

CSAW HSF 2016 Report for Murder Case of Presidential Candidate Candice Deyte

Case Number: 2729-473-2016

To the OSIRIS Police Organization and any authorized Detective of OSIRIS Investigative Services.

I there by make an Affidavit to the incident of Murder or Death of Candice Deyte, who was a Presidential Candidate, and had "Collapsed mid stride on stage of OSIRIS Studios" as said by the witnesses present at the location.

It was said that she had been there to give a speech on an Influenced Hacker "Pat Rogers".

The physical forensic reports and another proof that is the smart watch she was wearing during the incident, both state the fact that she had died due to a heart attack which was caused by excess stress and the trauma of an incident relating to her opponent.

When the two suspects, Chris Snell and Meghan Rowley were asked to comment on the incident, they however refused, thereby increasing the percentage of them being involved in the case.

Investigation Proofs:

Connect

fedlink.csaw.io

After going to her own website that is <http://voteforme.csaw.io/> and scrolling down till the end, we see another related link under "Connect" - <http://fedlink.csaw.io/>. Here we are able to recognize that this is a LinkedIn for the feds where we can view their profiles.

By using a bit of recon, we come to know that there are 100 profiles and of those 100, we identify that 3 emails are ending with "@gmail.com" domain whereas the rest comprise of a "@usss.dhs.gov".

These 3 emails belong to Chris Snell, Meghan Rowley and Candice Deyte herself.

The screenshot shows a web search tool interface. At the top, there are tabs for 'Search Engine', 'Web Site', 'Url List', 'Website Owners', 'Email Accounts', and 'Local Files'. Below these are icons for 'Start Search', 'Stop', 'Save Emails', 'Clear Results', 'Load Sites', and 'Clear Sites'. The 'Urls' section shows the search path 'http://fedlink.csaw.io/'. The 'Scan depth' is set to 7, and the search is limited to 'only this domain'. A search bar contains the text 'gmail', with 'Find' and 'Clear' buttons. Below the search bar is a table with the following data:

N#	Url	Name	Item
20	http://fedlink.csaw.io/profile/320		sachrissnell@gmail.com
21	http://fedlink.csaw.io/profile/321		sameghanrowley@gmail.com
86	http://fedlink.csaw.io/profile/386		meforprez16@gmail.com

URLs:

<http://fedlink.csaw.io/profile/320>

<http://fedlink.csaw.io/profile/321>

<http://fedlink.csaw.io/profile/386>

Emails:

sachrissnell@gmail.com

sameghanrowley@gmail.com

meforprez16@gmail.com

Looking at these emails we found a common starting with 'sa' in both email addresses (of Chris Snell and Meghan Rowley) which might be a short form of "SECRET AGENTS" or " Société Anonyme" (which means an anonymous company).

Another thing which makes their identities suspicious is they have registered one after another which can be seen from the URL (User IDs: 320, 321)

This makes a relation between Meghan Rowley and Chris Nell.

Furthermore, if we go deeper and try to check if those emails are valid, we went to <https://tools.verifyemailaddress.io/> and wrote down both of the emails. The email addresses did exist as according to the website.

We tried to check the recovery methods of Meghan and Chris's gmail accounts, we come to the fact that they both have a phone number ending with 19. As displayed by the google phone verification page – "●●●●●●●19".

Page Source materials:

```
-1360x860.jpg"> <!--fedlocker.csaw.io-->
```

If we go into the page source of the website <http://voteforme.csaw.io/> and view page source (included in developer tools) and scroll down a bit, we see an html comment which is having a URL "fedlocker.csaw.io". This then takes us to another website namely "FEDLocker". Then again when

inspect the source of the website "<http://fedlocker.csaw.io>" we see a comment from line 3 to 17 saying "aVolition" and "in sqlmap we trust" which gives us a hint that we need/have find a vulnerability in the website SQL database using the python script sqlmap.

```

3  <!--
4
5
6
7
8
9
10
11
12
13
14
15
16  aVoaVoaVoaVoaVoaVoaVoaVoaVoaVoaVoaVoaVoa
17  -->

```

Now we are running the python script 'sqlmap' to get access into the the SQL database.