Improving Analyst Efficiency in Office365 Business Email Compromise Investigation Scenarios Through the Implementation of Open Source Tools

Aaron Elyard

Improving Analyst Efficiency in Office365 Business Email Compromise Investigation Scenarios Through the Implementation of Open Source Tools

GIAC (GSEC) Gold Certification

Author: Aaron Elyard, aelyard@gmail.com

Advisor: Clay Risenhoover

Accepted: *05/12/2020*

Abstract

Working within Microsoft's browser-based O365 Graphical User Interface (GUI) can be challenging for DFIR practitioners when time is of the essence. PowerShell-based cmdlets are often preferred due to their flexibility, speed, and efficiency compared to a browser-based approach. However, in his professional career, the author has observed that more junior analysts may not feel comfortable using command line tools. Additionally, they may not have devoted the appropriate time to learning the various options needed to obtain the data they need for their investigations. This paper explores a tool the author created to bridge the gap between the browser-based GUI and raw PowerShell. It examines the impact of the use of such a tool on the analyst's efficiency, measured in the number of interactive actions an analyst must take.

1. Introduction

Microsoft's browser-based GUI is the dominant way to view and interact with O365 logs during a Business Email Compromise. The problem with the current browser-based process is that the information an analyst wants is often spread amongst multiple Microsoft frontends, including the Security and Compliance Center, Exchange Control Panel, and Azure AD Portal, etc. These pages, while powerful, can be clunky, and the information needed to make a full analysis may be buried under a host of submenus.

PowerShell neatly solves this problem by providing a slew of command line tools called cmdlets. Like surgical tools, these cmdlets are tailored to perform an exact function in a quick manner. However, they present their own set of challenges. They can be intimidating for a junior analyst who has never worked with the command line before. Analysts also require study to learn the precise switches required for each cmdlet to pull exactly what they may need. One cmdlet, such as Get-MsolUser, may expect the User's UserPrincipalName (UPN)("Get-MsolUser", 2020), while another, such as Get-AzureADUser ("Get-AzureADUser", 2020), could expect the unique "ObjectId" field that serves as the Globally Unique Identifier (GUID) for that UPN. These specifics are often listed in the documentation; however, it can be confusing for an analyst attempting to learn to use these cmdlets for the first time. Microsoft does make their documentation widely available; however, it can take time for an analyst to learn and memorize the exact commands necessary for gaining actionable intelligence quickly. Microsoft even makes articles available on their documentation sites that advise security personnel on what to do during an incident, such as "Responding to a Compromised Email Account" (Microsoft, 2020). This article walks an analyst through the steps that need to be taken

via the browser to remediate a compromised email account. However, no such resource exists for the PowerShell commands that would be necessary to complete the same tasks.

While Microsoft does offer an automation solution for performing certain tasks called Automated investigation and response (AIR), it is only available to O365 customers who purchase a certain license type.(Vangel, 2020) Because of this, the automated investigations the author targets with his tool are effectively unavailable to be performed if an organization does not have adequate licensing. It is worth noting that the author's tool does not replace AIR, as AIR has additional functionality related to other investigation types. Rather, both the author's tool and AIR serve to enhance analyst efficiency while working with O365.

Balancing efficiency and ease-of-use is always a difficult task, but even more so in a Security Operations Center (SOC), where many analysts are overwhelmed by alerts daily. According to a survey performed by Exabeam, "One-third of the respondents who reported being understaffed estimate they are short by as many as 6-10 employees" (Ma, 2019). Increased efficiency during workflows is vital to ensure a SOC is performing at its best, even while being understaffed.

To bridge this gap, the following research has developed a tool to solve this discrepancy between an analyst's use of the browser and the efficiency potential of using only PowerShell. It utilizes the PowerShell cmdlets wrapped within a GUI frontend. The overall goal of the tool is to enhance analyst efficiency and effectiveness while providing all the actionable intelligence needed to decide and respond to a Business Email Compromise.

2. Research Methodology

In order to investigate the efficacy of the tool, the number of interactive actions during investigation of a mock Business Email Compromise scenario were tracked. Interactive actions were defined as clicks and text entry. These results were tracked for the investigation of the same incident using both Microsoft's browser-based tools and the author's GUI tool.

This data was then compiled, and the efficiency of the author's tool was compared against Microsoft's browser-based tools. The mock scenario in the next section will be used to show the various features of the tool.

3. Tool Features

The following sections will explain the various features of the author's tool utilizing the mock scenario used to perform the research. In this scenario, an Incident Response analyst is given the task of investigating a suspected Business Email Compromise. The organization's SIEM has alerted to an abnormal login from 77.243.191.27, and the analyst has been tasked with investigating this user to see if the alert is legitimate. If so, they are expected to document why they believe the alert is legitimate and then take appropriate action. It should be noted that various figures have been edited to remove sensitive information from the author's organization.

3.1 User Information

The "User Information" tab, shown in Figure 1, shows basic information about the User, such as Name, Title, Department, Address, and Contact Phone numbers. It also calculates the User's password expiration date and shows any email forwarding settings. All this information is shown in the left pane. In the right pane, license information is shown for the user, sorted on the provisioning status of the license.

3.1.1 Scenario Example

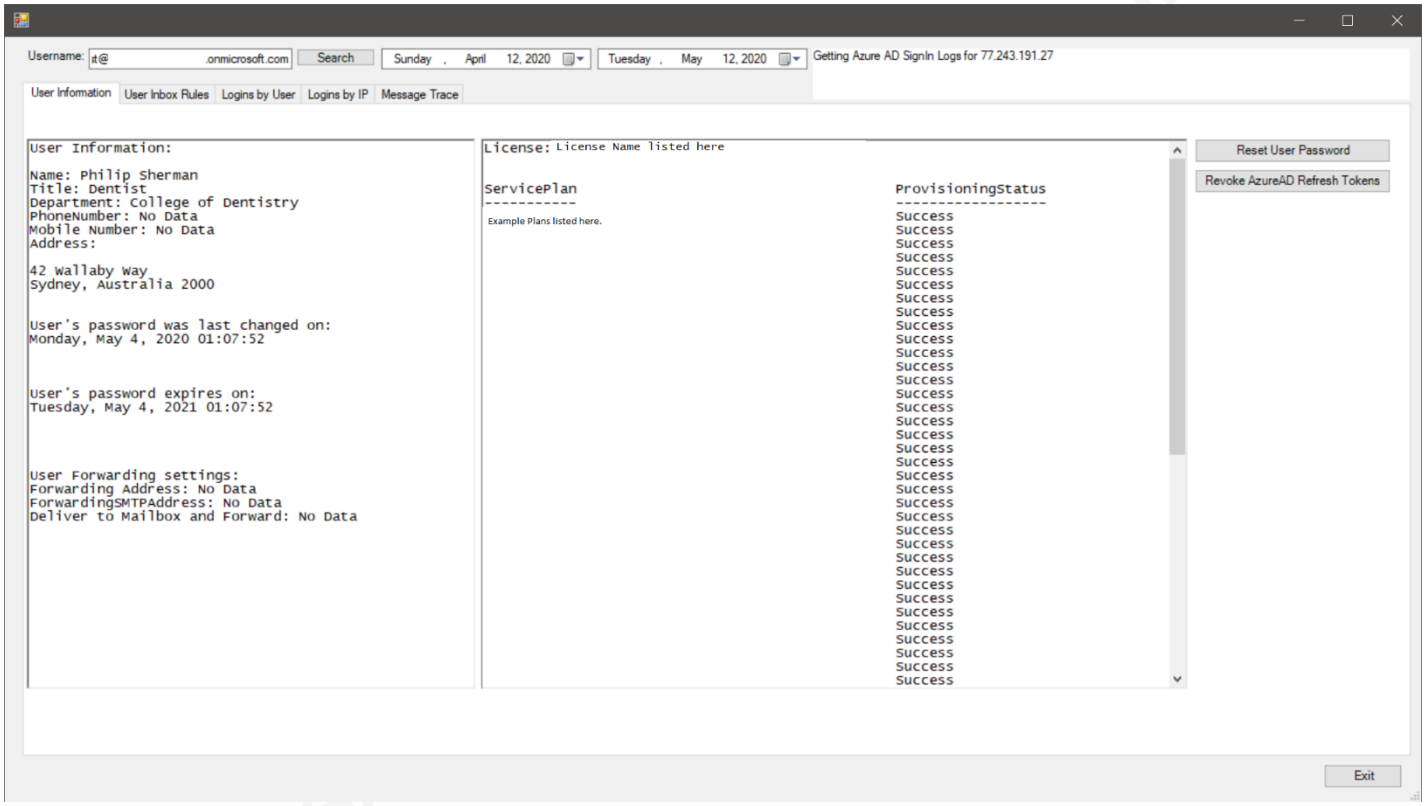

Figure 1. “User Information” Tab

3.2 User Inbox Rules

Using this tab, an analyst is shown a list of the Inbox Rules for a user on the left, and when a rule is selected in the left pane, the right pane will be populated with the Name, Description, and Status of the rule (Enabled/Disabled). This allows an analyst to quickly gather intelligence as to whether a rule is suspect.

3.2.1 Scenario Example

In the mock scenario example shown in Figure 2, this user has one inbox rule, with a name of “asfd;lkjasdf”. As shown, this rule moves any message with the words “Mailbox Update Required” in the subject line into the user’s Conversation History Folder. This was likely placed by the attacker to hide any replies about the email they sent out.

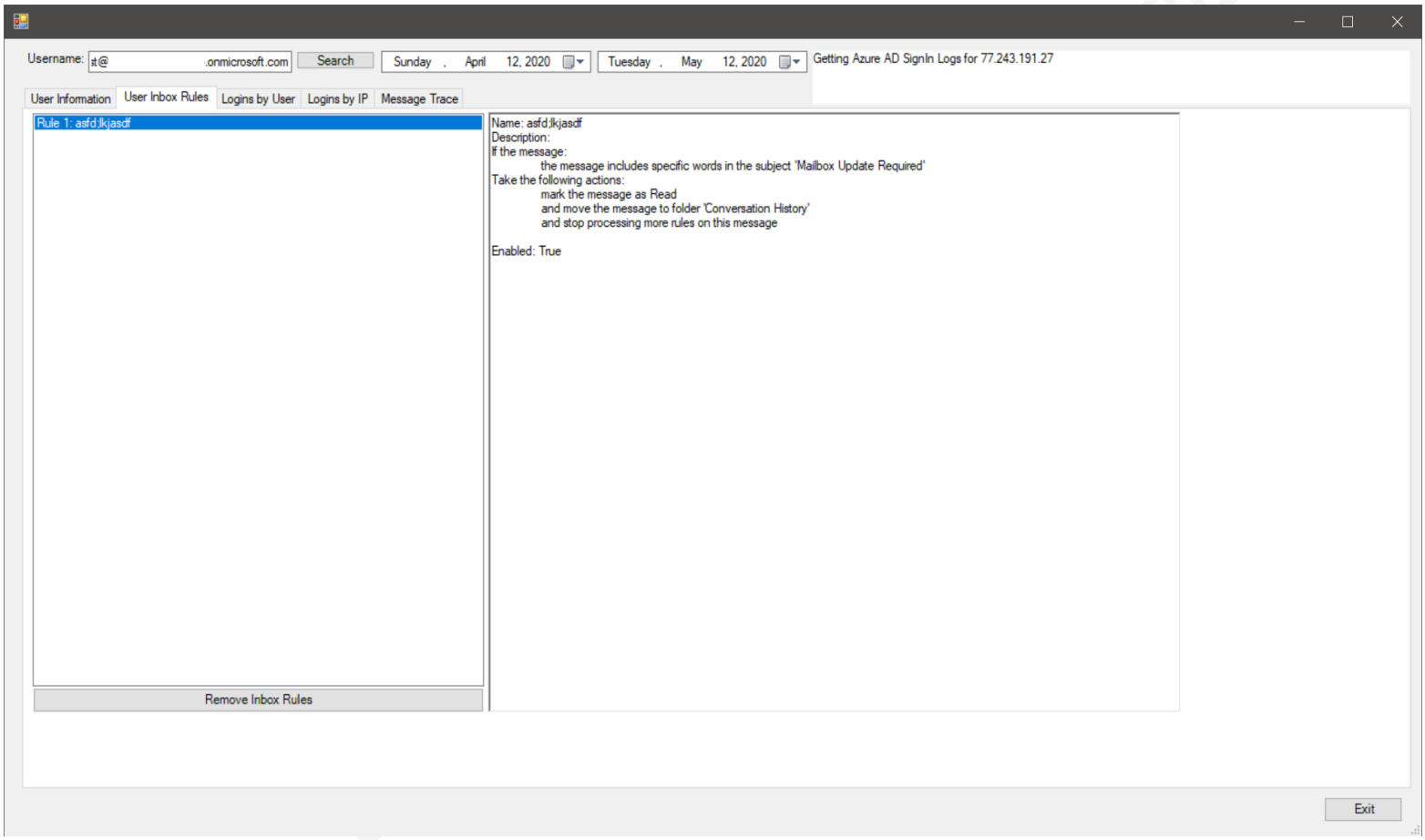

Figure 2. "User Inbox Rules" Tab

3.3 Logins by User

Next, the "Logins by User" is shown in Figure 3. When the analyst clicks the "Get AzureAD SignIn Logs" button, the textbox on the left will populate with statistics about the logs that are pulled, including:

- The number of the unique IP addresses used to sign in.
- The number of the unique browsers/OS combinations used to sign in.
- The number of the Applications the user account signed into.

Additionally, there is the option to view the logs and download them. The View option also allows the analyst to apply granular filtering options to the logs shown to them. Additionally, the analyst can download the logs. Both functionalities are broken

down into “Triage” and “Full”. “Triage” is simply a subset of the Full logs that contains only the fields an analyst may need to quickly make an actionable decision.

3.3.1 Scenario Example

In the mock scenario shown in Figure 3, we can see a login from our IP of interest.

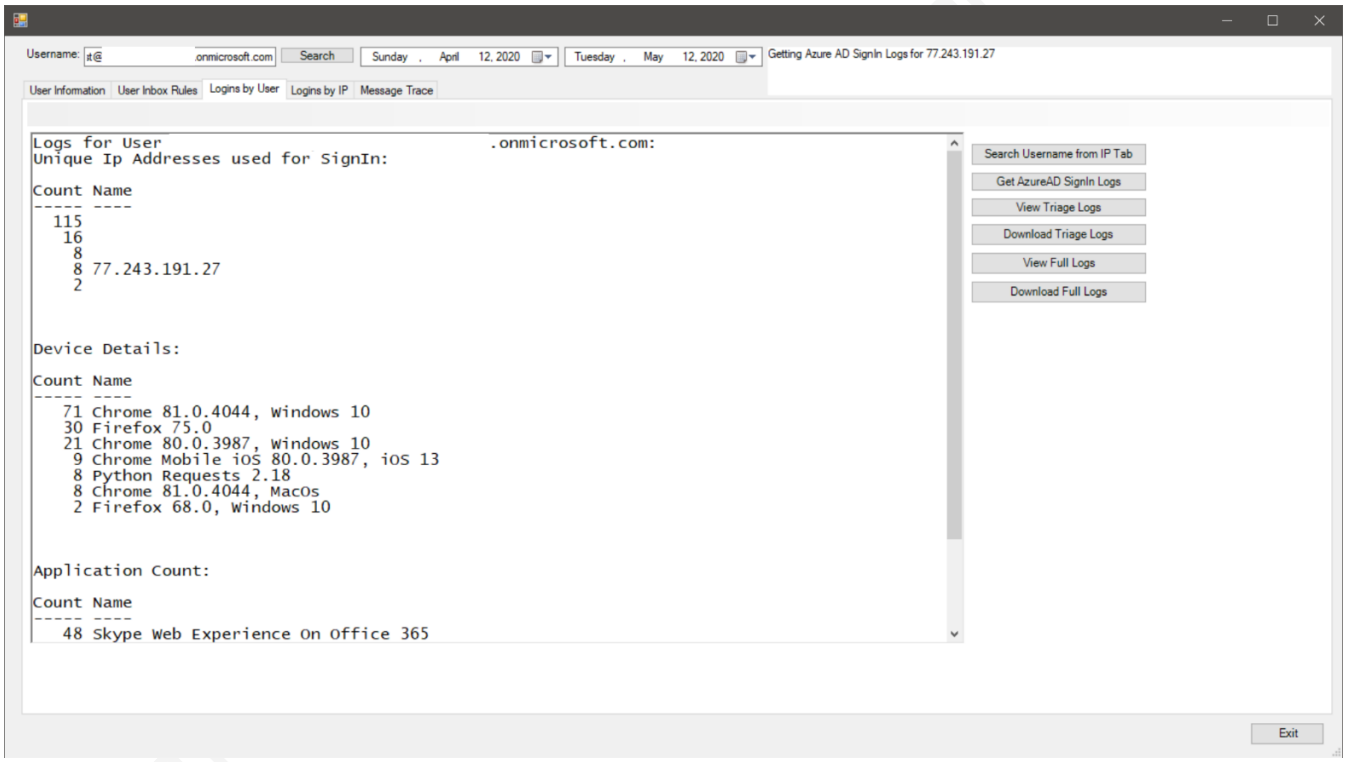

Figure 3. “Logins by User” Tab

3.4 Logins by IP

Next, the analyst has the option to search AzureAD SignIn logs by IP address. The analyst has the option of entering a fresh IP, or they can opt to search using an existing IP from the “Logins by User” tab. Once a search is performed, the textbox will show a similar statistical view to previous tabs:

1) The number of the unique User Accounts used to sign in from an IP Address.
2) The number of unique browsers/OS combinations used to sign in from an IP Address.
3) The number of Applications the IP signed into.

The analyst has a similar set of options to the “Logins by User” tab, in that they can download and view “Triage” and “Full” logs.

3.4.1 Scenario Example

Next, the analyst uses the “Use Existing IP from User Logins” button in order to see if this malicious actor compromised any other accounts. This will open a selection box that allows the analyst to use an IP that they already found. Based on Figure 4, the attacker compromised no other user accounts from this IP. We can tell that because there are only login events from a single user, which is the one we already know about.

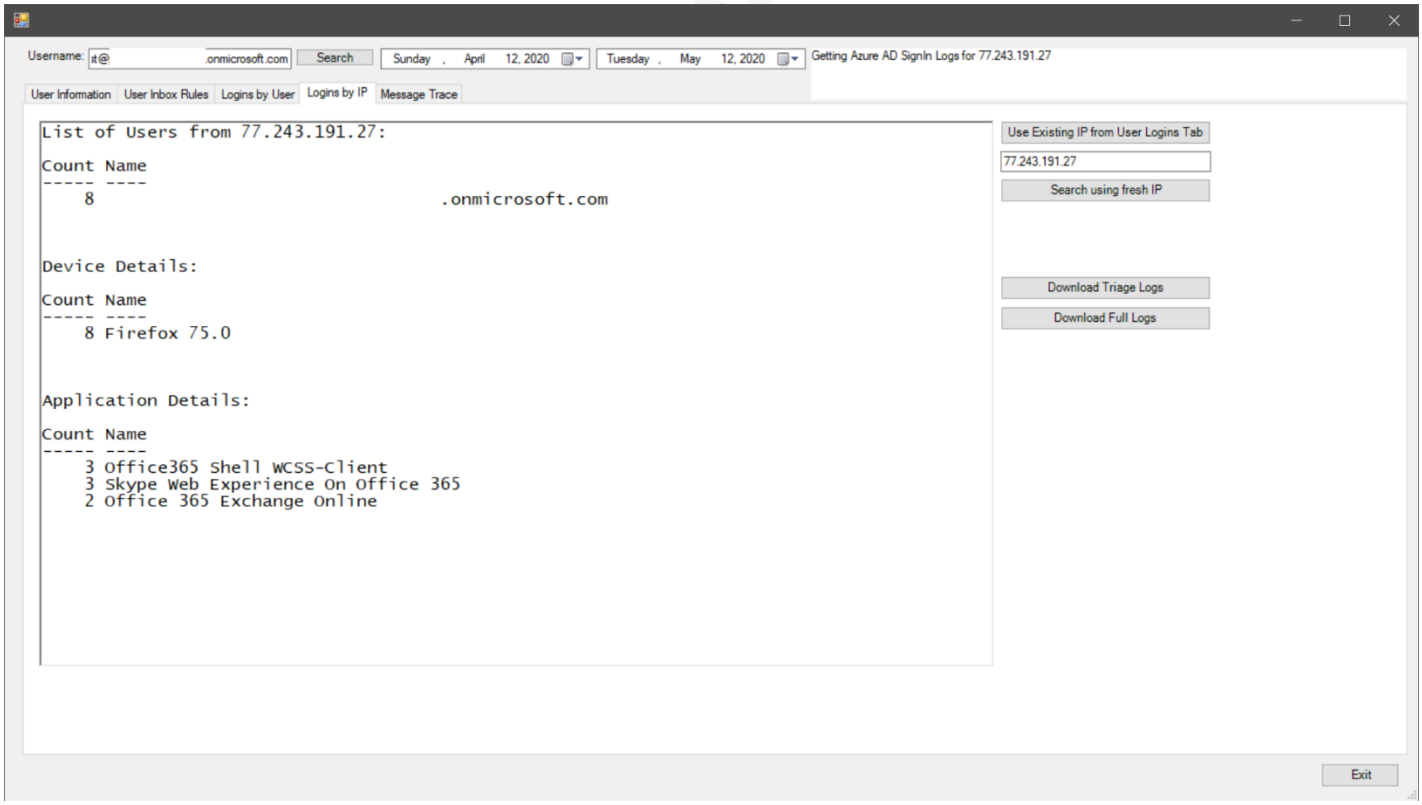

Figure 4. “Logins by IP” Tab

3.5 Message Trace

Next, under “Message Trace”, the analyst can input a “Sender Email Address”, “Recipient Email Address”, or a “From IP”. This will then perform a message trace search, showing which messages were sent, who they were sent to, the IP address that sent/received the message, and its delivery status. The analyst has the option to view and

download the search results. The textbox again shows statistical output of the search, including:

1) Total number of messages
2) List of unique senders and the number of messages
3) List of unique recipients and the number of messages
4) List of the number and delivery status of messages within search
5) List of IPs that messages were sent from.

3.5.1 Scenario Example

Finally, the analyst notes that the attacker IP sent two email messages using this account, as shown in Figure 5. From here, we can use the information provided in the "View Message Trace" output to investigate further in another tool.

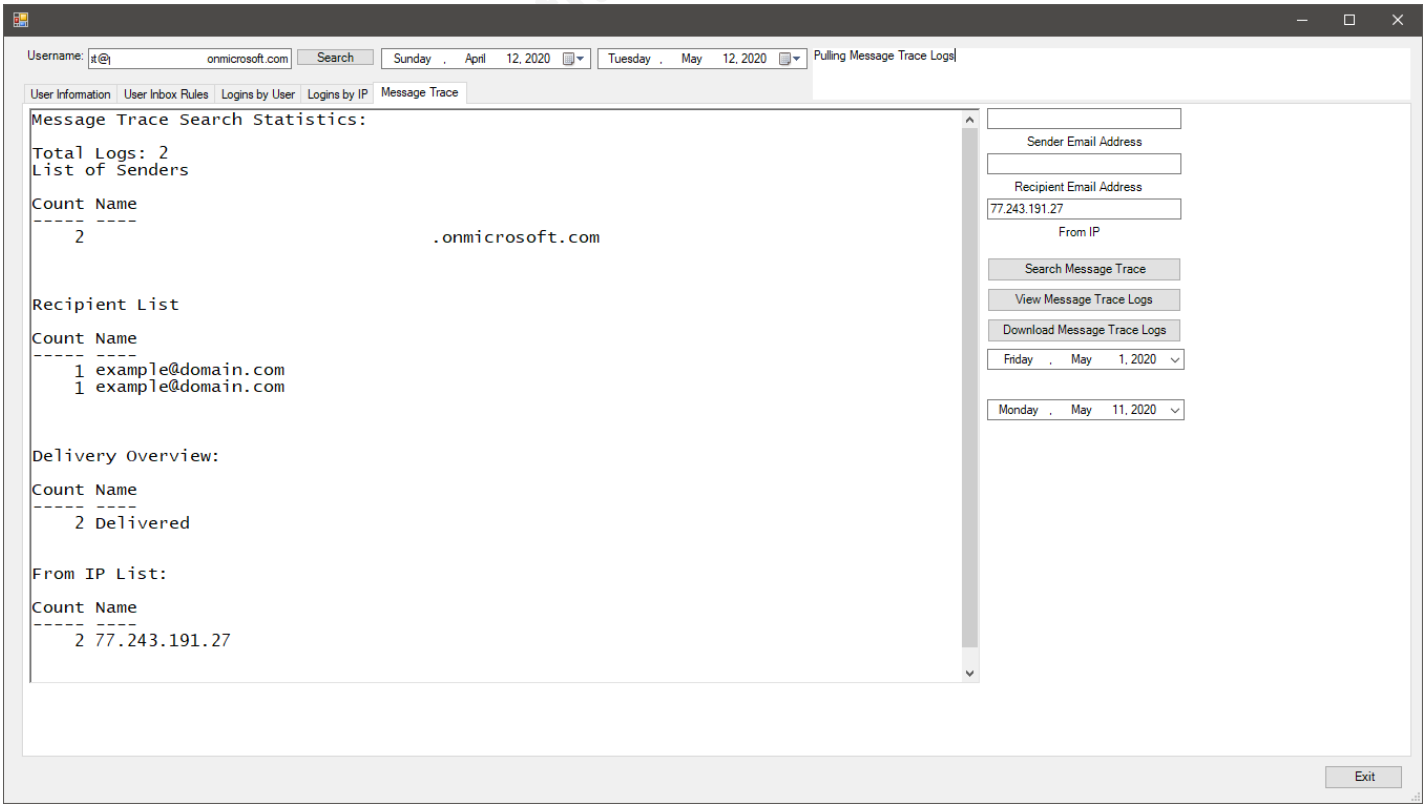

Figure 5. "Search Message Trace" Tab

In Figure 6, the output of "View Message Trace Logs" is shown. This allows for an analyst to view the messages that meet the criteria searched for. Additionally, filters can be applied that allow the analyst to add additional criteria to their view.

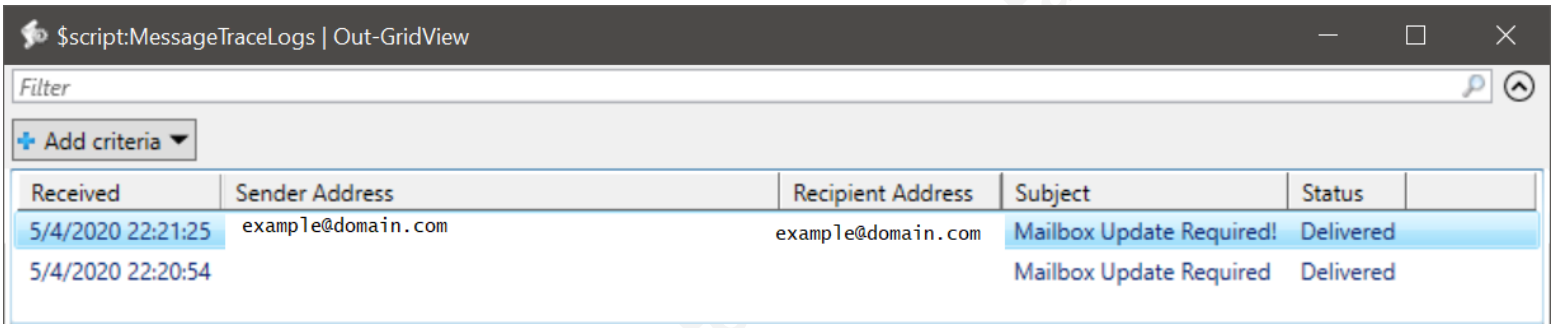

Figure 6. View of the Message Trace Information.

The analyst concludes that the account is indeed compromised by an attacker utilizing an IP of 77.243.191.27. In addition, the analyst observed that the attacker added one Inbox Rule to hide their sent mail. The analyst can easily change the user's password, revoke the AzureAD refresh token, and remove the malicious Inbox Rule using functionality on other tabs. Logs can then be downloaded to justify actions in a ticketing platform as needed.

4. Findings

4.1 Explanation of Findings

As stated above, the goal of this research and the tool is to bridge the gap between the browser-based GUI and the use of PowerShell, with the goal of making the analyst more efficient. Listed below in Figure 7 is a table depicting the total amount of user interactive actions that were taken for each feature set within the tool. This was done using Microsoft's browser-based tools during May of 2020. The author's tool was approximately 48.1% more efficient for a value of 100 SignIn Logs and 68.8% more efficient when viewing 1000 SignIn Logs. This was calculated using a formula of ((Final Value - Starting Value)/Starting Value) *100 which is used to calculate the improvement in efficiency. In addition, the author's tool performs certain functions that are either not available at all as a standalone option in the browser or are not available at all. These are

noted below and are not calculated as part of the overall efficiency improvement calculation. The author also notes the site used for each function and does not count duplicate interactive actions. In other words, if two actions use the same site, the act of browsing to the site and logging in is not counted twice. The author strived to be as judicious as possible when listing the actions, taking care to be as efficient as possible using both methods. However, it is possible, given that the author is not a specialist in O365, that there may exist a more efficient way to perform these actions that the author is not aware of despite extensive personal experience using these tools.

4.2 Interactive action measurements

Actions	Author's O365 Tool (1000 Logs)	Browser (100 Logs)	Browser (1000 Logs)	Site used
Sign Into the tool/Browser (Assuming MFA)	6	3	3	N/A
Gather User Info (Name, department, etc)	3	3	3	Portal.Azure.com
View Password Info	0	1	1	Portal.Azure.com
View Forwarding Settings	0	2	2	Admin.microsoft.com
View Licenses	0	2	2	Portal.Azure.com
View Inbox Rules	2	N/A	N/A	No way to do this via Browser GUI
Remove Inbox Rules	2	N/A	N/A	No way to do this via Browser GUI
Search SignIn Logs by Username	2	13	31	Portal.Azure.com
Search Message Trace Logs	5	9	9	Protection.Office.com
Search SignIn Logs by IP	4	16	34	Portal.Azure.com
Reset Password	2	2	2	Portal.Azure.com
Revoke AzureAD Refresh Token	1	N/A	N/A	No Standalone way to do this via Browser GUI
Download SignIn Logs by IP	2	1	1	Portal.Azure.com
Download SignIn Logs by Username	2	1	1	Portal.Azure.com
Download Message Trace Logs	2	1	1	Protection.Office.com
Total (Not including N/A)	28	54	90	3

Figure 7. Table of Interactive actions measured

4.3 Caveats in measurement

The higher amount of interactive actions required for searching the SignIn logs for Username and IP, along with searching Message Trace, are mostly due to the browser's behavior in limiting the fields and results shown in the browser. The results are limited to 50 shown for the SignIn Logs. Because of this, the numbers listed for those include one interactive action to represent the analyst loading an additional page of results. However, since there could be many results returned, the default behavior for an analyst is likely going to be to simply download the logs, rather than load the results in the browser 50 results at a time for the SignIn Logs. The numbers listed also include the actions of adding all available columns to the "columns" view so an analyst would be

able to see all information available in the browser. Message Trace functions similarly in the browser, however, there is a "Load All" button available. Again, the default behavior is not to perform log analysis via these functions, but rather to guide the analyst to download the logs for analysis in another tool such as Excel. Additionally, the author noted that there is no way as of May 2020 to perform a standalone search of the Message Trace logs using only the "Original Client IP Address" while maintaining the ability to view the results in the browser.

5. Recommendations and Implications

If available, the author recommends that organizations devote time and resources to custom tool development, especially if their security operations personnel are largely inexperienced. Since these analysts tend to rely on a GUI-based tool when a command line tool is available, custom development can bridge that gap. Joint researchers from the University of South Florida, Kansas State University and Honeywell Labs concluded in 2016 that "useful security tools for SOCs may best be built within SOCs, by people who can identify and understand the contradictions within the work environments. (Sundaramurthy, McHugh, Ou, Wesch, Bardas, Rajagopalan, 2016, p. 247).

As shown above, this tool has the potential to dramatically increase the efficiency of security operations personnel when performing certain tasks that are repeatable and well-defined. In that spirit, the author will make this tool available for download as an open source project on GitHub. He welcomes feedback and additional development from the community.

Another great benefit of using open source tools is that analysts can learn from the tool itself. "You can execute the tool, examine the output to understand the logic behind the tool's options and output, and finally examine the code that produced the output to understand the logic behind the tool's operation." (Altheide and Carvey, 2011, p. 6). The author encourages all analysts who not only use this tool, but also dissect it, and learn the cmdlets and their options. This will lead to a better understanding of how to use PowerShell to accomplish O365 DFIR tasks.

Additional research in this area could compare the addition of other PowerShell cmdlets into this tool or a similar tool, to further study how adding more actions available in both the browser and PowerShell could improve analyst efficiency.

5.1 Roadmap for features

The author hopes to add the following features to the tool as his personal time allows (with the disclaimer that this is not a commitment to any future development of any specific feature listed below):

- Dark mode
- Settings page that makes settings available to the user via the GUI
- Accessibility features (Larger print font, colorblindness settings, etc.)

These features would not impact the analysis done in this paper, but rather they are simply bonus features that the author wishes to add to the tool to enhance its usability by others.

6.Conclusion

While working with Microsoft's browser-based tools can be challenging, PowerShell can provide a much-needed efficiency boost to the analyst working an O365 Business Email Compromise Investigation. Utilizing a custom-developed tool, such as the one the author highlights here, can improve productivity of analysts with no cost to the organization. Such a tool allows less seasoned analysts to contribute to the investigative workflow with the same efficiency as a more seasoned analyst.

References

Ma, J. (2019, July 23). Staffing, Budget Among Top Challenges SOCs Face in 2019 [infographic]. Retrieved February 11, 2020, from https://www.exabeam.com/security-operations-center/state-of-the-soc-infographic/

Get-MsolUser. (2020). Retrieved May 25, 2020, from https://docs.microsoft.com/en-us/powershell/module/msonline/Get-MsolUser?view=azureadps-1.0

Get-AzureADUser. (2020). Retrieved May 25, 2020, from https://docs.microsoft.com/en-us/powershell/module/azuread/get-azureaduser?view=azureadps-2.0

Responding to a Compromised Email Account in Office 365 - Office 365. (2020, January 28). Retrieved February 10, 2020, from https://docs.microsoft.com/en-us/microsoft-365/security/office-365-security/responding-to-a-compromised-email-account

Altheide, C. A., & Carvey, H. A. (2011). *Digital Forensics with Open Source Tools*. Waltham, MA: Syngress.

Vangel, D. (2020, May 20). Automated investigation and response (AIR) - Getting Started - Office 365. Retrieved May 28, 2020, from https://docs.microsoft.com/en-us/microsoft-365/security/office-365-security/office-365-air?view=o365-worldwide#how-to-get-air

Sundaramurthy, S. C., McHugh, J., Ou, X., Wesch, M., Bardas, A. G., & Rajagopalan, S. R. (2016). Turning contradictions into innovations or: How we learned to stop whining and improve security operations. In *Twelfth Symposium on Usable Privacy and Security ({SOUPS} 2016)* (pp. 237-251).

Last Updated: July 17th, 2020

Upcoming SANS Training

Click here to view a list of all SANS Courses

SANS Amsterdam August 2020 Part 1	Amsterdam, NL	Aug 03, 2020 - Aug 08, 2020	Live Event
SANS Reboot - NOVA 2020	Arlington, VAUS	Aug 10, 2020 - Aug 15, 2020	Live Event
SANS Amsterdam August 2020 Part 2	Amsterdam, NL	Aug 17, 2020 - Aug 22, 2020	Live Event
SANS FOR508 Canberra August 2020	Canberra, AU	Aug 17, 2020 - Aug 22, 2020	Live Event
SANS Virginia Beach 2020	Virginia Beach, VAUS	Aug 30, 2020 - Sep 04, 2020	Live Event
SANS London September 2020	London, GB	Sep 07, 2020 - Sep 12, 2020	Live Event
SANS Philippines 2020	Manila, PH	Sep 07, 2020 - Sep 19, 2020	Live Event
SANS Baltimore Fall 2020	Baltimore, MDUS	Sep 08, 2020 - Sep 13, 2020	Live Event
SANS Munich September 2020	Munich, DE	Sep 14, 2020 - Sep 19, 2020	Live Event
SANS Network Security 2020	Las Vegas, NVUS	Sep 20, 2020 - Sep 25, 2020	Live Event
SANS Australia Spring 2020	, AU	Sep 21, 2020 - Oct 03, 2020	Live Event
SANS Northern VA - Reston Fall 2020	Reston, VAUS	Sep 28, 2020 - Oct 03, 2020	Live Event
SANS San Antonio Fall 2020	San Antonio, TXUS	Sep 28, 2020 - Oct 03, 2020	Live Event
SANS Amsterdam October 2020	Amsterdam, NL	Oct 05, 2020 - Oct 10, 2020	Live Event
SANS FOR500 Milan 2020 (In Italian)	Milan, IT	Oct 05, 2020 - Oct 10, 2020	Live Event
SANS London October 2020	London, GB	Oct 12, 2020 - Oct 17, 2020	Live Event
SANS Orlando 2020	Orlando, FLUS	Oct 12, 2020 - Oct 17, 2020	Live Event
SANS October Singapore 2020	Singapore, SG	Oct 12, 2020 - Oct 24, 2020	Live Event
SANS Prague October 2020	Prague, CZ	Oct 12, 2020 - Oct 17, 2020	Live Event
SANS OnDemand	OnlineUS	Anytime	Self Paced
SANS SelfStudy	Books & MP3s OnlyUS	Anytime	Self Paced