

KringleCon 4: Calling Birds!

The Great Room



1. Click to talk to Angel Candysalt

Greetings North Pole visitor! I'm Angel Candysalt!

A euphemism? No, that's my name. Why do people ask me that?

Anywho, I'm back at Santa's Splunk terminal again this year.

There's always more to learn!

Take a look and see what you can find this year.

With who-knows-what going on next door, it never hurts to have sharp SIEM skills!

2. Click [splunk](#) > terminal

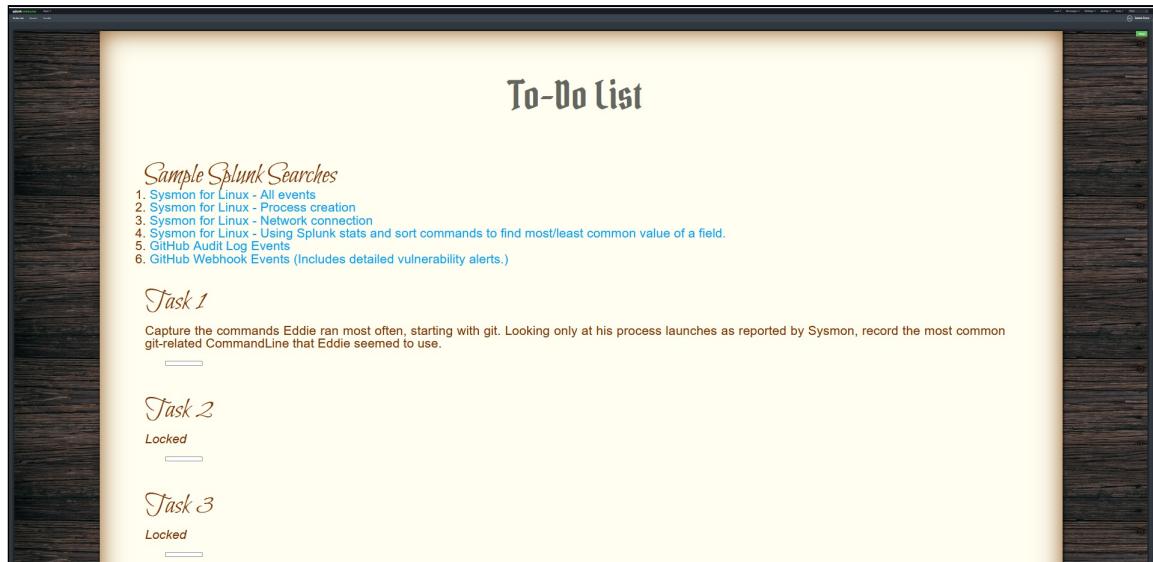
Santa's To-Do List

x

1. Your goal is to complete the eight tasks below.
2. When you complete the final task, you will see a special message to paste into your KringleCon Badge.
3. Eddie McJingles was a key DevOps engineer in Santa's North Pole Partner Program, but he left suddenly. Your job is to document Eddie's project.
4. To complete this challenge, you need to search in Splunk and maybe a few places on the Internet! To access the Splunk search interface, just click the [Search](#) link in the navigation bar in the upper left hand corner of the page.
5. New to Splunk? Check out the sample search links provided.
6. This challenge is designed for a laptop or desktop computer with screen width of 1600 pixels or more.
7. **WARNING** This is a defensive challenge. Do not attack this system, Splunk, Splunk apps, or back-end APIs. Thank you!

[Close](#)

3. Click the **Close** button



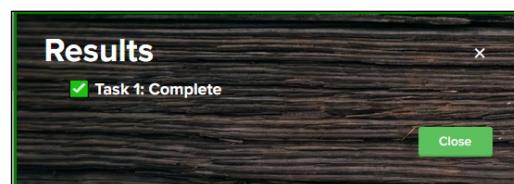
4. Click the **Sysmon for Linux - Using Splunk stats and sort commands to find most/least common value of a field.** link

The screenshot shows the Splunk search interface. At the top, there's a search bar with the query: `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 user=eddie | stats count by CommandLine | sort -count`. Below the search bar, it says "136 events" and "All time". The results are displayed in a table with columns "CommandLine" and "count". The table shows several entries, with "docker ps" having a count of 10, "git status" having a count of 5, and others like "-bash", "/bin/sh /usr/bin/lesspipe", "/usr/lib/git-core/git rev-list --objects --stdin --not --all --quiet --alternate-refs", "locale", and "ls --color=auto" each having a count of 4.

CommandLine	count
docker ps	10
git status	5
-bash	4
/bin/sh /usr/bin/lesspipe	4
/usr/lib/git-core/git rev-list --objects --stdin --not --all --quiet --alternate-refs	4
locale	4
ls --color=auto	4

5. Click the **Second** tab and then click the **Task 1** text-box

6. Type **git status**



7. Click the **Close** button

To-Do List

Sample Splunk Searches

1. Sysmon for Linux - All events
2. Sysmon for Linux - Process creation
3. Sysmon for Linux - Network connection
4. Sysmon for Linux - Using Splunk stats and sort commands to find most/least common value of a field.
5. GitHub Audit Log Events
6. GitHub Webhook Events (Includes detailed vulnerability alerts.)

Task 1

Capture the commands Eddie ran most often, starting with git. Looking only at his process launches as reported by Sysmon, record the most common git-related CommandLine that Eddie seemed to use.

git status

Task 2

Looking through the git commands Eddie ran, determine the remote repository that he configured as the origin for the 'partnerapi' repo. The correct one!

8. Click the **Third** tab and then press **CTRL + F**

9. Type **add origin**

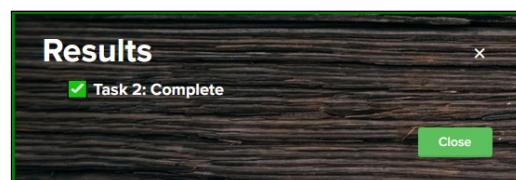
CommandLine	count
gettext usage: \$dashless \$USAGE	1
git --exec-path	1
git -C -c core.quotePath=false diff-index --name-only --relative HEAD -- pack*	1
git branch -M main	1
git clone --depth=1 -q -b 1.1.7-alpha.4 git://github.com/richardgirges/express-fileupload.git /home/eddie/.npm/_cacache/tmp/git-clone-17e21124	1
git commit -a -m Checked into the new GitHub org per the Big Guy	1
git commit package.json -m Added holiday-utils-js dependency	1
git init	1
git ls-remote -h -t git://github.com/richardgirges/express-fileupload.git	1
git pull	1
git push	1
git remote -v	1
git remote add origin git@github.com:elfnp3/partnerapi.git	1
git remote add origin https://github.com/elfnp3/partnerapi.git	1

add origin

^ v Highlight All Match Case Match Diacritics Whole Words 1 of 2 matches X

10. Click the **Second** tab and then click the **Task 2** text-box

11. Type **git@github.com:elfnp3/partnerapi.git**



12. Click the **Close** button



13. Close the **Third** tab and then click the **Sysmon for Linux - All events** link

Time	Event
24/11/2021 17:19:09.949	<Event><System><Provider Name="Linux-Sysmon" Guid="{ff032593-a8d3-4f13-b0d6-01fc615a0f97}" /><EventID>23</EventID><Version>5</Version><Level>4</Level><Task>23</Task><Opcode>0</Opcode><Keywords>0x800000000000</Keywords><TimeCreated SystemTime="2021-11-24T17:19:09.949215000Z"/><EventRecordID>39370398</EventRecordID><Correlation/><Execution ProcessID="686" ThreadID="686"/><Channel>Linux-Sysmon /Operational</Channel><Computer>mcjingles-1</Computer><Security UserID="0"/><System><EventData><Data Name="RuleName">-</Data><Data Name="UtcTime">2021-11-24 17:19:08.703</Data><Data Name="ProcessGUID">(ec26d882-58e9-619d-08d0-232812560000)</Data><Data Name="ProcessID">1081</Data><Data Name="User" /><Data Name="Image">/opt/splunkforwarder/bin/splunkd</Data><Data Name="TargetFilename" />/opt/splunkforwarder/var/spool/splunk/tracker.log</Data><Data Name="Hashes">-</Data><Data Name="Is"

14. Add **CommandLine="docker*"** to the end of the Search

15. Click the **Search** button

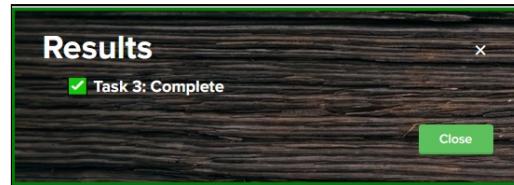
The screenshot shows the Splunk Enterprise search interface. The search bar contains the query: `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational CommandLine="docker*"`. The results section shows 19 events from 09/09/2020 18:05:22.000 to 24/12/2021 19:34:00.000. A single event is highlighted in the timeline for Mon Nov 29 2021 at 14:13:37.241. The selected fields are listed on the left: `a CommandLine 10`, `a host 1`, `a source 1`, and `a sourcetype 1`.

16. Click **CommandLine**

Top 10 Values	Count	%
docker ps	10	52.632%
docker compose up	1	5.263%
docker-init --version	1	5.263%
docker-untar /var/lib/docker/overlay2 /374631798e22c9223595c98382d31b99a9460d3f3fb3614	1	5.263%
4520f8a7891d614d/diff		
docker-untar /var/lib/docker/overlay2 /683f004fc3702d1483d21e2f516c0cf3fdb721007e7581	1	5.263%
85299ecc2dd244bd/diff		
docker-untar /var/lib/docker/overlay2 /76c9fdef6261aae741d008e2b444b6ccb6bef05c9bc49a	1	5.263%
d25aff7b3b8fb62a0/merged/home/dwvs-node		
docker-untar /var/lib/docker/overlay2 /902ae10f7c5f7704c298a1885cf93e83dfe5a543682a6df	1	5.263%
280a27eed642c20f6/diff		
docker-untar /var/lib/docker/overlay2 /b22e9df04bc82018ee3162a8812d217a2e6c9a8ab0c816	1	5.263%
93bb561977352f8a1/diff		
docker-untar /var/lib/docker/overlay2 /f079db12f1375995d81a2046a8780f3e830de432be4924b	1	5.263%
3f41d5f32432c3aff/merged		
docker-untar /var/lib/docker/tmp/docker-builder803754044	1	5.263%

17. Click the **Second** tab and then click the **Task 3** text-box

18. Type `docker compose up`



19. Click the `Close` button

Sample Splunk Searches

- Sysmon for Linux - All events
- Sysmon for Linux - Process creation
- Sysmon for Linux - Network connection
- Sysmon for Linux - Using Splunk stats and sort commands to find most/least common value of a field.
- Github Audit Log Events
- Github Webhook Events (Includes detailed vulnerability alerts.)

Task 1
Capture the commands Eddie ran most often, starting with git. Looking only at his process launches as reported by Sysmon, record the most common git-related CommandLine that Eddie seemed to use.
 git status

Task 2
Looking through the git commands Eddie ran, determine the remote repository that he configured as the origin for the 'partnerapi' repo. The correct one!
 git clone https://github.com/elfnp3/partnerapi.git

Task 3
Eddie was running Docker on his workstation. Gather the full command line that Eddie used to bring up a the partnerapi project on his workstation.
 docker compose up

Task 4
Eddie had been testing automated static application security testing (SAST) in GitHub. Vulnerability reports have been coming into Splunk in JSON format via GitHub webhooks. Search all the events in the main index in Splunk and use the sourcetype field to locate these reports. Determine the URL of the vulnerable GitHub repository that the elves cloned for testing and document it here. You will need to search outside of Splunk (try GitHub) for the original name of the repository.

20. Close the `Third` tab

21. Click the [GitHub Webhook Events \(Includes detailed vulnerability alerts.\)](#) link

New Search

index=main sourcetype=github_json

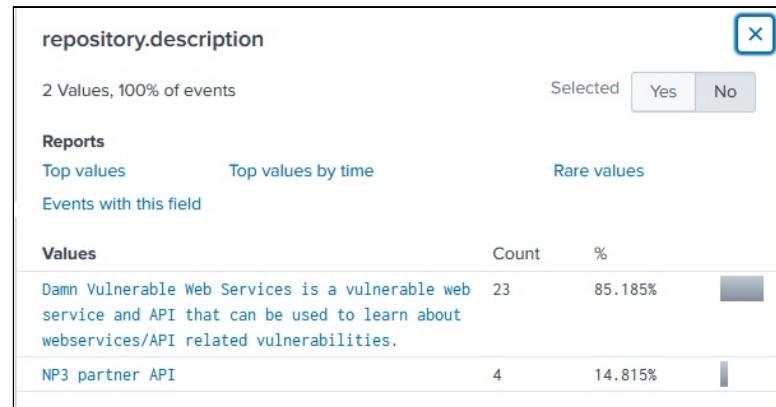
27 events (09/09/2020 18:05:22.000 to 24/12/2021 15:39:22.000) No Event Sampling

Events (27) Statistics Visualization

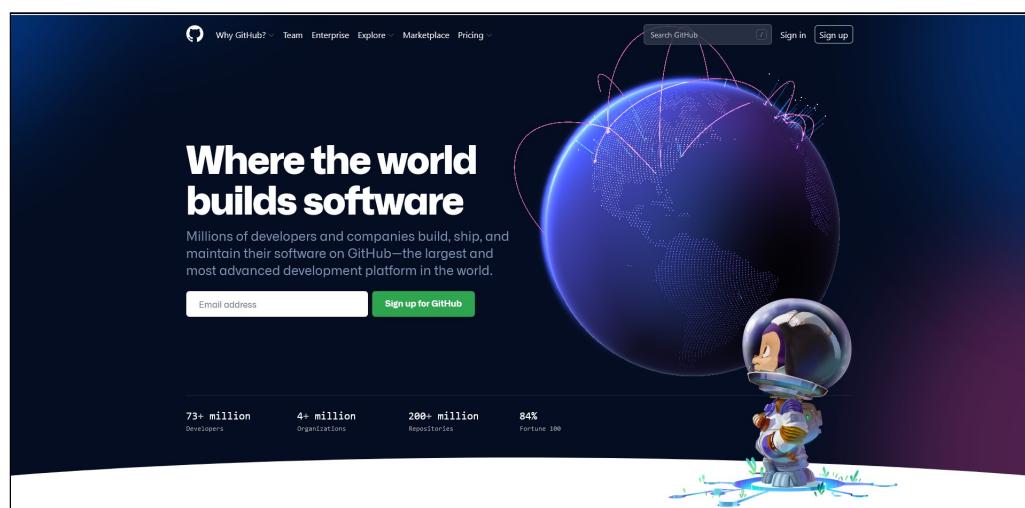
Format Timeline ▾ - Zoom Out + Zoom to Selection X Deselect 1 day per column

List ▾ ✓ Format 20 Per Page ▾			
◀ Hide Fields	☰ All Fields	i Time	Event
SELECTED FIELDS	> 24/11/2021 16:42:41.000	{ [-]	after: 058ac9be65edfc3a1996763e6a816e9162cba3a1 base_ref: null before: 431f0687fe734baea4537d1d3da4ab591854926b commits: [*+]] compare: https://github.com/elfnp3/partnerapi/compare/431f0687fe73...058ac9be65ed created: false deleted: false
INTERESTING FIELDS	a host 1 a source 1 a sourcetype 1	a action 1 a alert.created_at 1 a alert.dismissed_at 1	

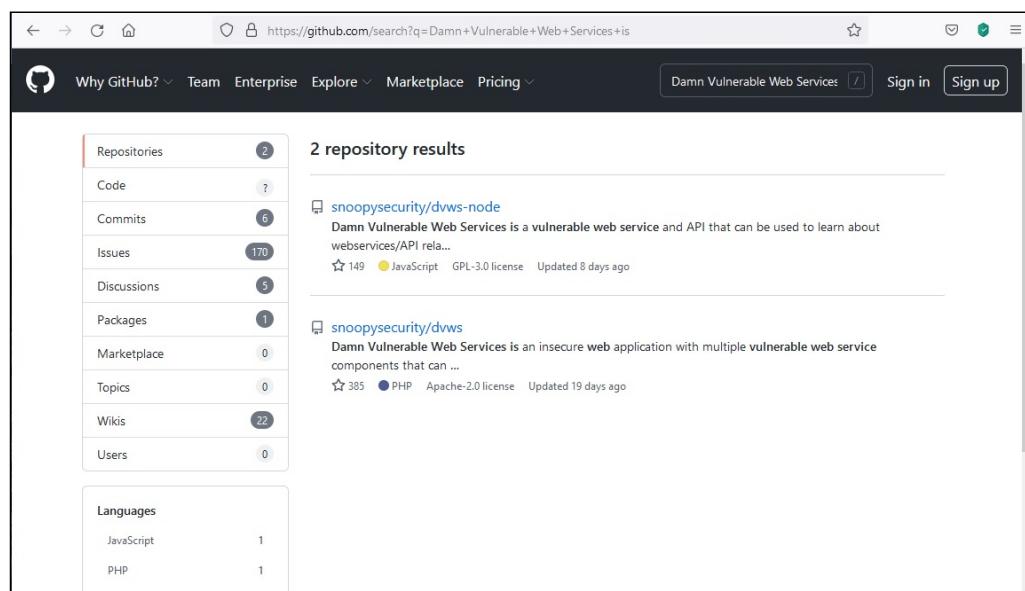
22. Scroll Down and then click `repository.description`



23. Open a **Fourth** tab and then navigate to <https://github.com/>



24. Type **Damn Vulnerable Web Services is** into the Search GitHub text-box



25. Click the [snoopysecurity/dvws-node](#) link and then scroll Down

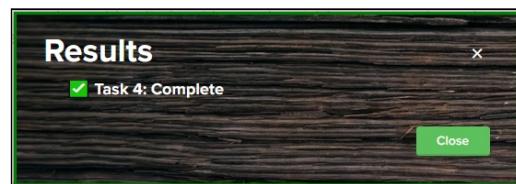
Damn Vulnerable Web Service is a Damn Vulnerable Insecure API/Web Service. This is a replacement for <https://github.com/snoopysecurity/dvws>

This vulnerable application contains the following API/Web Service vulnerabilities:

- Insecure Direct Object Reference
- Horizontal Access Control Issues

26. Close the **Fourth** tab and then click the **Second** tab

27. Click the **Task 4** text-box and then type **dvws-node**



28. Click the **Close** button

Task 1
Capture the commands Eddie ran most often, starting with git. Looking only at his process launches as reported by Sysmon, record the most common git-related CommandLine that Eddie seemed to use.
 git status

Task 2
Looking through the git commands Eddie ran, determine the remote repository that he configured as the origin for the 'partnerapi' repo. The correct one!
 <https://github.com/ellrp3/partnerapi.git>

Task 3
Eddie was running Docker on his workstation. Gather the full command line that Eddie used to bring up a the partnerapi project on his workstation.
 docker compose up

Task 4
Eddie had been testing automated static application security testing (SAST) in GitHub. Vulnerability reports have been coming into Splunk in JSON format via GitHub webhooks. Search all the events in the main index in Splunk and use the sourcetype field to locate these reports. Determine the URL of the vulnerable GitHub repository that the elves cloned for testing and document it here. You will need to search outside of Splunk (try GitHub) for the original name of the repository.
 <https://com/snoopysecurity/dvws-node>

Task 5
Santa asked Eddie to add a JavaScript library from NPM to the 'partnerapi' project. Determine the name of the library and record it here for our workshop documentation.

29. Close the **Third** tab

30. Click the [Sysmon for Linux - Using Splunk stats and sort commands to find most/least common value of a field. link](#)

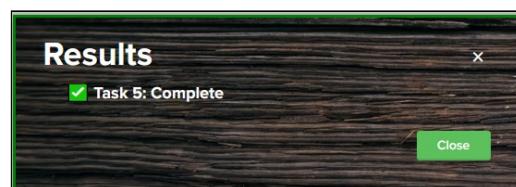
31. Press **CTRL + F** and then type **npm**

The screenshot shows a search interface with a search bar containing 'npm'. Below the search bar is a list of 8 matches found in a log file. The matches are listed in descending order of count, with the first two having a count of 2. The matches include various command-line entries such as 'git push -u origin main', 'sed -e s/-/ /', 'vi package.json', and '/usr/bin/env node /home/eddie/partnerapi/node_modules/.bin/node-pre-gyp install --fallback-to-build'.

Match	Count
git push -u origin main	2
sed -e s/-/ /	2
vi package.json	2
/bin/bash preinstall.sh	1
/bin/sh -c /bin/bash preinstall.sh	1
/bin/sh /usr/lib/git-core/git-submodule update -q --init --recursive	1
/bin/systemctl --user set-environment DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1002/bus	1
/lib/systemd/systemd --user	1
/usr/bin/env node /home/eddie/partnerapi/node_modules/.bin/node-pre-gyp install --fallback-to-build	1
/usr/bin/env node /home/eddie/partnerapi/node_modules/libxmljs/node_modules/.bin/node-pre-gyp install --fallback-to-build --loglevel http	1
/usr/bin/env node /usr/bin/npm install	1
/usr/bin/env node /usr/bin/npm install holiday-utils-js	1
/usr/bin/node /usr/share/npm/node_modules/update-notifier/check.js {"pkg": "holiday-utils-js", "name": "npm", "version": "6.14.4"}	1
/usr/bin/python3 /usr/bin/docker-compose up	1

32. Close the **Third** tab and then click the **Task 5** text-box

33. Type **holiday-utils-js**



34. Click the **Close** button

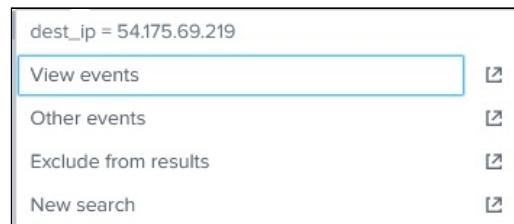
The screenshot shows a task list interface with several tasks:

- Task 2**: Looking through the git commands Eddie ran, determine the remote repository that he configured as the origin for the 'partnerapi' repo. The correct one! (checkbox checked, value: `github.com:elfnp3/partnerapi.git`)
- Task 3**: Eddie was running Docker on his workstation. Gather the full command line that Eddie used to bring up a the partnerapi project on his workstation. (checkbox checked, value: `docker compose up`)
- Task 4**: Eddie had been testing automated static application security testing (SAST) in GitHub. Vulnerability reports have been coming into Splunk in JSON format via GitHub webhooks. Search all the events in the main index in Splunk and use the sourcetype field to locate these reports. Determine the URL of the vulnerable GitHub repository that the elves cloned for testing and document it here. You will need to search outside of Splunk (try GitHub) for the original name of the repository. (checkbox checked, value: `j.com/snappysecurity/dwss-node`)
- Task 5**: Santa asked Eddie to add a JavaScript library from NPM to the 'partnerapi' project. Determine the name of the library and record it here for our workshop documentation. (checkbox checked, value: `holiday-utils-js`)
- Task 6**: Another elf started gathering a baseline of the network activity that Eddie generated. Start with [their search](#) and capture the full process_name field of anything that looks suspicious. (checkbox unchecked, value:)

35. Click their search

dest_ip	dest_port	count
192.30.255.113	9418	2
54.175.69.219	16842	1

36. Click 54.175.69.219



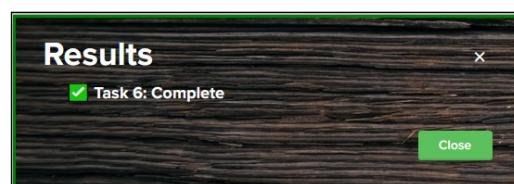
37. Click View event

38. Scroll Down and then click process_name

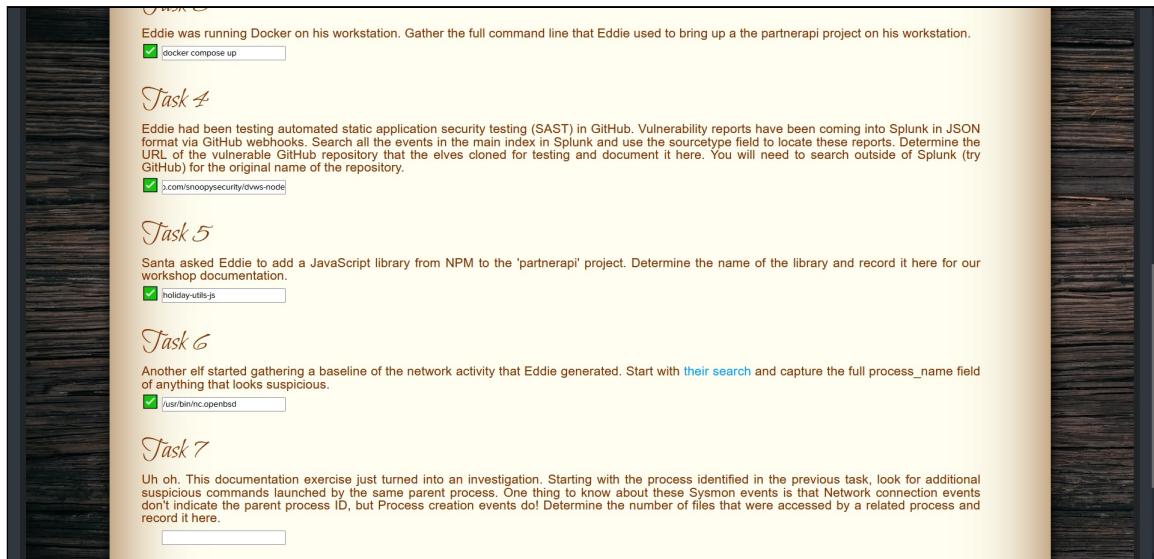
Values	Count	%
/usr/bin/nc.openbsd	1	100%

39. Click the Second tab and then click the Task 6 text-box

40. Type /usr/bin/nc.openbsd



41. Click the **Close** button



42. Click the **Third** tab and then click **/usr/bin/nc.openbsd**

Time	Event
11/24/21 2:16:23.739 PM	<Event><System><Provider Name="Linux-Sysmon" Guid="{ff032593-a8d3-4f13-b0d6-01fc615a0f97}"><EventID>3</EventID><Version>5</Version><Level>4</Level><Task>3</Task><Opcode>0</Opcode><Keywords>0x8000000000000000</Keywords><TimeCreated SystemTime="2021-11-24T14:16:23.739276000Z"/><EventRecordID>39367582</EventRecordID><CorrelationID><Execution ProcessID="686" ThreadID="686"><Channel>Linux-Sysmon/Operational/Channel</Channel><Computer>emcjingles-l</Computer><Security UserID="0"/><System><EventData Name="RuleName"></EventData><EventData Name="UtcTime">2021-11-24 14:16:22.492</EventData><EventData Name="ProcessGuid">{ec26d882-4936-619e-0537-70ed74550000}</EventData><EventData Name="ProcessId">6791</EventData><EventData Name="Image">/usr/bin/nc.openbsd</EventData><EventData Name="User">eddie</EventData><EventData Name="Protocol">tcp</EventData>

43. Scroll **Down** and then click **PID**

Value	Count	%
6791	1	100%

44. Modify the Search to **index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 PID=6791**

45. Click the Search button

The screenshot shows the Splunk interface for a "New Search". The search bar at the top contains the query: `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 PID=6791`. Below the search bar, it says "1 event (9/9/20 6:05:22.000 PM to 12/24/21 6:25:53.000 PM)" and "No Event Sampling". The "Events (1)" tab is selected. The event list shows one entry with the following details:

- Time:** 11/24/21 2:16:23.668 PM
- Event:** XML payload (too long to show fully here)

On the left, there are sections for "SELECTED FIELDS" (including `a action`, `a host`, `# PID`, `a source`, `a sourcetype`) and "INTERESTING FIELDS" (including `a BOOT_ID`, `a COMM`, `a CommandLine`, `a Company`, `a Computer`). The bottom right of the interface has a green "Search" button.

46. Scroll Down and then click ParentProcessId

This is a modal window titled "ParentProcessId". It displays the following information:

- Selected:** Yes
- Reports:**
 - Average over time
 - Maximum value over time
 - Minimum value over time
 - Top values
 - Top values by time
 - Rare values
- Events with this field:**
 - Avg: 6788 Min: 6788 Max: 6788 Std Dev: 0
- Values:**

Values	Count	%
6788	1	100%

47. Modify the Search to `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 ParentProcessId=6788`

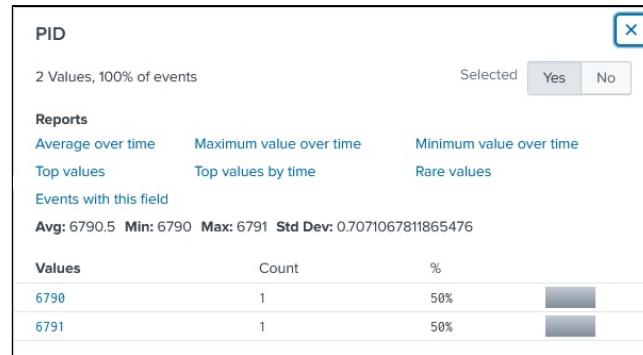
48. Click the Search button

The screenshot shows the Splunk interface for a "New Search". The search bar at the top contains the modified query: `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 ParentProcessId=6788`. Below the search bar, it says "2 events (09/09/2020 18:05:22.000 to 24/12/2021 19:27:36.000)" and "No Event Sampling". The "Events (2)" tab is selected. The event list shows two entries with the following details:

- Time:** 24/11/2021 14:16:23.668
- Event:** XML payload (too long to show fully here)

On the left, there are sections for "SELECTED FIELDS" (including `a CommandLine`) and "INTERESTING FIELDS" (including `a host`, `a source`, `a sourcetype`). The bottom right of the interface has a green "Search" button.

49. Scroll Down and then click PID



50. Click 6790

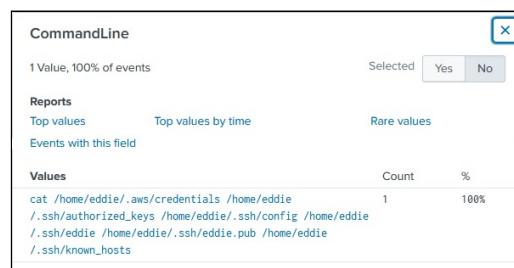
Events (1)

```

<Event><System><Provider Name="Linux-Sysmon" Guid="{ff032593-a8d3-4f13-bd6-01fc615af97}><EventID>1</EventID><Version>5</Version><Level>4</Level><Task>1</Task><Opcode>0</Opcode><Keywords>0x8000000000000000</Keywords><TimeCreated SystemTime="2021-11-24T14:16:23.666733000Z"><EventRecordID>39367579</EventRecordID><Correlation/><Execution ProcessID="686" ThreadID="686"><Channel>Linux-Sysmon/Operational</Channel><Computer>emcjingles-1</Computer><Security UserID="0"/></System><EventData><Data Name="RuleName"></Data><Data Name="UtcTime">2021-11-24 14:16:22.419</Data><Data Name="ProcessGuid">{ec26d882-4936-619e-31d4-00c33d560000}</Data><Data Name="ProcessId">6790</Data><Data Name="Image">/usr/bin/cat</Data><Data Name="FileVersion"></Data><Data Name="Description"></Data><Data Name="Product"></Data><Data Name="Company"></Data><Data Name="OriginalFileName"></Data><Data Name="CommandLine">cat /home/eddie/.aws/credentials /home/eddie/.ssh/authorized_keys /home/eddie/.ssh/config /home/eddie/.ssh/eddie /home/eddie/.ssh/eddie_pub /home/eddie/.ssh/known_hosts</Data><Data Name="CurrentDirectory">/home/eddie/partnerapi/node_modules/holiday-util</Data><Data Name="User">eddie</Data><Data Name="LogonGuid">{ec26d882-460a-619e-ea03-000000000000}</Data>

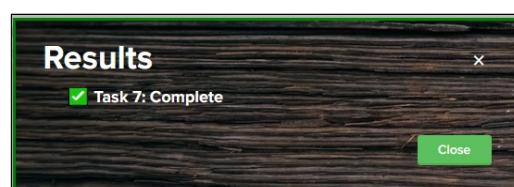
```

51. Click CommandLine

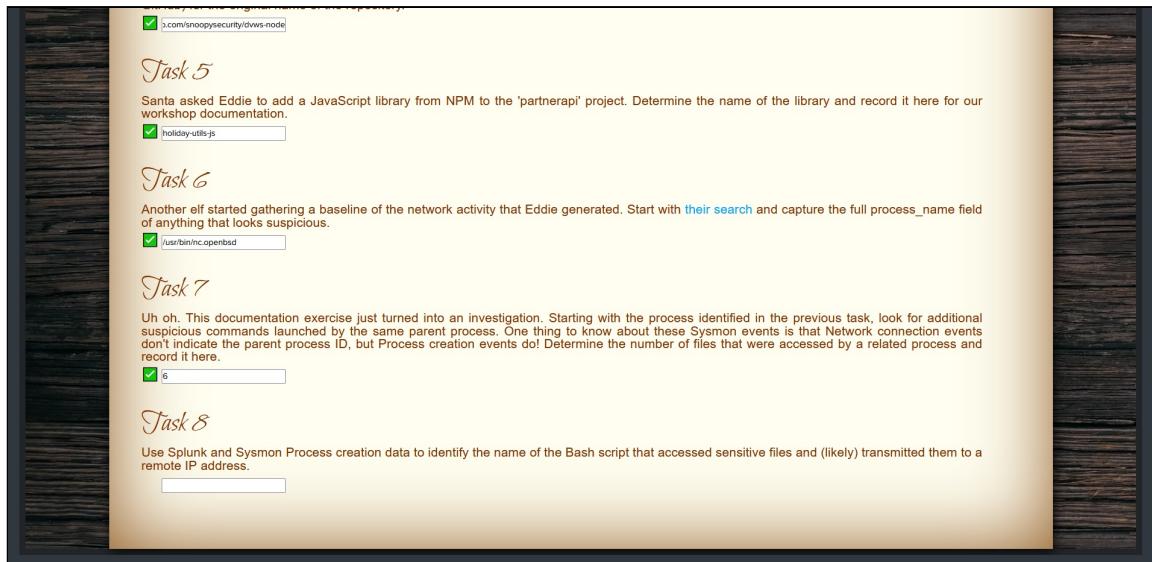


52. Click the Second tab and then click the Task 7 text-box

53. Type 6



54. Click the **Close** button



55. Click the **Third** tab

56. Modify the Search to `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 .sh`

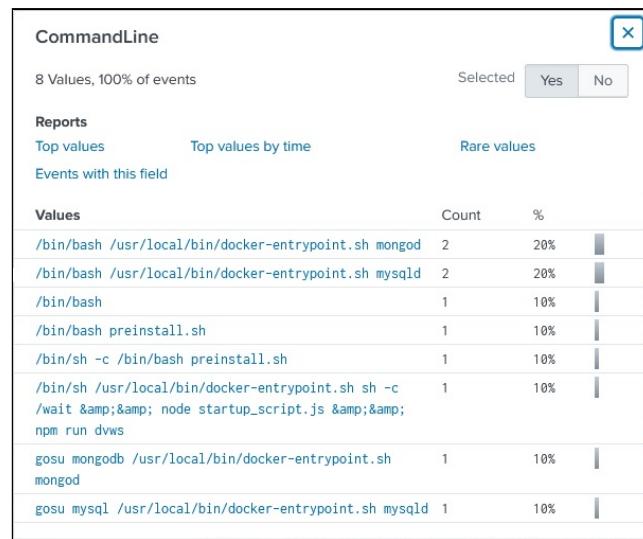
57. Click the **Search** button

The screenshot shows the Splunk Enterprise search interface with the following details:

- Search Bar:** `index=main sourcetype=journald source=Journald:Microsoft-Windows-Sysmon/Operational EventCode=1 .sh`
- Results Summary:** ✓ 10 events (9/9/20 6:05:22.000 PM to 12/24/21 7:42:51.000 PM) | No Event Sampling | Job | All time | Smart Mode
- Event List:** A timeline view showing 10 events from Mon Nov 29, 2021, to Mon Dec 20, 2021. The events are listed in descending order of time.
- Selected Fields:**
 - SELECTED FIELDS: `a CommandLine 8`, `a host 1`, `a source 1`, `a sourcetype 1`
 - Time Filter: `11/24/21 2:16:23.664 PM`
 - Event Data Preview:

```
<Event><System><Provider Name="Linux-Sysmon" Guid="{ff032593-a8d3-4f13-b0d6-01fc615a0f97}"><EventID><Version>5</Version><Level>4</Level><Task>1</Task><Opcode>0</Opcode><Keywords>0x8000000000000000</Keywords><TimeCreated SystemTime="2021-11-24T14:16:23.664352000Z"/><EventRecordID>39367578</EventRecordID><Correlation/><Execution ProcessID="686" ThreadID="686"/><Channel>Linux-Sysmon/Operational</Channel><Computer>emcjingles-1</Computer><Security UserID="0"/></System><EventData><Data Name="RuleName">-</Data><Data Name="UtcTime">2021-11-24 14:16:22.416</Data><Data Name="SourceID">1</Data>
```

58. Click **CommandLine**



59. Close the **Third** tab

60. Type **preinstall.sh**



61. Click the **Close** button

Task 5

Santa asked Eddie to add a JavaScript library from NPM to the 'partnerapi' project. Determine the name of the library and record it here for our workshop documentation.

holiday-utils.js

Task 6

Another elf started gathering a baseline of the network activity that Eddie generated. Start with their search and capture the full process_name field of anything that looks suspicious.

/usr/bin/nc.openbsd

Task 7

Uh oh. This documentation exercise just turned into an investigation. Starting with the process identified in the previous task, look for additional suspicious commands launched by the same parent process. One thing to know about these Sysmon events is that Network connection events don't indicate the parent process ID, but Process creation events do! Determine the number of files that were accessed by a related process and record it here.

6

Task 8

Use Splunk and Sysmon Process creation data to identify the name of the Bash script that accessed sensitive files and (likely) transmitted them to a remote IP address.

preinstall.sh

62. Scroll to the **Top** of the window

The screenshot shows the Splunk interface with the title bar "splunk enterprise". The main content area is titled "To-Do List" with a decorative banner below it. A message says "Thank you for helping Santa complete his investigation! Santa says you're a whiz!". Below this, there's a section titled "Sample Splunk Searches" with a numbered list of 6 items. Two tasks are listed under "Task 1" and one under "Task 2".

Sample Splunk Searches

1. Sysmon for Linux - All events
2. Sysmon for Linux - Process creation
3. Sysmon for Linux - Network connection
4. Sysmon for Linux - Using Splunk stats and sort commands to find most/least common value of a field.
5. GitHub Audit Log Events
6. GitHub Webhook Events (Includes detailed vulnerability alerts.)

Task 1

Capture the commands Eddie ran most often, starting with git. Looking only at his process launches as reported by Sysmon, record the most common git-related CommandLine that Eddie seemed to use.

git status

Task 2

Looking through the git commands Eddie ran, determine the remote repository that he configured as the origin for the 'partnerapi' repo. The correct

63. Close the **Second** tab

64. Click the **Tick** (Objectives) icon and then click **9) Splunk!**

65. Type **whiz**

66. Click the **Submit** button

The screenshot shows the achievement details for objective 9) Splunk!. It includes the achievement name, difficulty level (3 red stars), and a description about solving the challenge in Santa's great hall.

9) Splunk!

Difficulty: ★★★

Help Angel Candysalt solve the Splunk challenge in Santa's great hall. Fitzy Shortstack is in Santa's lobby, and he knows a few things about Splunk. What does Santa call you when you complete the analysis?

The screenshot shows a notification box with a yellow icon and the text "New [Achievement] Unlocked: Splunk!!" followed by a link "Click here to see this item in your badge."

New [Achievement] Unlocked: Splunk!!

[Click here to see this item in your badge.](#)

67. Click **[Exit]**

68. Move **Left** to enter the **Kitchen**