55] [war] server ip not set! type: config ip <ip_address>
```

Me, running Evilginx for the first time:

1. Download the latest **precompiled release** from GitHub
2. Configure a phishlet to target Office 365
3. Generate a "lure" (landing URL)
4. Visit the lure in the browser
5. Everything looks ok -->


Outlook

Microsoft

Sign in

to continue to Outlook

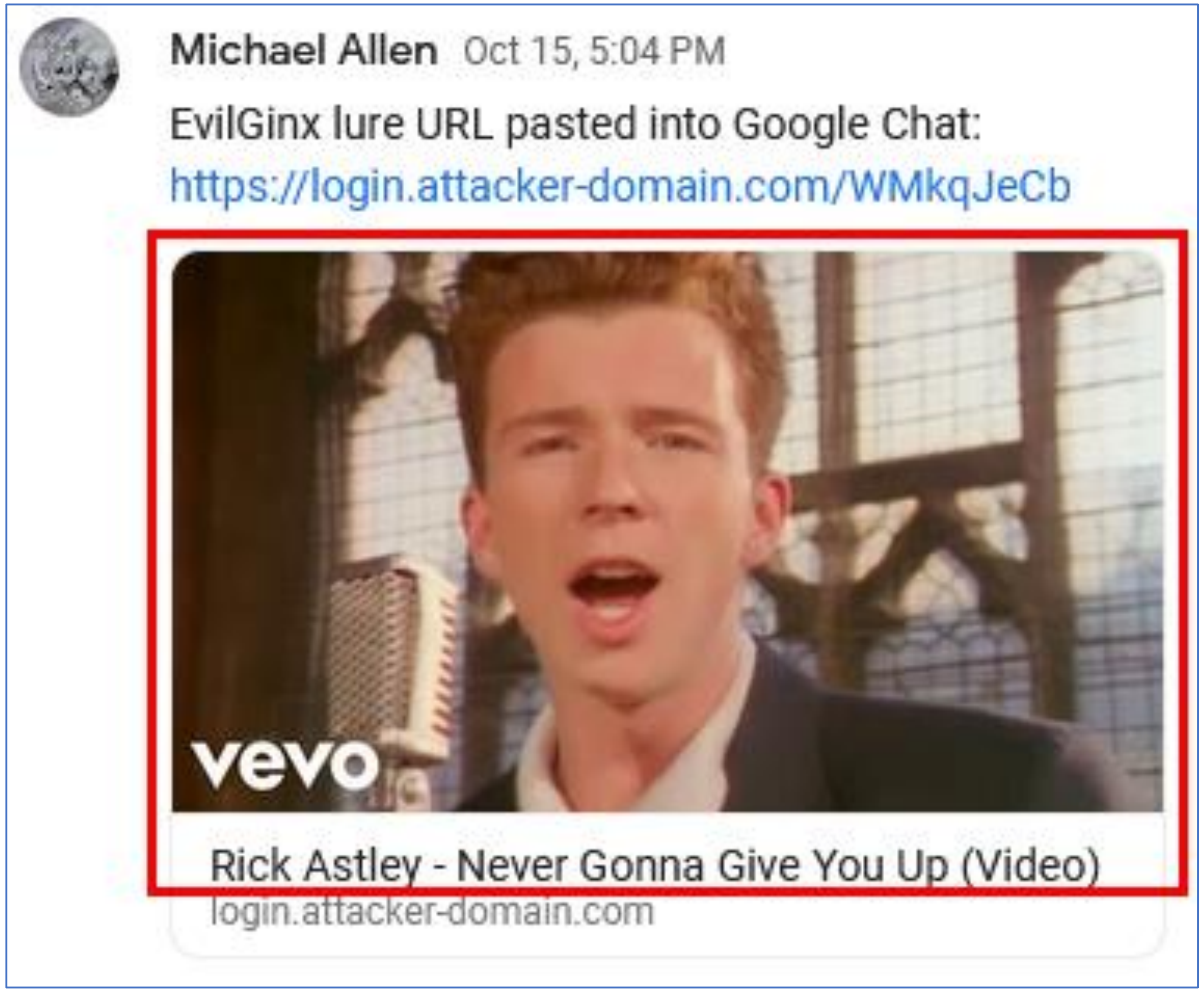
Email or phone

Can't access your account?

Sign in with a security key

Me, pasting the lure URL into Google Chat:

- Everything does **NOT** look ok



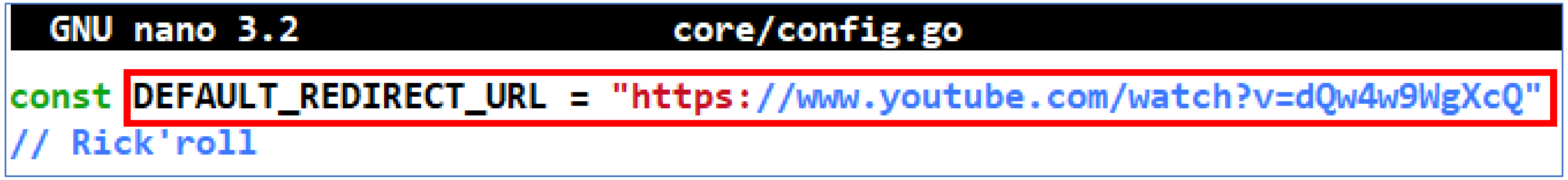
Michael Allen Oct 15, 5:04 PM

EvilGinx lure URL pasted into Google Chat:
https://login.attacker-domain.com/WMkqJeCb

Rick Astley - Never Gonna Give You Up (Video)
login.attacker-domain.com

Me, pasting the lure URL into Google Chat:

- Everything does **NOT** look ok



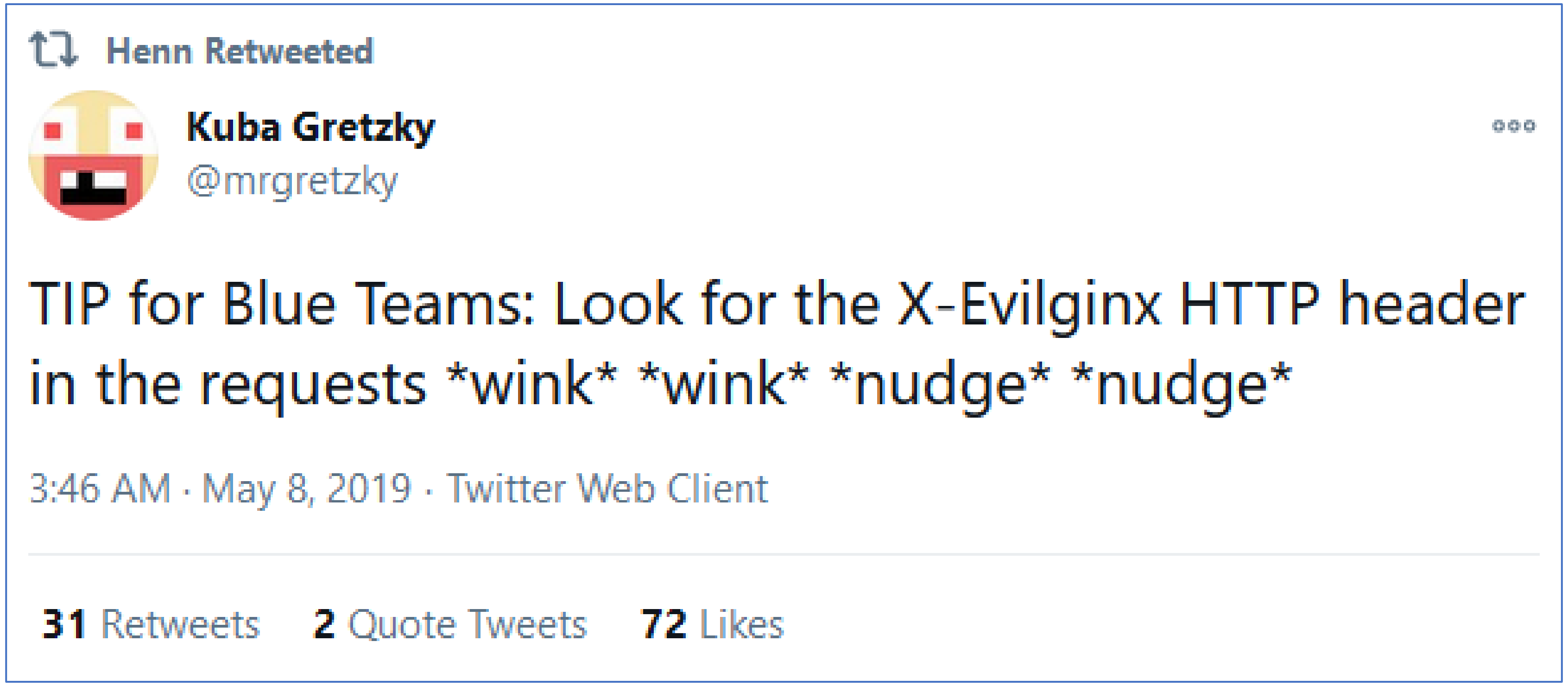
```
  GNU nano 3.2                    core/config.go

const DEFAULT_REDIRECT_URL = "https://www.youtube.com/watch?v=dQw4w9WgXcQ"
// Rick'roll
```

- Easy fix:

```
config redirect_url "https://www.office.com"
```

- Lesson learned:
  - **Read the source code**
  - **Understand the configuration options**

But wait, there's more...


Henn Retweeted

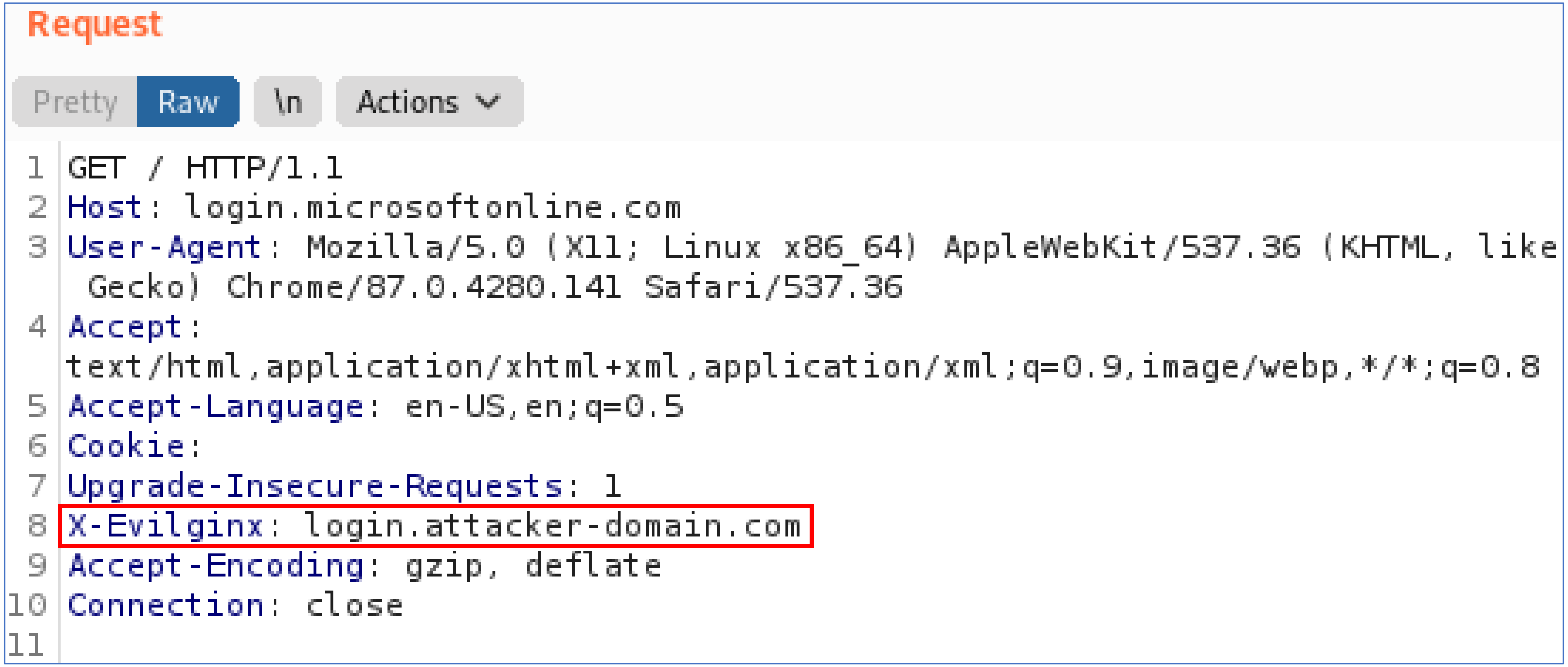
Kuba Gretzky
@mrgretzky

TIP for Blue Teams: Look for the X-Evilginx HTTP header in the requests *wink* *wink* *nudge* *nudge*

3:46 AM · May 8, 2019 · Twitter Web Client

31 Retweets **2** Quote Tweets **72** Likes

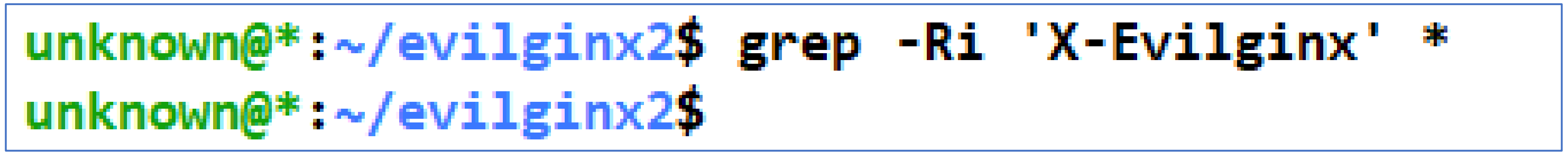
Inspecting Evilginx traffic with Burp:



```
GET / HTTP/1.1
Host: login.microsoftonline.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like
 Gecko) Chrome/87.0.4280.141 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Cookie:
Upgrade-Insecure-Requests: 1
X-Evilginx: login.attacker-domain.com
Accept-Encoding: gzip, deflate
Connection: close

```

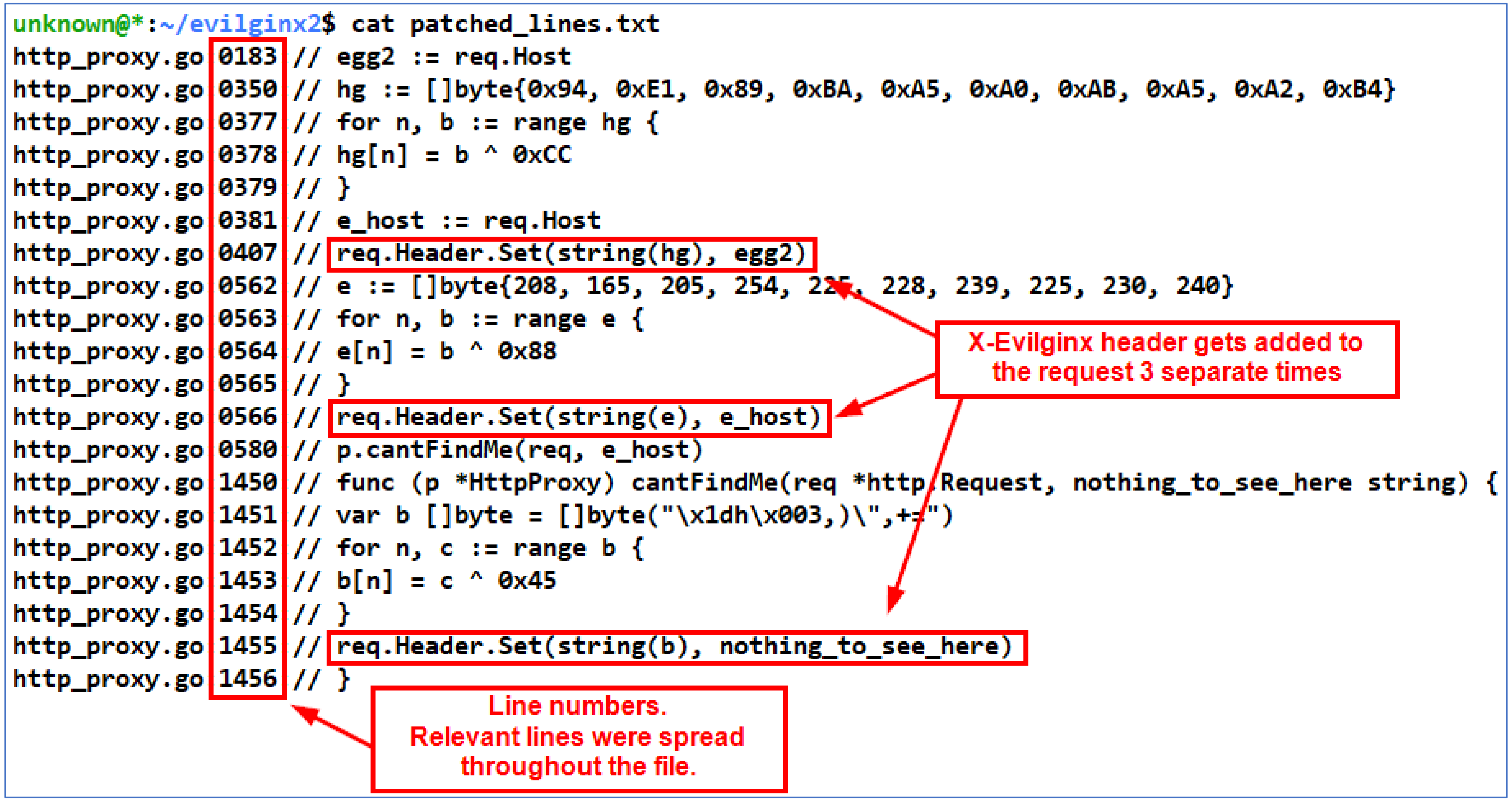
Removing the X-Evilginx header

- No settings found to change/disable the header
- No mention in the documentation
- No references found when searching the source-code



```
unknown@*:~/evilginx2$ grep -Ri 'X-Evilginx' *
unknown@*:~/evilginx2$
```

Looking more closelier...

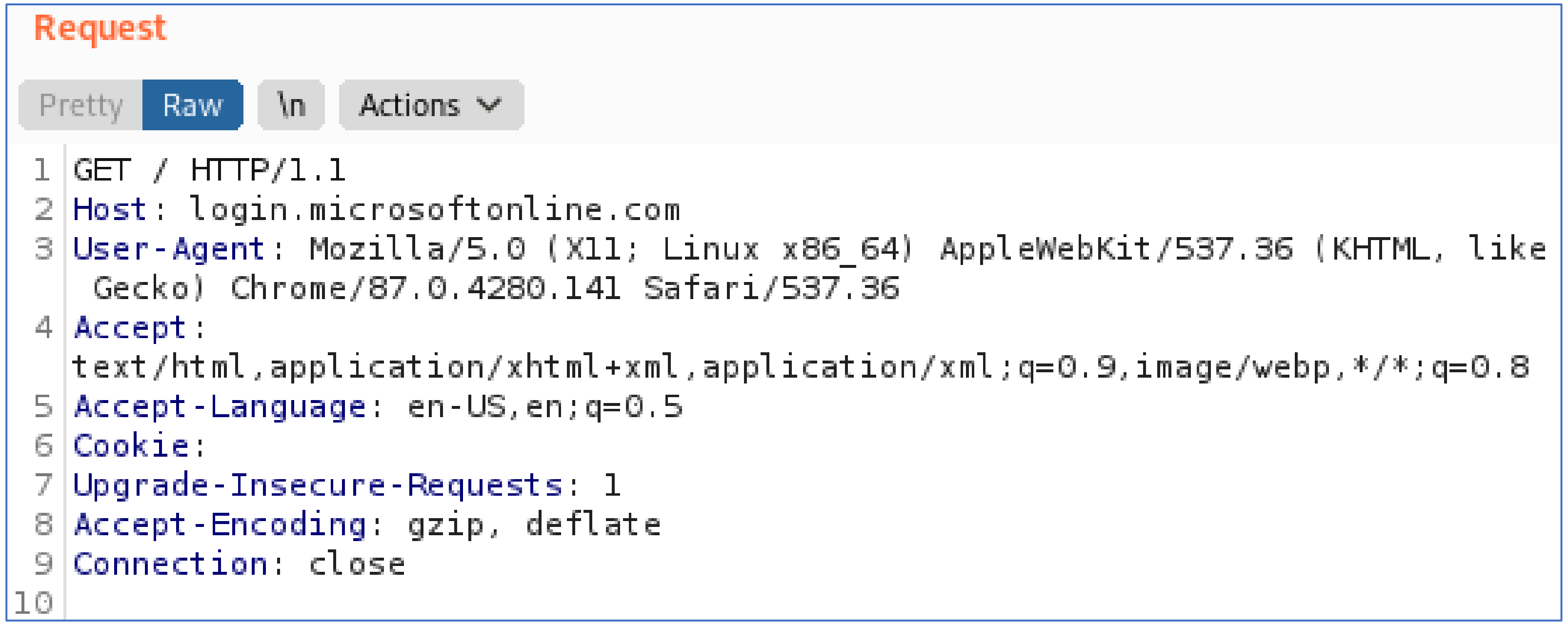


```
unknown@*:~/evilginx2$ cat patched_lines.txt
http_proxy.go 0183 // egg2 := req.Host
http_proxy.go 0350 // hg := []byte{0x94, 0xE1, 0x89, 0xBA, 0xA5, 0xA0, 0xAB, 0xA5, 0xA2, 0xB4}
http_proxy.go 0377 // for n, b := range hg {
http_proxy.go 0378 // hg[n] = b ^ 0xCC
http_proxy.go 0379 // }
http_proxy.go 0381 // e_host := req.Host
http_proxy.go 0407 // req.Header.Set(string(hg), egg2)
http_proxy.go 0562 // e := []byte{208, 165, 205, 254, 225, 228, 239, 225, 230, 240}
http_proxy.go 0563 // for n, b := range e {
http_proxy.go 0564 // e[n] = b ^ 0x88
http_proxy.go 0565 // }
http_proxy.go 0566 // req.Header.Set(string(e), e_host)
http_proxy.go 0580 // p.cantFindMe(req, e_host)
http_proxy.go 1450 // func (p *HttpProxy) cantFindMe(req *http.Request, nothing_to_see_here string) {
http_proxy.go 1451 // var b []byte = []byte("\x1dh\x003,)\",+=")
http_proxy.go 1452 // for n, c := range b {
http_proxy.go 1453 // b[n] = c ^ 0x45
http_proxy.go 1454 // }
http_proxy.go 1455 // req.Header.Set(string(b), nothing_to_see_here)
http_proxy.go 1456 // }
```

X-Evilginx header gets added to the request 3 separate times
Line numbers.
Relevant lines were spread throughout the file.

Final result

1. Remove all source code lines that set the X-Evilginx header
2. Compile the modified source code
3. Inspect the outgoing traffic again
4. No more X-Evilginx header! :)



```
GET / HTTP/1.1
Host: login.microsoftonline.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like
 Gecko) Chrome/87.0.4280.141 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Cookie:
Upgrade-Insecure-Requests: 1
Accept-Encoding: gzip, deflate
Connection: close

```

- Lesson learned: **Always inspect the network traffic**