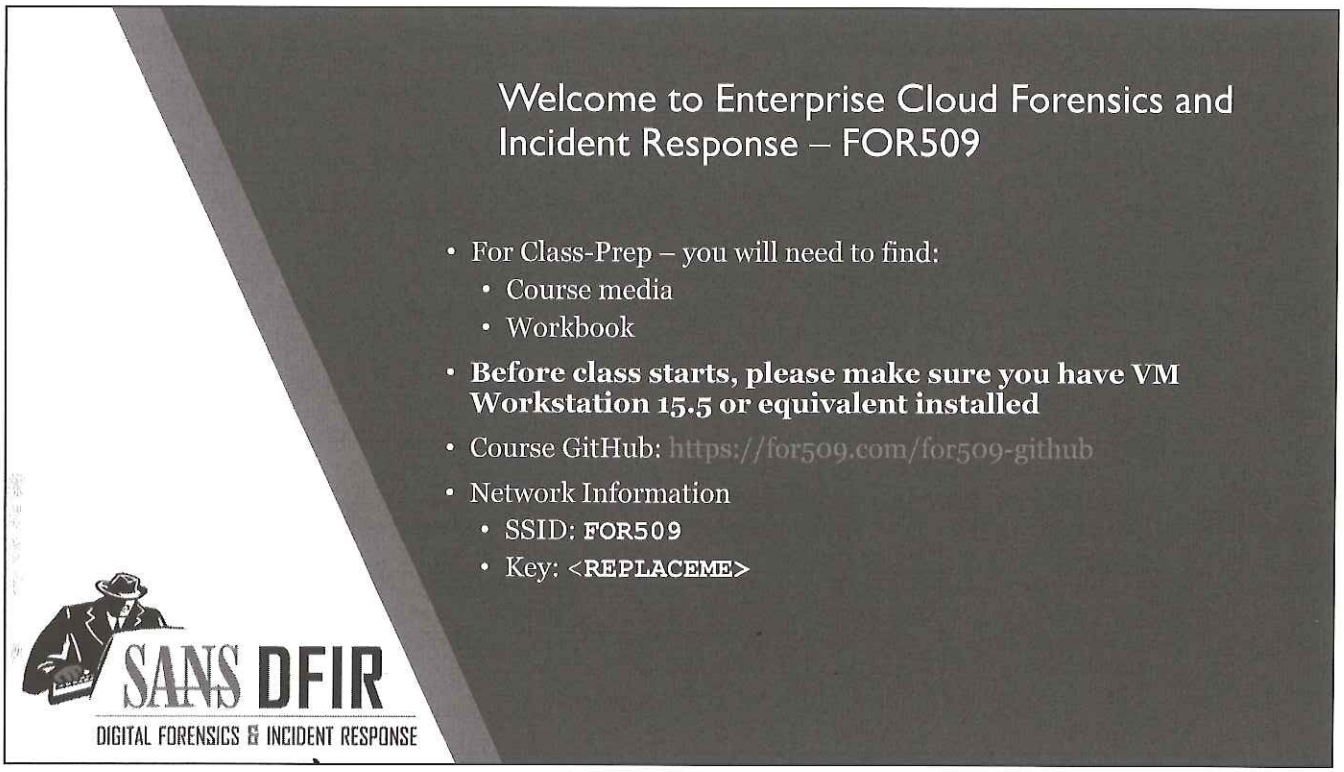
509.1
Cloud Forensic Fundamentals and Microsoft 365

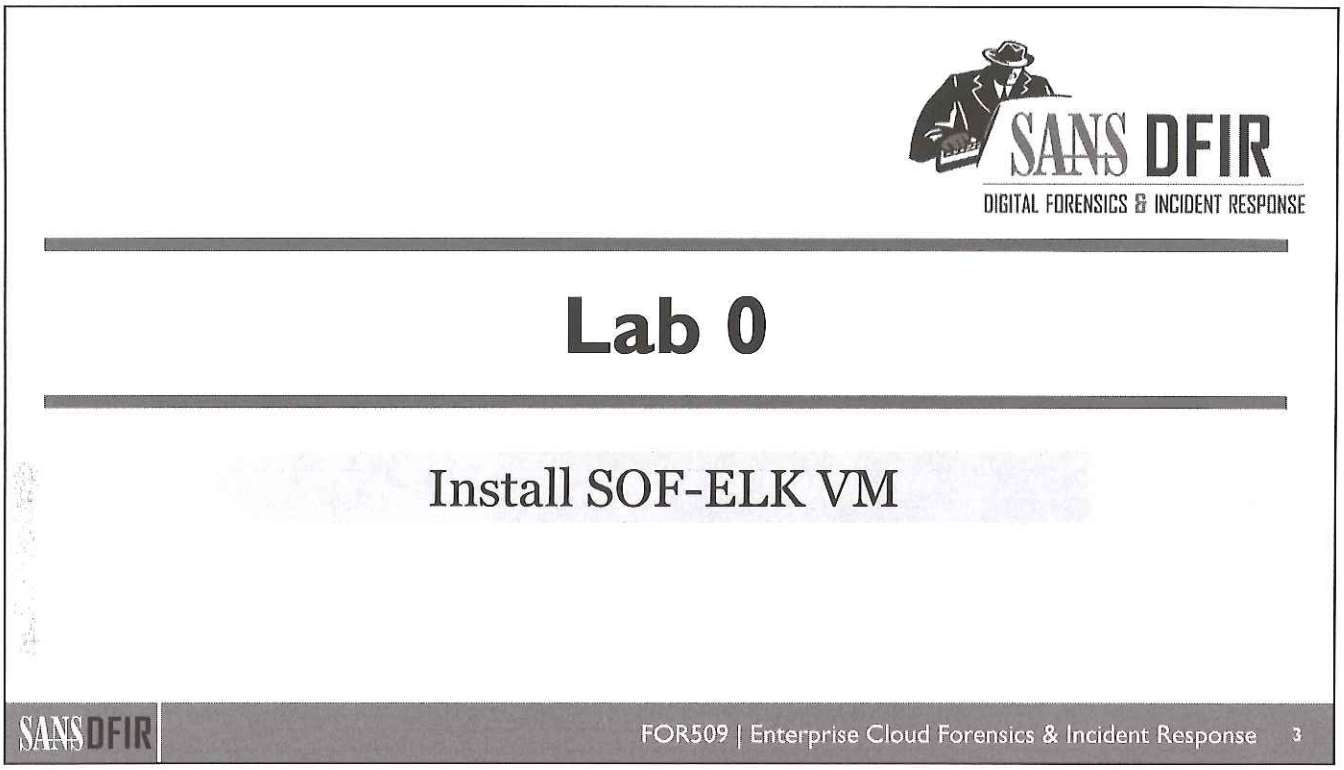
SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

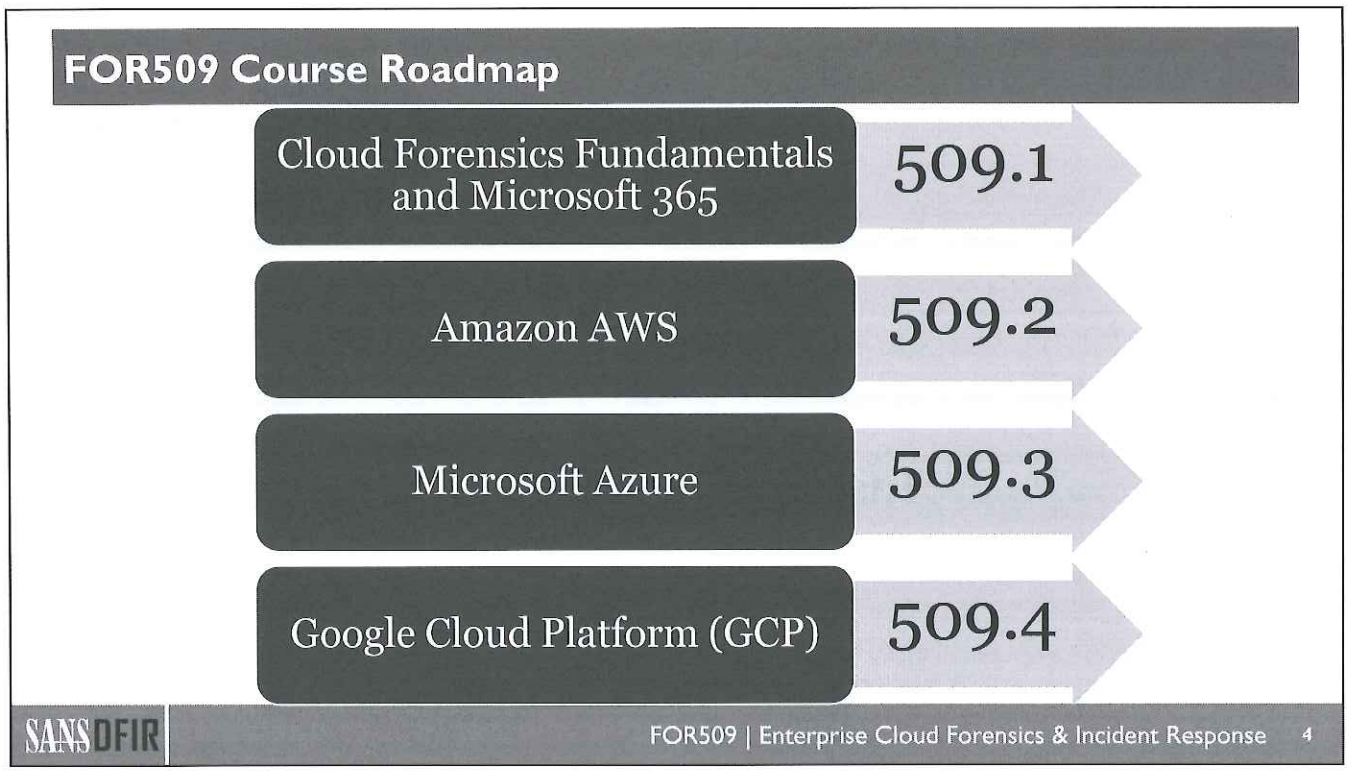


This page intentionally left blank.

This page intentionally left blank.



This page intentionally left blank.



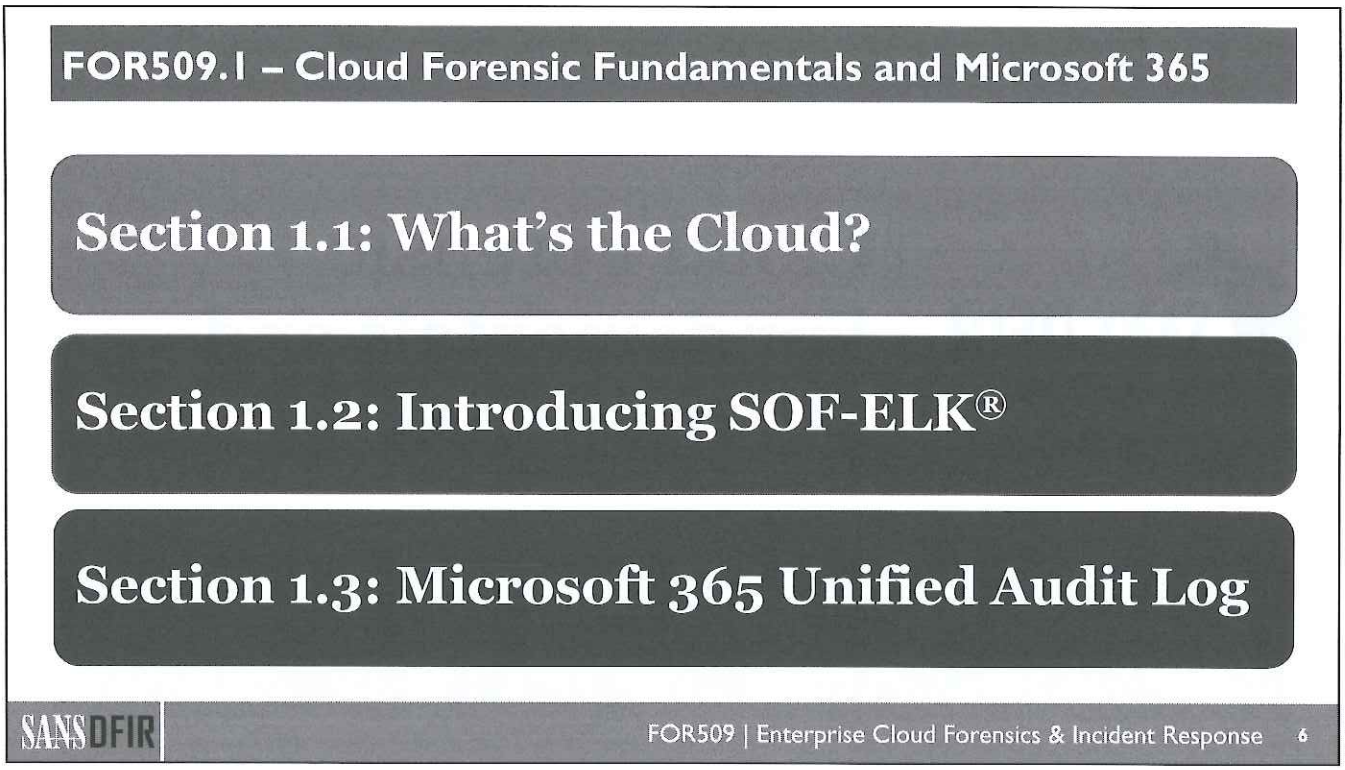
This course is made up of four parts:

1. In 509.1, we discuss the fundamentals of cloud forensics. This is an important discussion to make sure that we have a common basis for the rest of the class. We will also discuss Microsoft 365 as it's the prevalent office productivity suite in corporate environment. This will give us an opportunity to review a business email compromise (BEC) case study.
2. In 509.2, Understanding AWS, log sources, Cloudwatch, in-cloud IR
3. In 509.3, Understanding Azure, log sources, NSG flow logs, VM logs, in-cloud IR
4. In 509.4, Understanding GCP, log sources, Stackdriver, data collection agent, in-cloud IR

FOR509.1 | Enterprise Cloud Forensics and Incident Response

Cloud Forensic Fundamentals and Microsoft 365

This page intentionally left blank.



This page intentionally left blank.

Purpose of this Course

- Gain a basic understanding of key cloud resources and logs to facilitate incident response and forensics
- Become familiar with logs for virtual machines, networking, storage as well as the clouds themselves (platform logs)
- Review the different methods available to access cloud logs (AWS, Azure, GCP)
- Import logs into SOF-ELK
- Discuss various case studies

AWS, Azure and GCP are each immense ecosystems made up of hundreds of services. However, when it comes to most investigations there are a number of common elements.

In this course, we will examine the logs available for virtual machines, network, and storage. In addition, we will review higher level logs (sometimes called platform logs) to understand who is creating these cloud resources and who is accessing them.

Identity and Access Management (IAM) is key to access any cloud resource. Each cloud offers its own version of IAM. You will find that many large enterprises operate in a hybrid environment: on-prem systems plus a mixture of various clouds.

One popular model for these companies is to leverage Azure Active Directory as their identity management system and control authorization with each cloud's own IAM system. The benefit is that employees of these companies only need to remember one set of login credentials no matter which cloud they access. From a DFIR point of view, this will represent some interesting challenges as you may be investigating an incident in GCP, but the authentication took place in Azure.

By the end of this course, you will have a strong understanding of the logging capabilities of the main 3 cloud providers. This will provide you a solid foundation to conduct incident response and forensics in your environment.

The elements (virtual machines, network, storage) we are covering are the ones the authors have found to be present in nearly all investigations. The course will discuss many scenarios based on real attacks including:

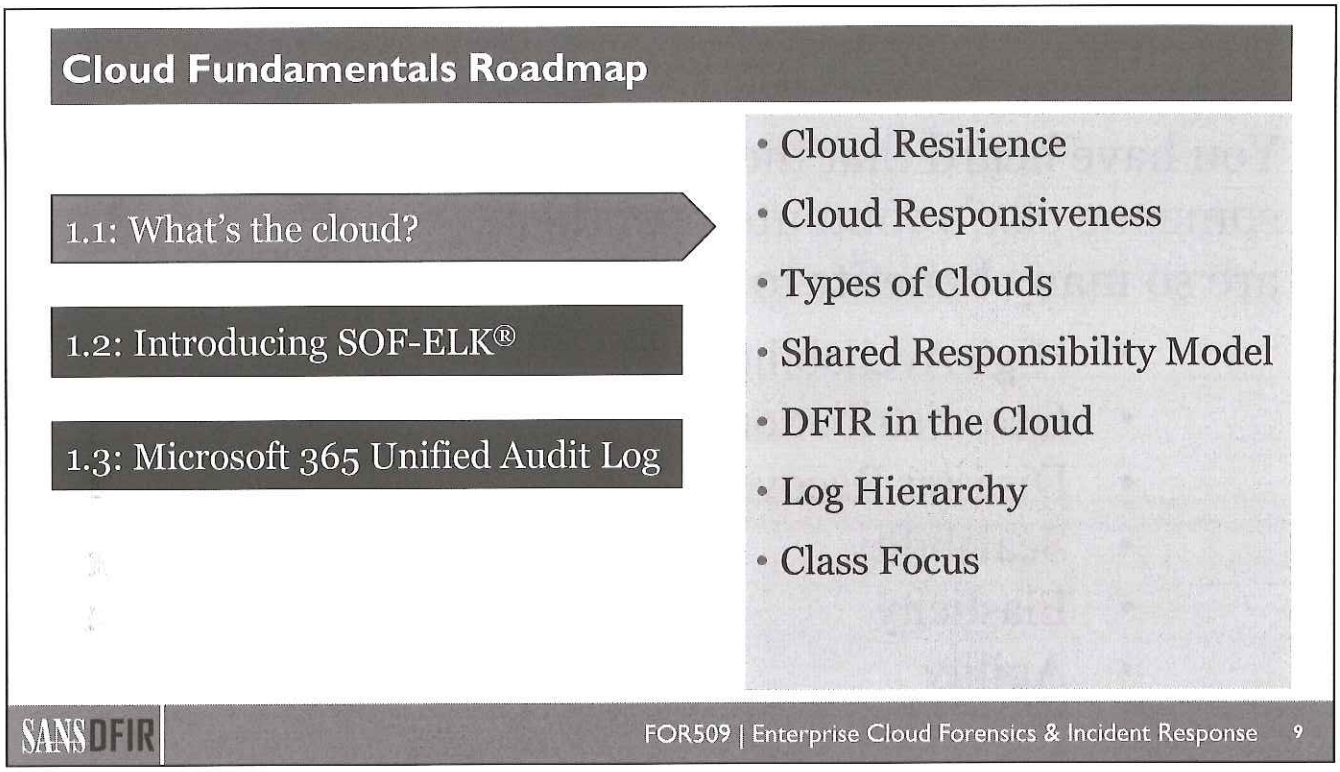
- A successful password spray attack
- Creation of new virtual machines for the purpose of crypto mining
- Exfiltration of data

While you may have conducted similar investigations within your on-prem environment, you will find that the cloud has a few unique challenges. One of the biggest challenges is that many logs are turned off by default, making investigations extremely difficult.

By knowing which logs are key to your investigations, you will be able to go back to your environment and make sure that all the necessary resources are configured and available to you.

References:

[1] LogicMonitor, "Cloud Vision 2020: The Future of the Cloud," www.scribd.com/document/403188911/LogicMonitor-Cloud-2020-The-Future-of-the-Cloud-pdf, page 3 [Subscription required]

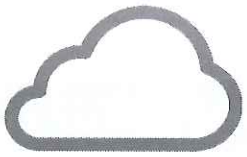


This page intentionally left blank.

The Cloud

You have heard that the cloud is someone else's computer, yet there are so many benefits to the cloud:

- High Availability
- Geo-distribution
- Disaster Recovery
- Scalability
- Elasticity
- Agility



There is no cloud

It's just someone else's computer

DFIR BENEFIT

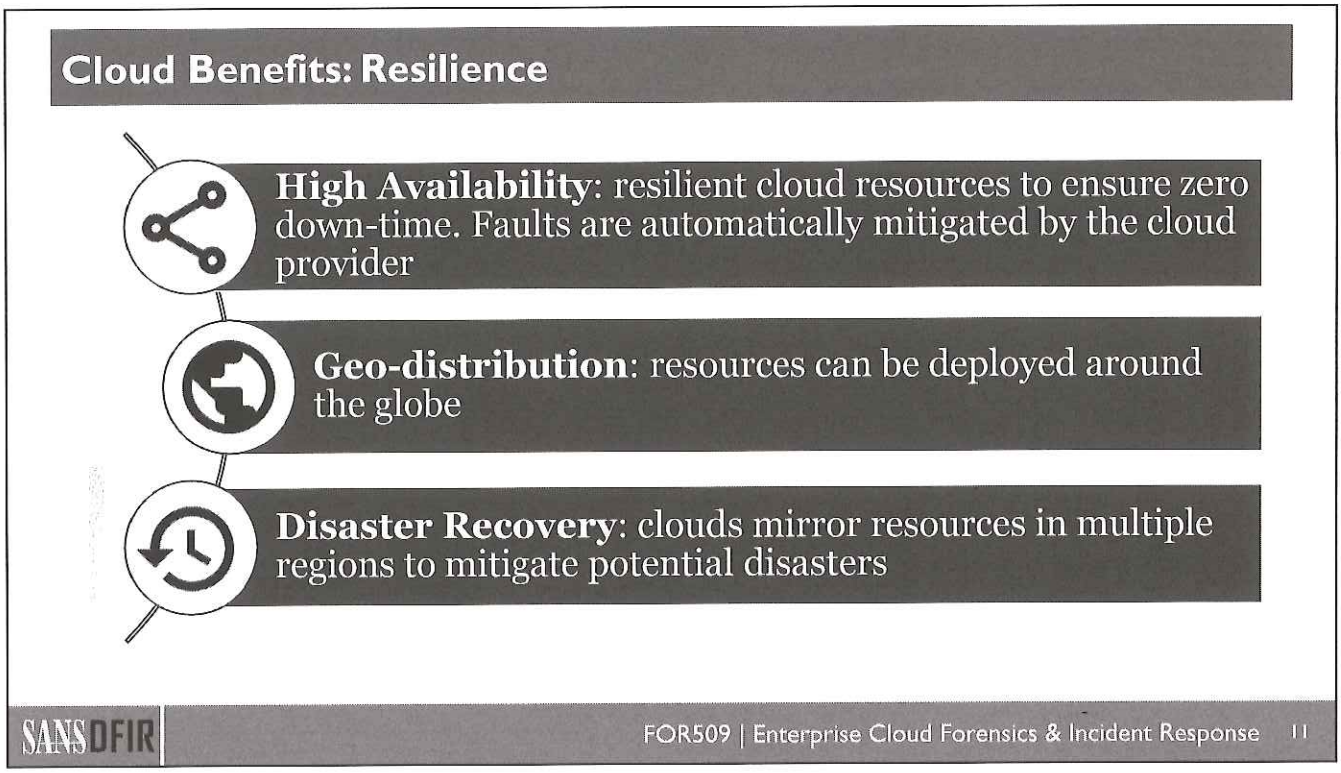
You don't have to care for and feed the hardware. Your DFIR workstation will work when you need it to!

The saying that "there is no cloud, it's just someone else's computer" is very misleading. The cloud has so many benefits that it's hard to list them all. The two main categories are resilience and responsiveness.

Resilience is made up of three attributes: high availability, geo-distribution and disaster recovery.

Responsiveness also has three attributes: scalability, elasticity and agility.

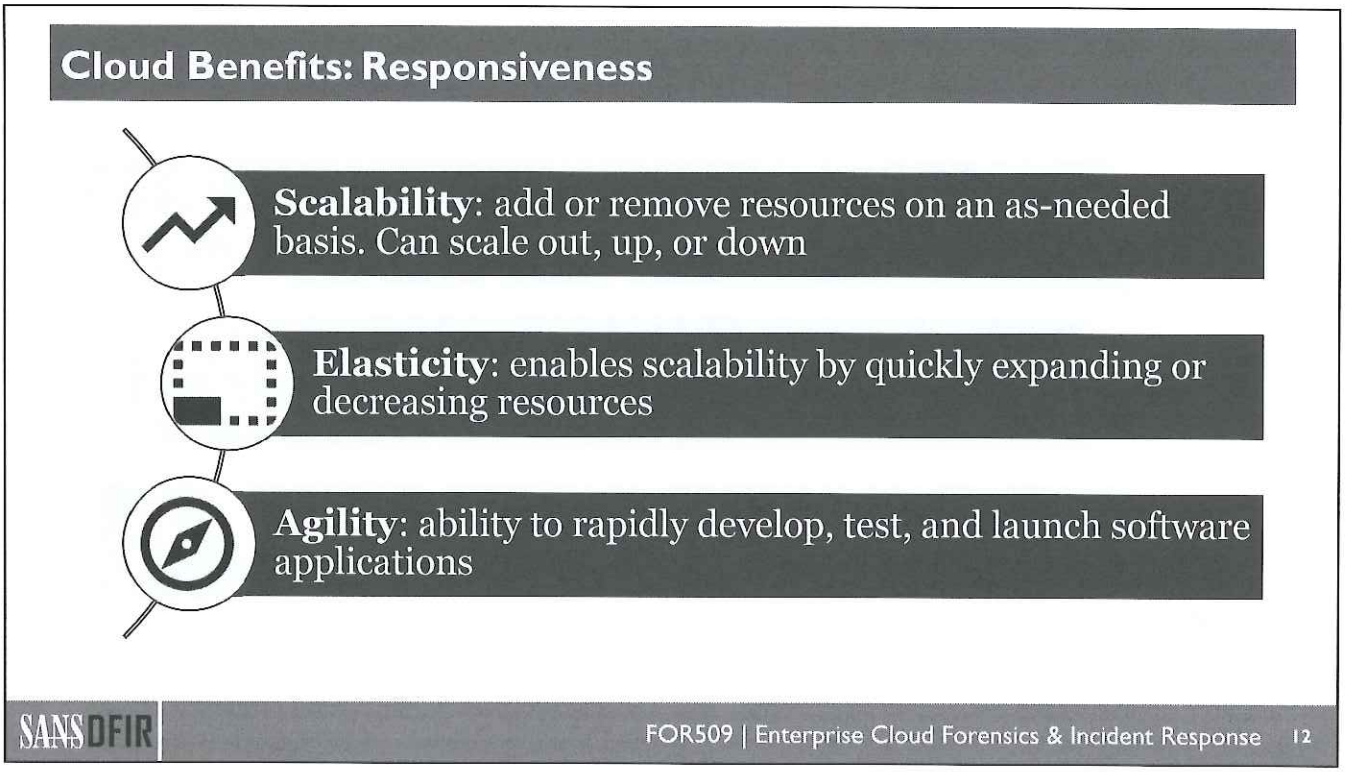
We will discuss these attributes in the next slides.



When we say that the cloud is resilient, we are talking about the availability of the resources to accomplish your business objectives. That availability goes beyond the uptime and includes the ability to instantly replace failed machines. This is typically handled by the cloud provider in the background with no apparent downtime to the user.

In addition, cloud providers offer numerous choices in geographies giving you the ability to deploy your services around the globe. This flexibility ensures the best performance for your users. It also enables you to meet in-country legal requirements for certain jurisdictions.

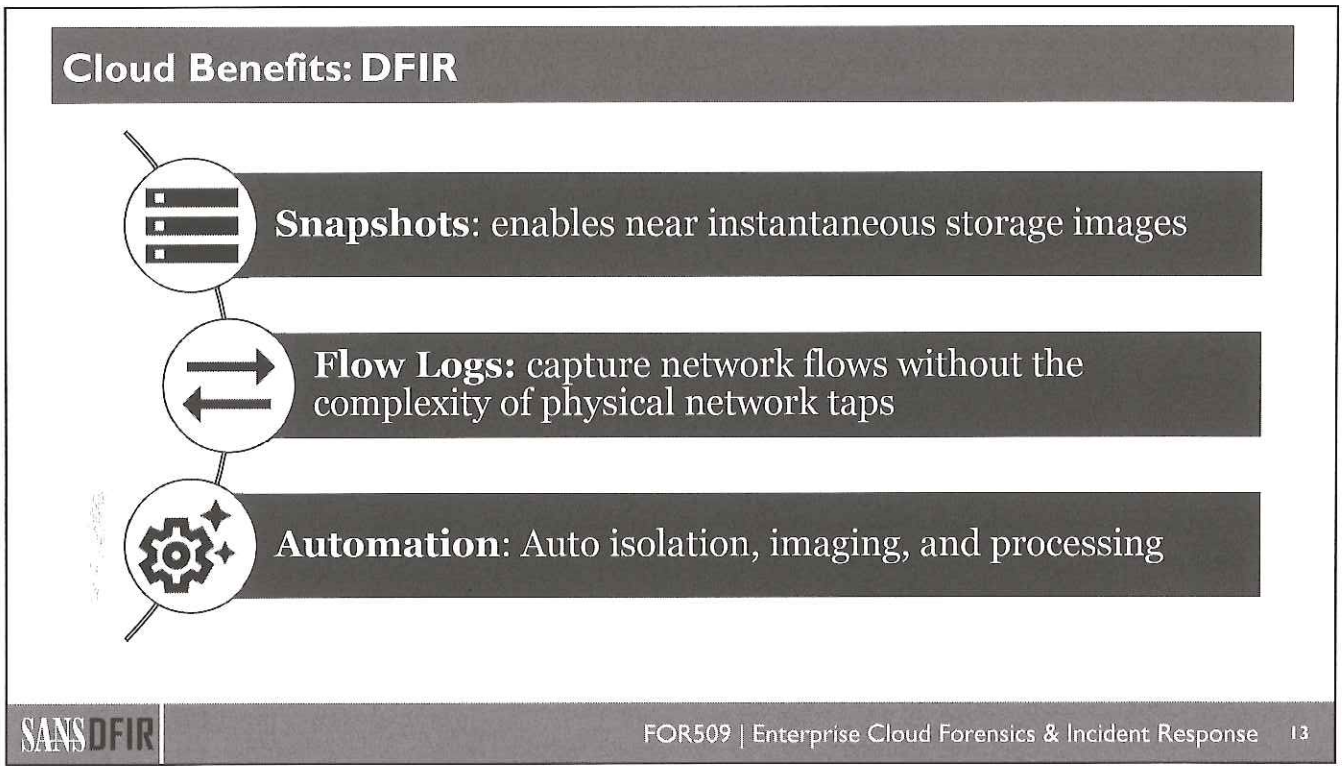
Finally, when these features are used in conjunction with various data replication features, you can eliminate single points of failure that could impact your services in the event of a disaster.



The second broad category of benefits is responsiveness. One of the amazing features of cloud computing is the ability to scale vertically and horizontally. Vertical scaling is the ability to increase compute capacity by adding RAM or CPUs to a virtual machine. Horizontal scaling is the ability to increase capacity by adding resources such as virtual machines and storage.

This scalability can be short term or long term. For example, if you need to process terabytes of data for your investigation, you can add a dozen high end virtual machines for a few days and shut them down when you work is complete. You only pay for the time you used these resources. This example can be taken further by having an application automatically add and remove virtual machines based on pre-determined factors. This feature is called elasticity.

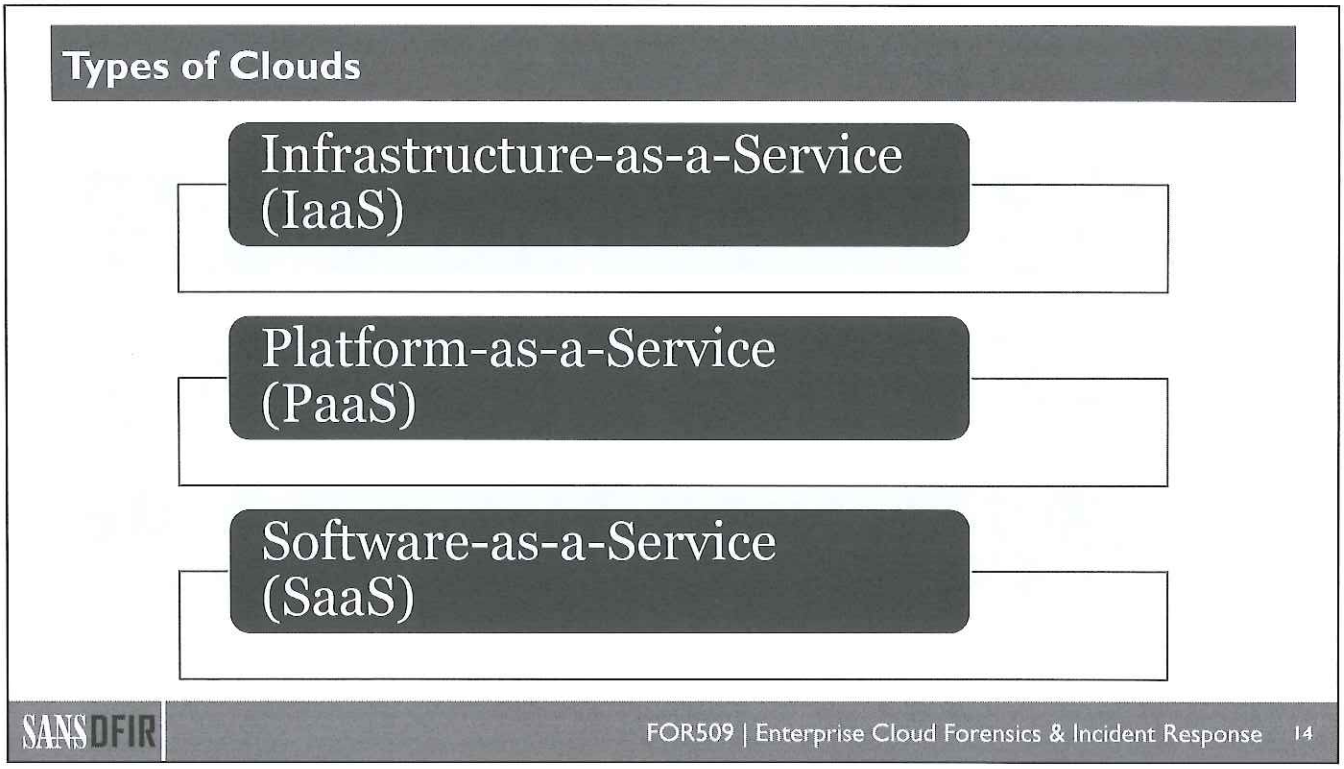
Finally, the cloud demonstrates agility by allowing you to deploy resources for as short of a period as you wish. This is particularly useful when running tests and avoids having to commit to long term contracts.



When it comes to Digital Forensics and Incident Response, the cloud offers some great benefits. The first one is snapshots. With snapshots, you are able to get an entire drive imaged nearly instantaneously. This is such an amazing time saver compared to disk duplicators and write blockers. No need to run to the computer store and hunt for a few precious hard drives to get your work done on time.

Network logs contain a wealth of information. However, to get them, the network team needs to install a network tap and setup capture software. This is something of the past in the cloud. You can simply configure your virtual network to capture network flows for you. If your environment implements a more advanced virtual firewall, you may even be able to get full packet capture.

Finally, the cloud gives us the opportunity for automation. Since we no longer need to handle any hardware, we can script these DFIR tasks. There is a wealth of scripting languages available in addition to serverless computing which can be triggered to perform any task you wish.



There are three broad models for the cloud. They are always called <Something>-as-a-Service. The "as-a-Service" terminology is frequently abused by marketing teams and there are now an endless list of things-as-a-service being advertised. But, when it comes to the cloud, you need to know about: IaaS, PaaS and SaaS.

The number one commonality between the three cloud models is that you don't have to worry about anything physical. That's right, no more 2AM calls that the power went out and your generator didn't fire up! The care and feeding of the hardware is entirely the responsibility of the cloud provider.

- IaaS: the lowest level of service. In other words, you are not responsible for the hardware, but everything else IS your responsibility. This is the closest to running your own datacenter in the cloud. It's very popular with enterprises under the "lift-and-shift" model.
- PaaS: in this model, the cloud provider manages the machines and the operating system. You are only responsible for managing the application(s).
- SaaS: everything (HW, VM, Apps) is managed by the cloud provider and you are only responsible for providing the data.

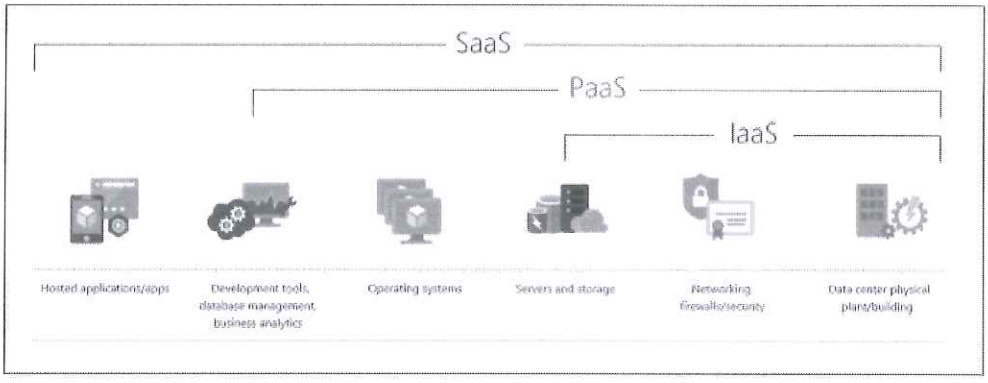
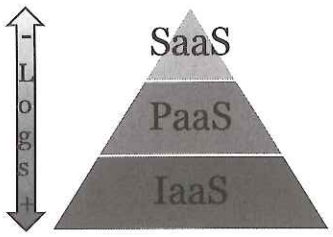


Image courtesy of Microsoft: https://for509.com/cloudtypes

Infrastructure-as-a-Service (IaaS)

Cloud provider hosts physical hardware (building, power, cooling) and makes virtualized resources available to customers: Virtual machines, networking, storage for example.



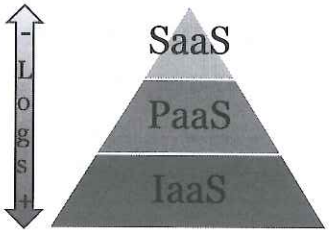
WHAT DOES IT MEAN FOR DFIR?

- Logs available all the way to the operating system level
- Anything running within the infrastructure is the customers forensic responsibility
- Customer is responsible for enabling and storing the logs
- Resources can come and go very quickly (scale up and down)
- Everything has a unit cost

When it comes to having control over your environment, IaaS is the best solution for investigators. As long as logs have been enabled, you will have the most amount of data under this cloud model. The biggest challenge in this model is that since everything has a unit cost, the business may make the financial decision to not enable logs. Logs in of themselves don't cost anything, but storing these logs can quickly get very expensive in large environments. It's critical that you work with senior management to implement tenant wide policies of mandatory logs.

Platform-as-a-Service (PaaS)

Cloud provider includes everything from IaaS plus application development tools such as business intelligence, database management, etc. Customer manages the applications.



WHAT DOES IT MEAN FOR DFIR?

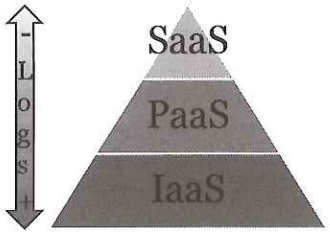
- Platform logs will be available at the discretion of the cloud provider
- Request app developer to implement logs for their application
- Additional logs may be available based on where the authentication and authorization are performed

Under the PaaS model, you are very much at the mercy of the application developers. Providing security input early in the application development lifecycle will be key.

When dealing with an existing application, in the absence of application logs, you will want to look for authentication logs as well as network logs.

Software-as-a-Service (SaaS)

Cloud provider provides a fully managed application. You pay an access fee to use but have no responsibility for the operation and maintenance. Examples: Microsoft 365, Google Workspace, Salesforce, ServiceNow.



WHAT DOES IT MEAN FOR DFIR?

- Logs are entirely at the discretion of the provider
- Access to logs should be part of the contract negotiation before using the service
- Service tier level may determine extent of logging

Under the SaaS model, logs will be at the discretion of the provider. That being said, it would be very surprising for a commercial application to not provide some type of logs. Both Microsoft 365 and Google Workspace provide extensive logs.

One caveat is that depending on the service level purchased, the extent of the logs may vary. For instance the Microsoft E5 license provides significantly more logs than the E3 license. Also, the retention of these logs may vary based on the service level.

The other challenge with logs under the SaaS model is the ability to consume them. You will want to either import these logs to your SIEM platform and/or access them via API. Be sure to obtain the log schema from the SaaS provider so you can effectively use the information provided in these logs.

Serverless & Containers

Not technically a cloud type, but extremely popular. You just provide code to execute, and the cloud provider takes care of everything else. Serverless examples: Azure Functions, AWS Lambda, Google Cloud Functions. Containers example: Kubernetes, Docker.

WHAT DOES IT MEAN FOR DFIR?

- Systems may exists for minutes, hours or days
- All log data may be purged on container exit
- Specialized tooling and configurations required for full visibility

Serverless and containers are a hot topic. Containers are a great way to sandbox an application while serverless is an efficient way to run a small amount of code. From a DFIR perspective they both represent challenges. Containers are likely to purge all log data on exit, while serverless typically run for a very short amount of time leaving very little in terms of logs.

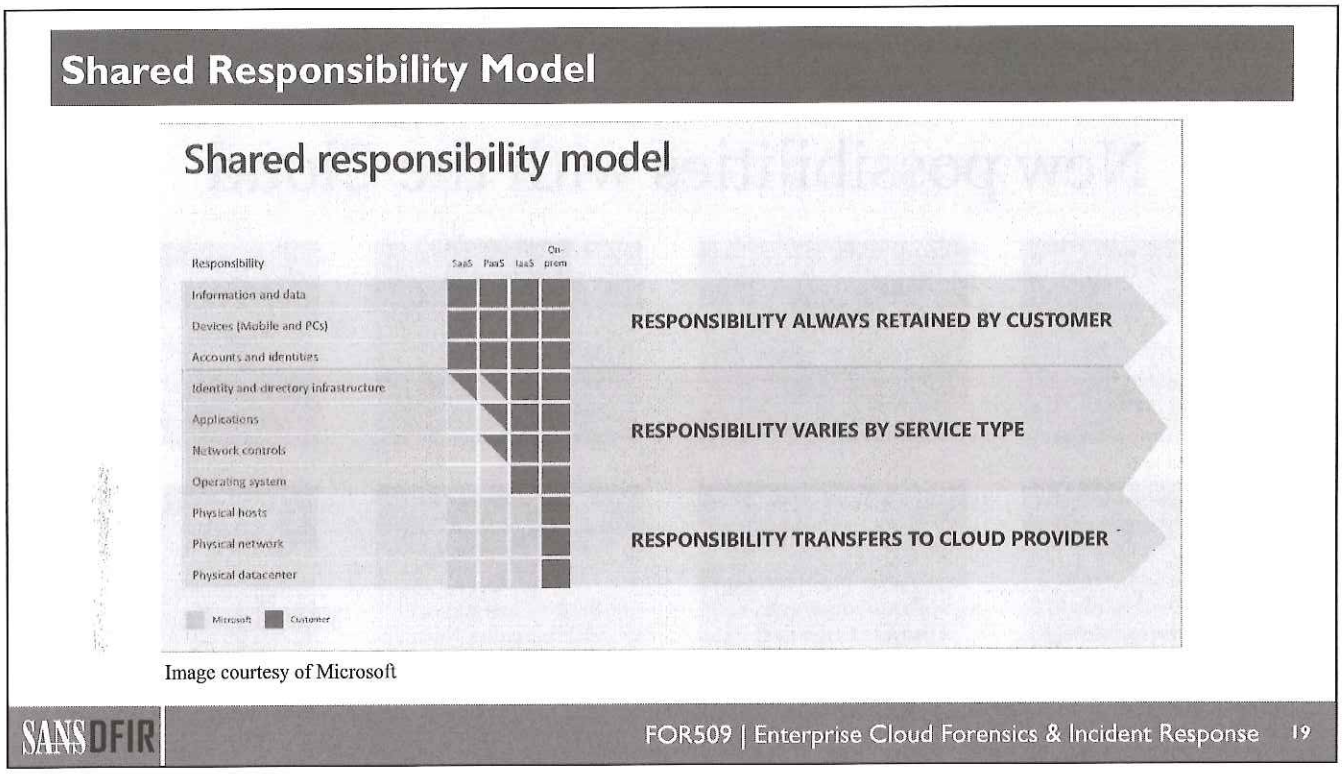


Image courtesy of Microsoft

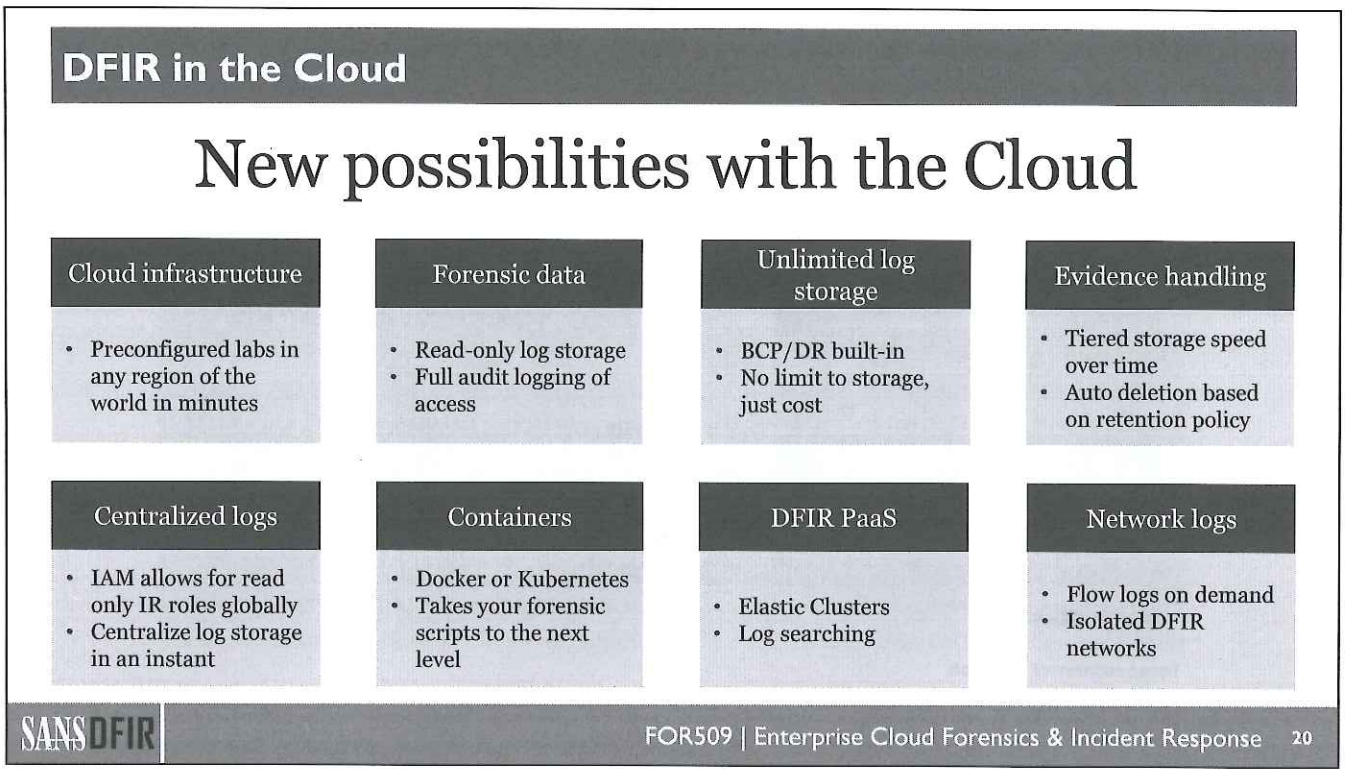
Each cloud vendor has a shared responsibility model explaining what they will take care of versus what the customer is expected to handle. This slide shows the Microsoft shared responsibility model[1] as an example.

As discussed in the prior slides, from a DFIR perspective, the question will always be what logs are available for us to conduct our investigation. Should we expect the cloud vendor to provide these logs? Is it our company's responsibility to have enabled these logs?

The more you can address these questions up front, the better prepared you will be when faced with an investigation.

References:

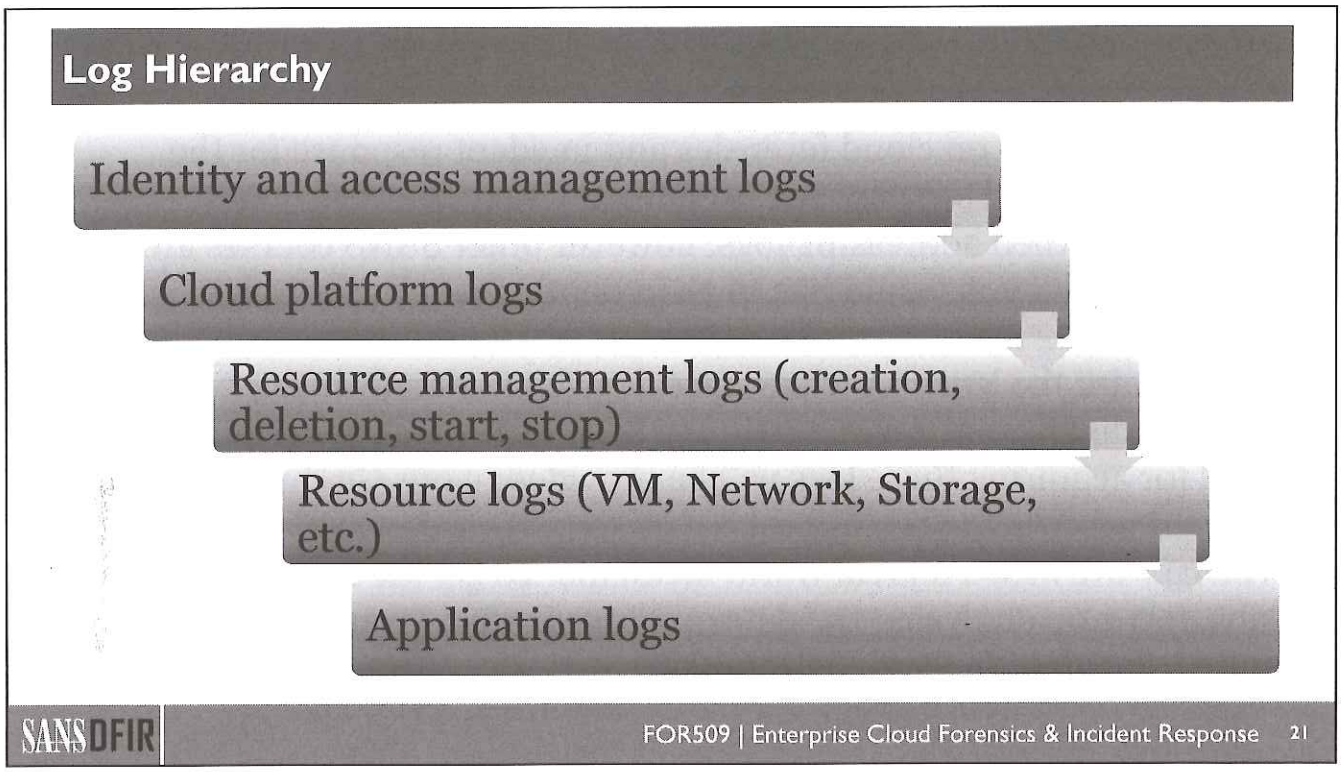
[1] https://for509.com/model



When it comes to DFIR, there are so many new possibilities with the cloud. Here are just a few examples:

- **Cloud infrastructure**: you can build a DFIR lab in any region offered by the cloud provider without leaving your office. Without the cloud, the mere logistics of acquiring physical hardware in certain countries was an insurmountable obstacle.
- **Forensic data**: you are no longer limited to network and endpoint logs. These logs can also be stored in read-only storage to meet specific regulatory requirements.
- **Unlimited log storage**: no need to ask for capital dollars to purchase more disk drives. You can store as much as you want in the cloud and only pay for what you use.
- **Evidence handling**: as your investigation progresses, you can move your evidence to slower and cheaper storage. You can also implement a retention policy to clean up older data.
- **Centralized logs** makes it so much easier to conduct a global investigation.
- **Containers** allow you to automate forensic investigations by scripting repetitive tasks.
- **DFIR PaaS** offers the opportunity to use a fully hosted elastic instance where you can upload your data and immediately start your investigation.
- **Network logs** no longer require hardware taps and sniffers. Virtual networks can easily be configured to provide flow logs.

These are just a few of the benefits and you will surely discover many more as you conduct your investigations in the cloud.



There are a huge number of log sources contained in the cloud. When starting an investigation, you will want to start from the most general log and then work your way down.

We recommend starting from the Identity and Access Management log. This will help you determine if credentials were possibly compromised.

The next set of logs to examine have different names depending on the cloud provider. As a generic name, we call them platform logs. These are logs at the top of the cloud organization: organization root in AWS, tenant in Azure and organization in GCP. A compromise at this level is usually very bad news. That being said, in a well configured cloud, there should be less than a handful of accounts with permissions to make changes at this level. These few high privileged accounts should be highly protected with complex passwords and multi-factor authentication.

It's now time to look at the resource management logs. These are logs that will tell you which resources were created, deleted, started and stopped. These are some of the most important logs to examine if you believe that a cloud account has been compromised. A cloud account could be either a user account or a service account. These logs are at the sub-organization or regular tenant level in AWS, subscription level in Azure and project level in GCP.

In many cases, the attackers will directly compromise a resource such as a VM or storage account. All cloud providers have logs that will help you track down such compromise. Don't forget that most clouds consider networking a resource and this is where you will find network logs. In AWS and GCP, these are called VPC flow logs. In Azure they are called NSG flow logs.

Finally, at the developer's discretion, applications running within cloud resources may or may not have logs. This is something you will need to research for each application. Applications provided by the cloud provider will normally generate logs.

Class Focus

FOR509 focuses on cloud forensics and incident response for the enterprise. Cloud security and architecture may be occasionally mentioned when addressing logs, however other SANS courses are better suited if you would like to go in depth about these topics.

To optimize the learning experience, logs have already been downloaded from the cloud onto the SOF-ELK VM. The scripts we used for acquiring these logs will be provided in our GitHub repository.

The labs are based on real world incidents. To get the most out of the labs, focus on analyzing the data and imagine how you would work a similar incident in your environment.

This page intentionally left blank.

Why Are We Not Using the Cloud Directly?

Incident response and forensics is primarily about following breadcrumbs left behind by attackers. These breadcrumbs are mostly found in logs. Your knowledge of the investigation process is far more important than the mechanics of acquiring the logs. As such, the authors decided to create a class that doesn't need direct access to the cloud as this access would encounter issues such as:

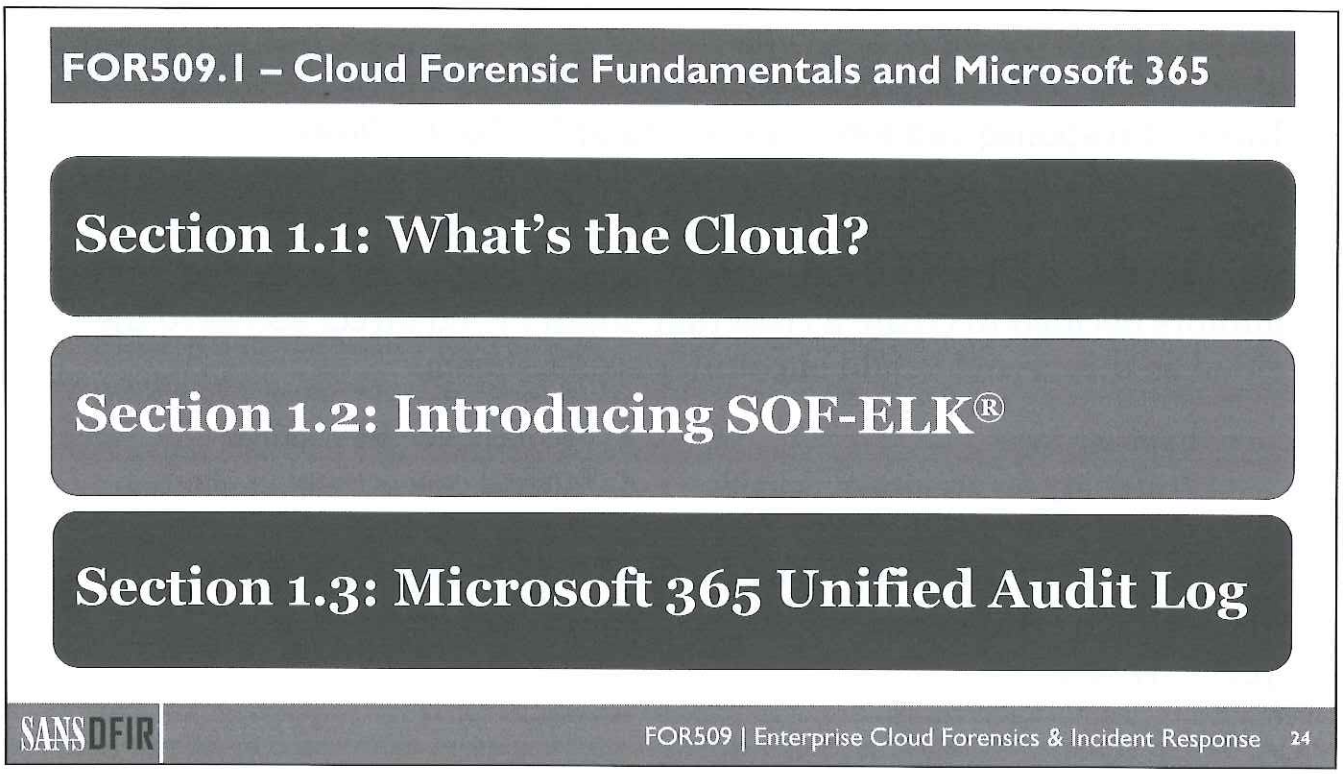
- Expiring logs
- Possibility for accidental deletion
- Frequent vendor changes
- Internet connectivity limitations

We believe that by focusing on the logs and the investigation process, you will gain valuable insights that you can apply immediately in your environment.

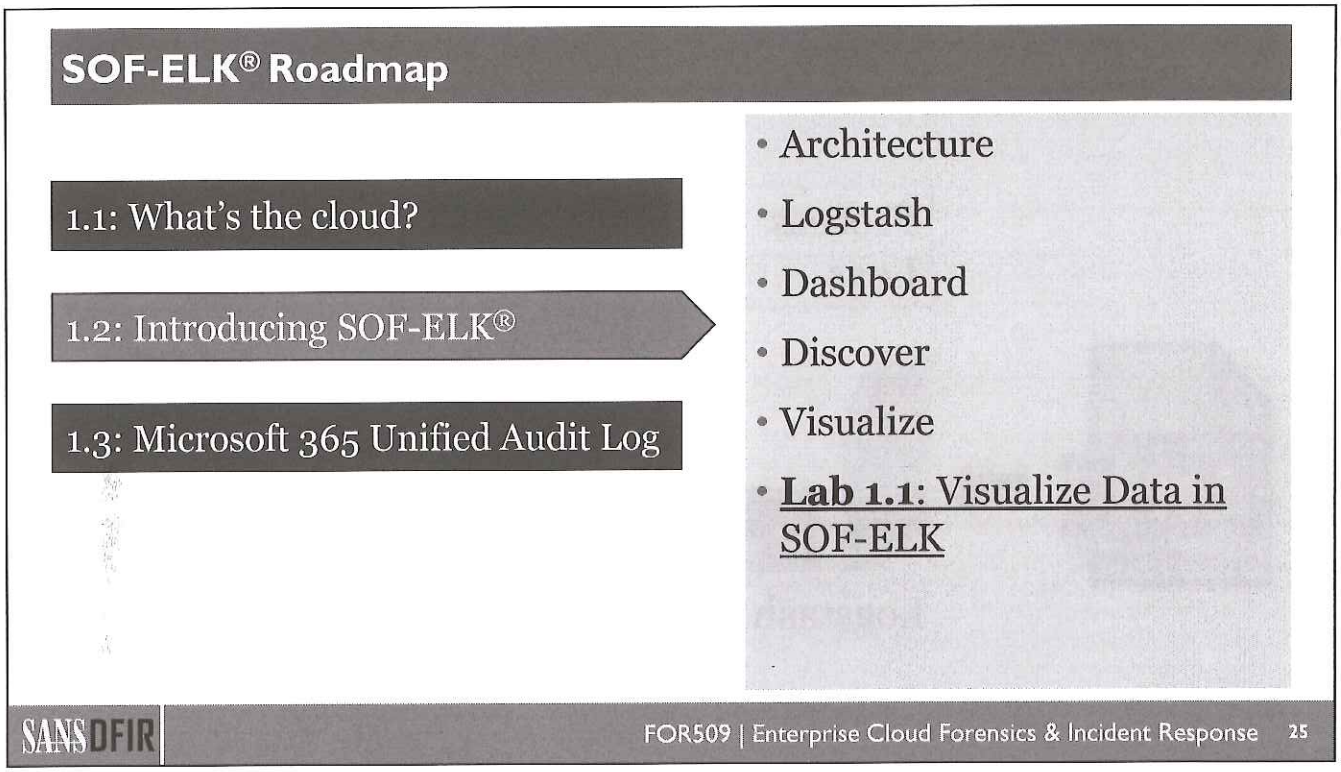
The authors had to make a difficult decision when writing this class. As technical folks, we love to have our hands on the keyboard. However, this class is about incident response and forensics and we quickly realized that the mechanics of the cloud were less important than the thought process we use during our investigations.

Next to the thought process, the most important aspect of our investigations are the source of logs. It quickly became very clear to us that cloud providers offer many log sources but they are not in a single location. So it's critical that you are aware of these log sources and the information that can be obtained from them.

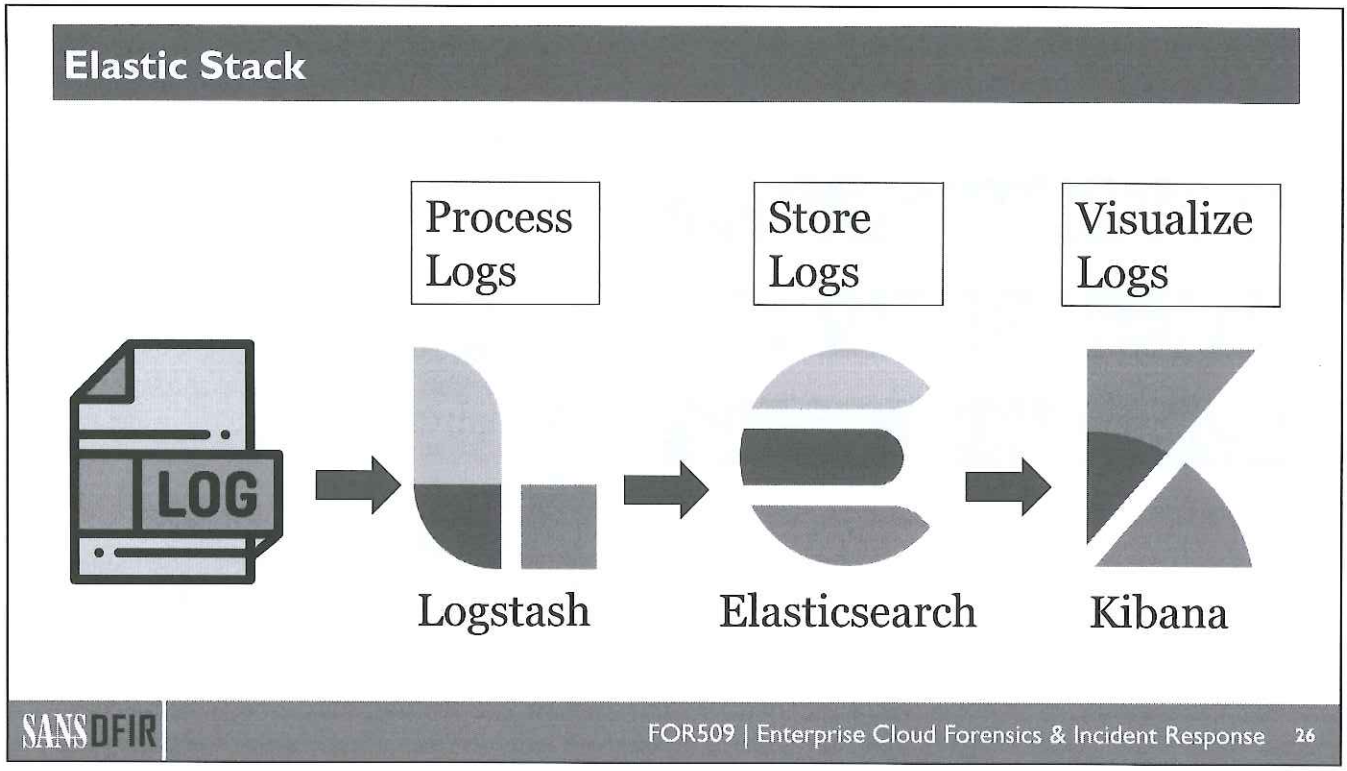
In the end, we made the decision to download all relevant logs onto the SOF-ELK VM so we can maximize our class time analyzing logs.



This page intentionally left blank.



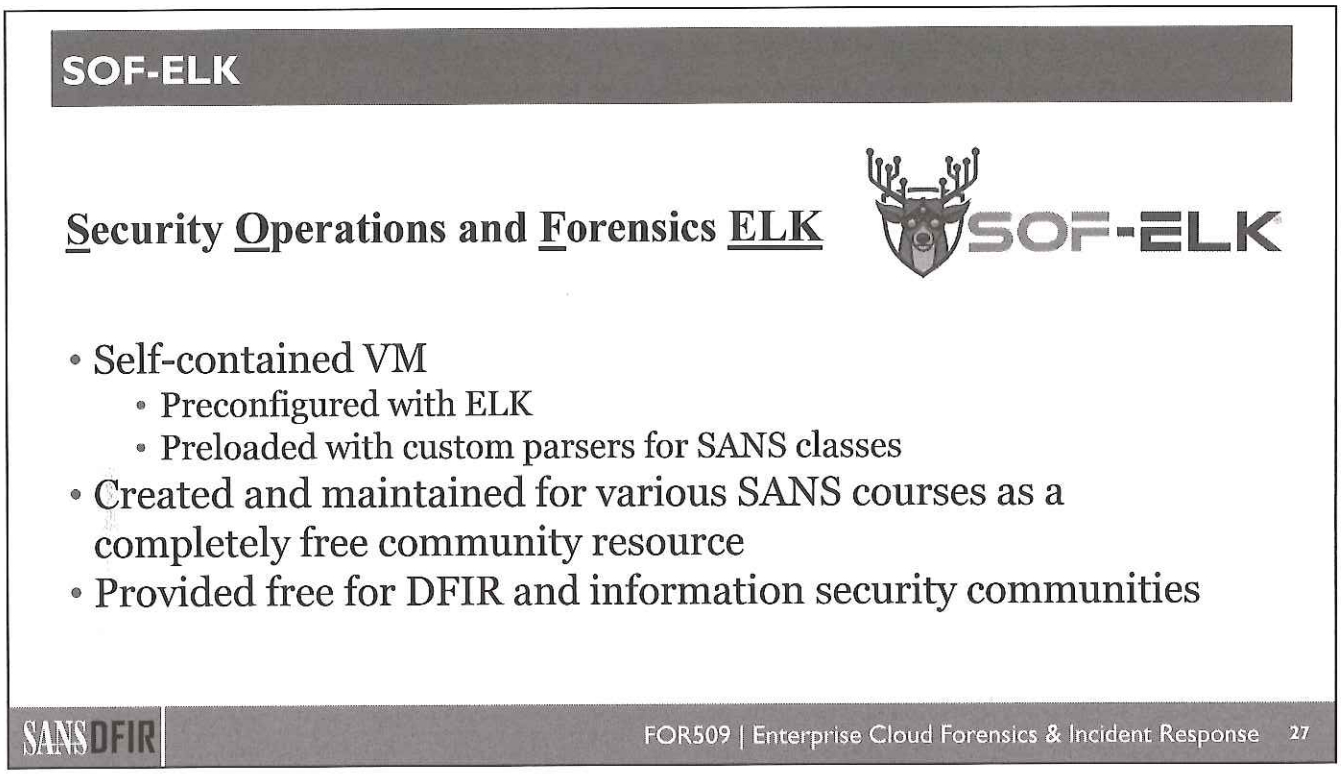
This page intentionally left blank.



The Elastic Stack (formerly known as ELK Stack and simply abbreviated ELK) is made up of three open-source projects: Elasticsearch, Logstash and Kibana:

- **Elasticsearch** is a document-centric storage and analytic engine where the data is actually stored. It features fast and scalable search functionality.
- **Logstash** is a data collection and log parsing engine. It reads input data, transforms and enriches it. The enriched data is then transported to one of more destination (such as Elasticsearch).
- **Kibana** is the web-based frontend that allows users to explore data through dashboards and visualizations.

In addition, there is a log shipper called "Beats" which facilitates sending data from a large number of machines to Logstash & Elasticsearch. It facilitates sending various types of data (such as raw log files, windows event logs, Linux syslog, etc.) into the ELK processing pipeline.



Building onto the existing SOF-ELK platform, numerous additional Logstash parsers were integrated to support AWS, Azure, GCP, and Microsoft 365 data. You'll be using a custom version of SOF-ELK that has been specifically built for FOR509, including class lab data and the Electronic Workbook. You must use the version distributed with your courseware for the labs, but the overall cloud functionality is also included in the public SOF-ELK release.[1]

If you would like to learn more about SOF-ELK, we highly recommend listening to Phil Hagen's talk on that subject.[2]

Additionally, SANS instructor John Hubbard gave a talk at the Philadelphia Security Shell meetup called "How to Use The Elastic Stack as a SIEM".[3]

References:

[1] https://for509.com/sof-elk

[2] https://for509.com/sof-elk-talk

[3] https://for509.com/elk-siem

Logstash

- Logstash is the key to SOF-ELK's ability to parse and ingest many log sources
- Logstash parsers written for FOR509: AWS, Azure, GCP, Microsoft 365
- Copy log to appropriate directory and logs are "magically" imported into SOF-ELK

Log	Directory
AWS	/logstash/aws
Azure	/logstash/azure
GCP	/logstash/gcp
Microsoft 365	/logstash/office365
Flow Logs (any clouds)	/logstash/nfarch

Flow logs require an additional step (see notes)

To support FOR509, a number of Logstash parsers were written:

- **AWS**: CloudTrail logs (includes a pre-processing ingest script)
- **Azure**: Tenant, subscription and resource logs – exported from storage account in JSON format
- **GCP**: Google Logging exports or using Pub/Sub which will be discussed in the GCP section
- **Microsoft 365**: Unified audit log – either exported from the portal or PowerShell, must be CSV formatted
- **Flow logs**: VPC flow logs from AWS or GCP. NSG flow logs from Azure. Requires additional step described below

SOF-ELK also has Logstash parsers for syslog, HTTPD logs, Passive DNS logs and Zeek logs (not used in FOR509).

SOF-ELK runs a filebeat process (part of the Beats log shippers mentioned earlier) that's continuously looking for changes in the directories mentioned in the slide. As soon as a file is copied in one of these directories, filebeat will grab it and send it to the appropriate Logstash parser.

Flow logs require a bit of additional massaging. The raw flow logs exported from AWS or Azure are in JSON format and need to be converted in a format that SOF-ELK can read with its NetFlow ingest feature. Run the appropriate ingest script as follows:

AWS: `$ aws-vpcflow2sof-elk.sh -r /path/to/aws/flow/log -w /logstash/nfarch/aws_flow_log.txt`

Azure: `$ azure-vpcflow2sof-elk.sh -r /path/to/aws/flow/log -w /logstash/nfarch/aws_flow_log.txt`

GCP: (logs are natively parsed, there is no ingest script needed)

If you want to know what the Logstash parsers do, look in the directory `/user/local/sof-elk/configfiles`

You will see different "levels" of files: input, preprocess, postprocess, output and the main files. The key files for FOR509 are:

- 6701-office365.conf
- 6801-azure.conf
- 6901-aws.conf
- 6950-gcp.conf

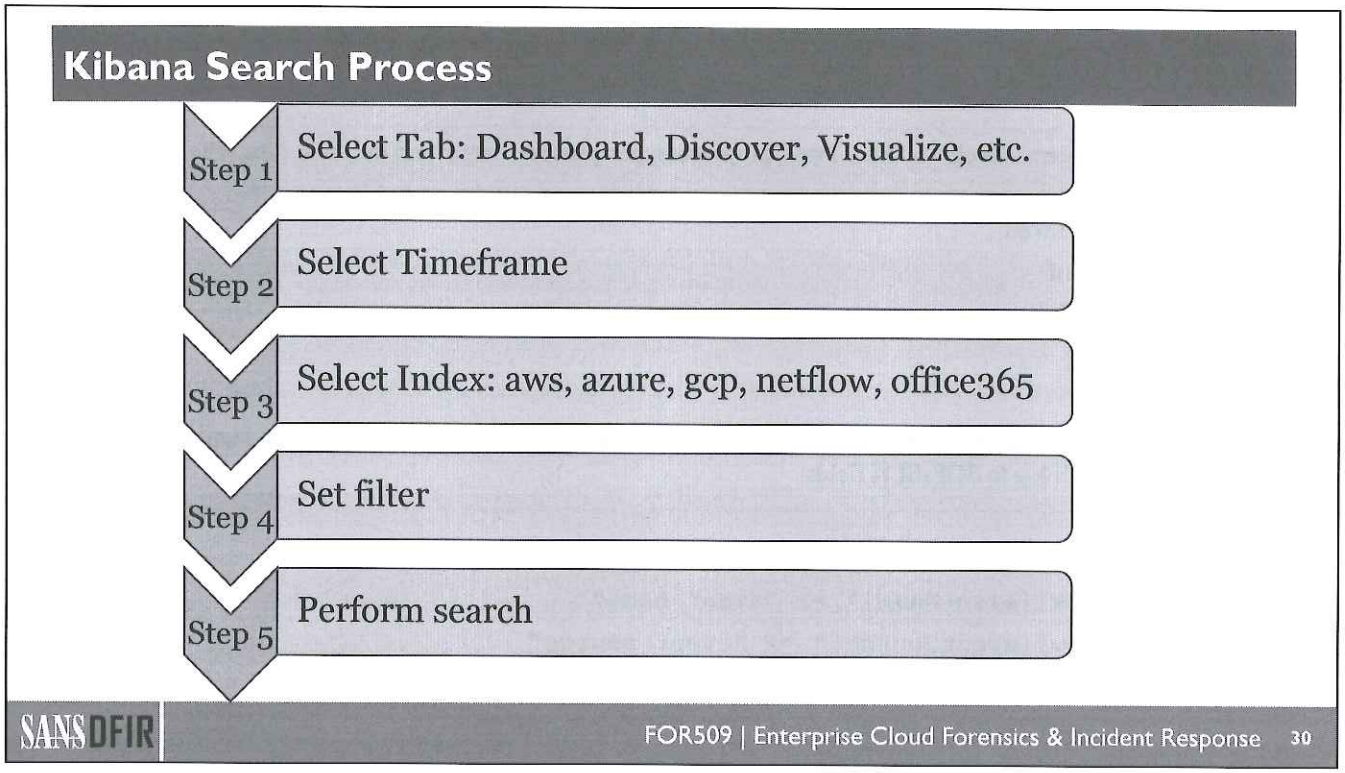
As an example, if you look at 6901-aws.conf, you will see that the parser's main function is to map the fields from the raw AWS log to SOF-ELK fields:

```
rename => {
        "[raw][eventName]" => "event_name"
        "[raw][eventSource]" => "event_source"
        "[raw][awsRegion]" => "aws_region"
        "[raw][sourceIPAddress]" => "source_host"
        "[raw][requestID]" => "request_guid"
        "[raw][eventID]" => "event_guid"
        "[raw][eventType]" => "event_type"
        "[raw][additionalEventData][bytesTransferredIn]" => "bytes_in"
        "[raw][additionalEventData][bytesTransferredOut]" => "bytes_out"
        "[raw][userIdentity][accessKeyId]" => "access_key_id"
        "[raw][requestParameters][bucketName]" => "bucket_name"
        "[raw][requestParameters][Host]" => "hostname"
        "[raw][resources][0][ARN]" => "aws_resource_name"
        "[raw][resources][0][type]" => "aws_resource_type"
        "[raw][userAgent]" => "useragent"
      }
```

For maximum efficiency, this mapping is limited to the most important fields and everything else is ignored:

```
# remove remaining fields
   mutate {
     remove_field => [ "raw" ]
   }
```

This is an important nuance as any other commercial SIEM will do some type of mapping but may not expose their decision process. You will want to check what fields your SIEM may have decided to drop for their schema.



We are now faced with a large amount of data in our SOF-ELK instance. Where do we start?

Step 1: Decide which SOF-ELK analytics tab is best for the query you would like to make

- Discover: filter and search raw events
- Visualize: create various types of charts (bar, percentage bar, area, pie, etc.) or tables
- Dashboard: collection of saved visualizations and/or searches in one location

Step 2: Select your timeframe

- This is a key step as your results will only be valid for the timeframe selected. If you don't get the expected results, double check your timeframe

Step 3: Select your index

- Each log must be imported into a specific index. Unlike certain commercial solutions, you can't search across all indices
- Field names may vary between indices
- If you don't get the results you expected, make sure you have the correct index and the correct field name

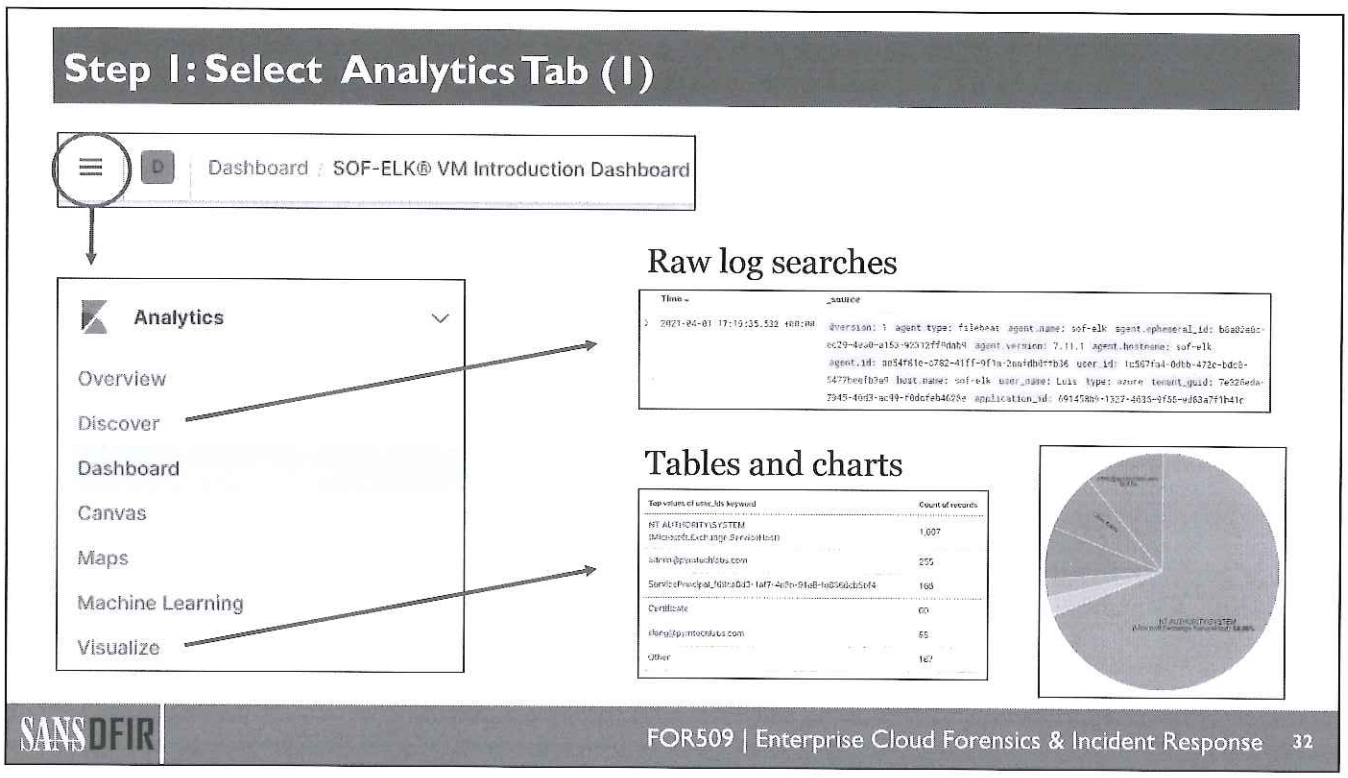
Step 4: Set a filter (optional)

- Data reduction can be done either with a filter or a search. If you know a specific piece of information, it's advisable to filter your data for that information and then perform your search in the next step. This will keep your search bar less cluttered. For example, you could set your filter for a specific userid and then use the search bar to find out what actions this person performed

Step 5: Perform your search

- You are now ready to search on any field that may be relevant to your investigation
- You can save your searches and use them later or add them to panels in a dashboard

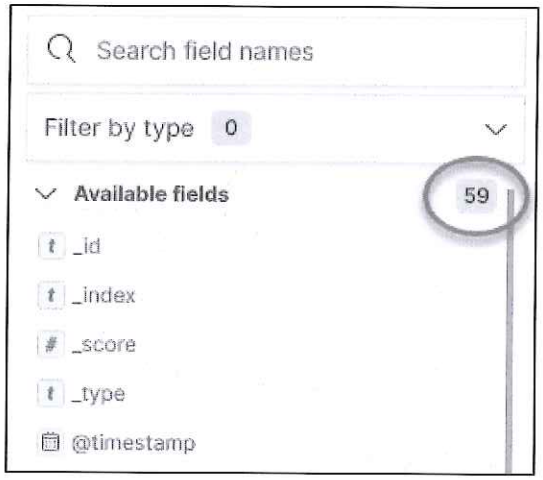
SOF-ELK has many other features, but the ones listed above are the ones we will use in FOR509.



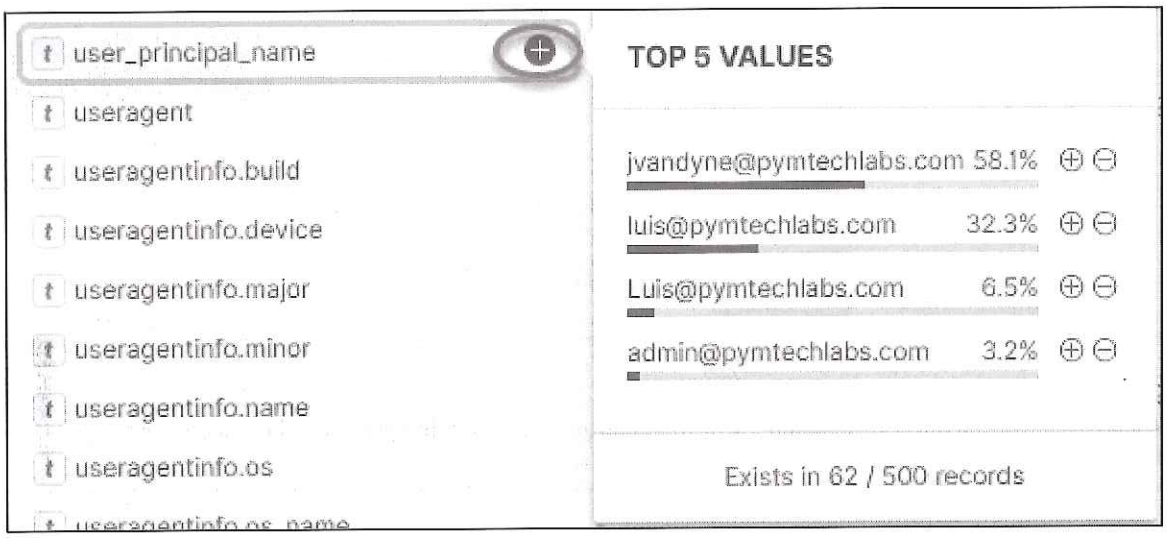
Under analytics, you have a number of options. The ones we will use in FOR509 are:

- Discover
- Dashboard
- Visualize

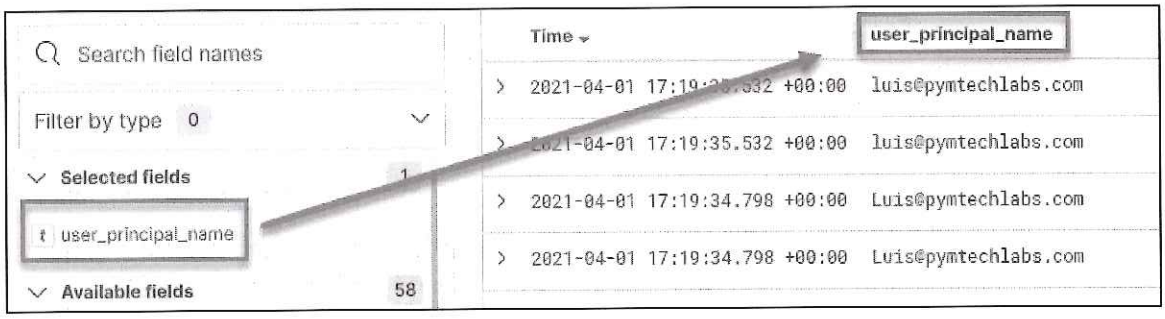
You should start with the Discover tab as it gives you direct access to raw events. On the left side of the Discover tab, you will get a list of available fields. This is extremely useful if you are not familiar with the log source. In the example below, you can see that this log has 59 different fields. You can also start typing the name of a field in the "Search field names" to narrow down possible fields.



You can select any field and get a short summary of the top 5 values. Most importantly, you can click on the "+" sign and add this field to your list of selected fields.

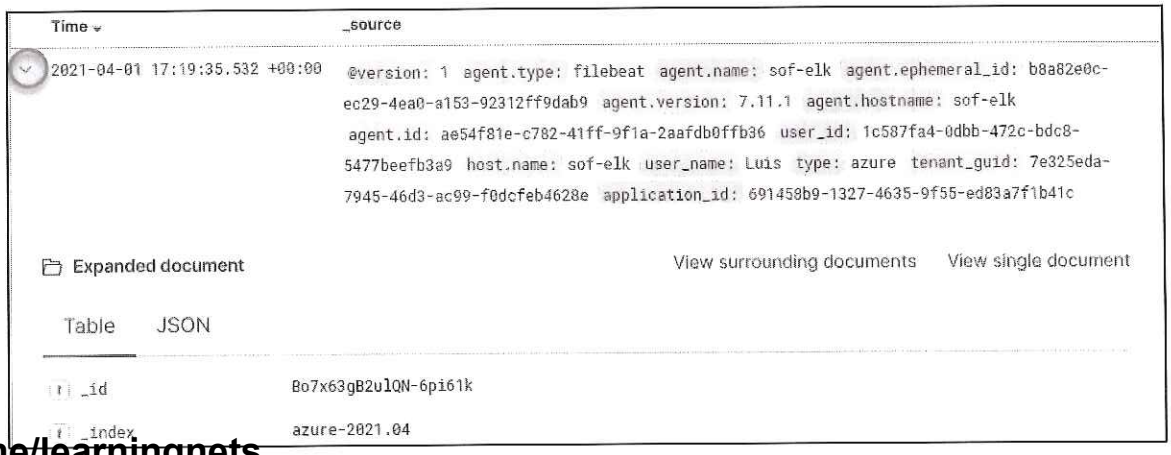


Once selected, your main window will no longer show the entire raw event. Rather, it will start building a table with the fields you have selected.

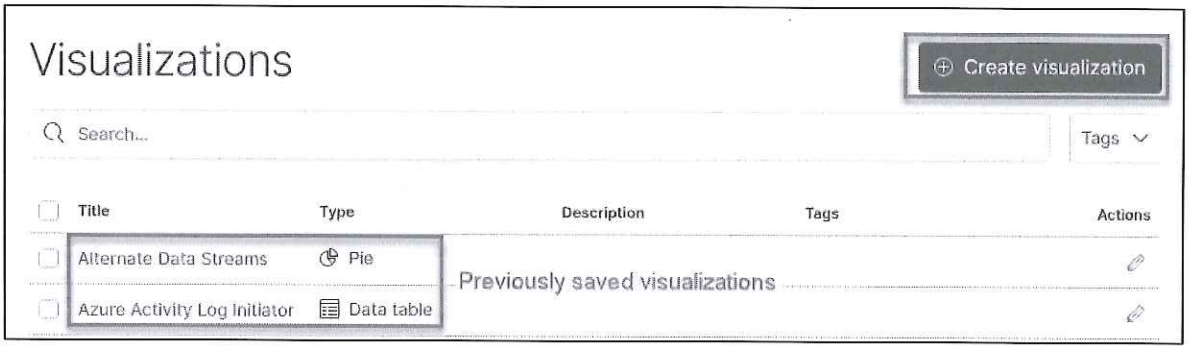


You can select as many fields as you wish, but anything above 4 or 5 will quickly exceed the real estate on your screen.

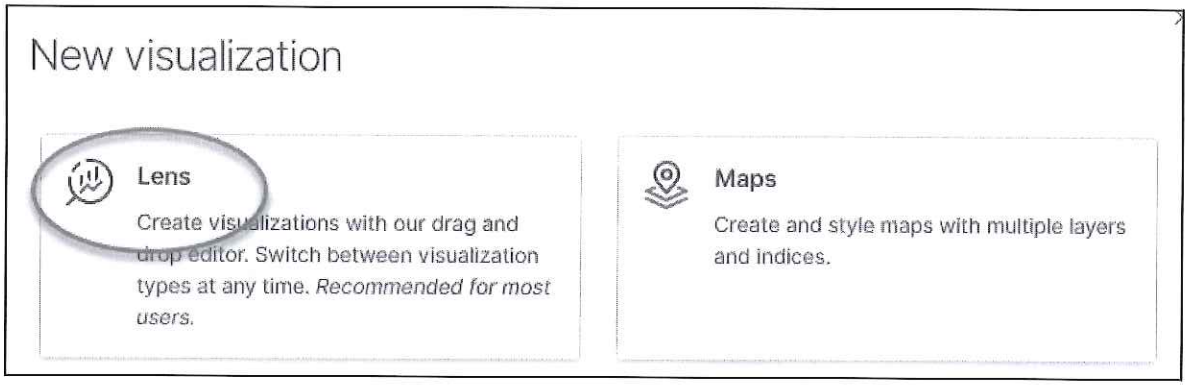
At any point in time, you can click on the chevron next to a raw event to expand it and see every field it contains:



The visualize tab allows you to create charts and graphs. When you select that tab, you will get a screen showing all your saved visualizations. Select "Create visualization" to start a new one.



When you select "Create visualization", you will be presented with a few options. The easiest one to use is "Lens":

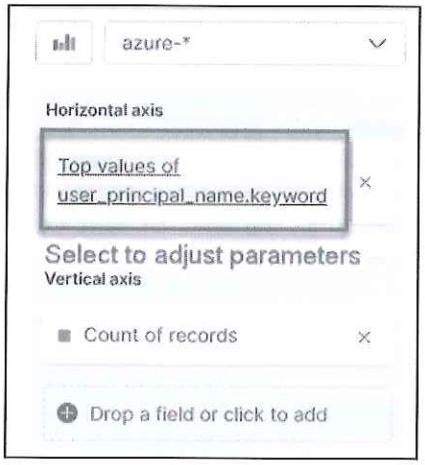


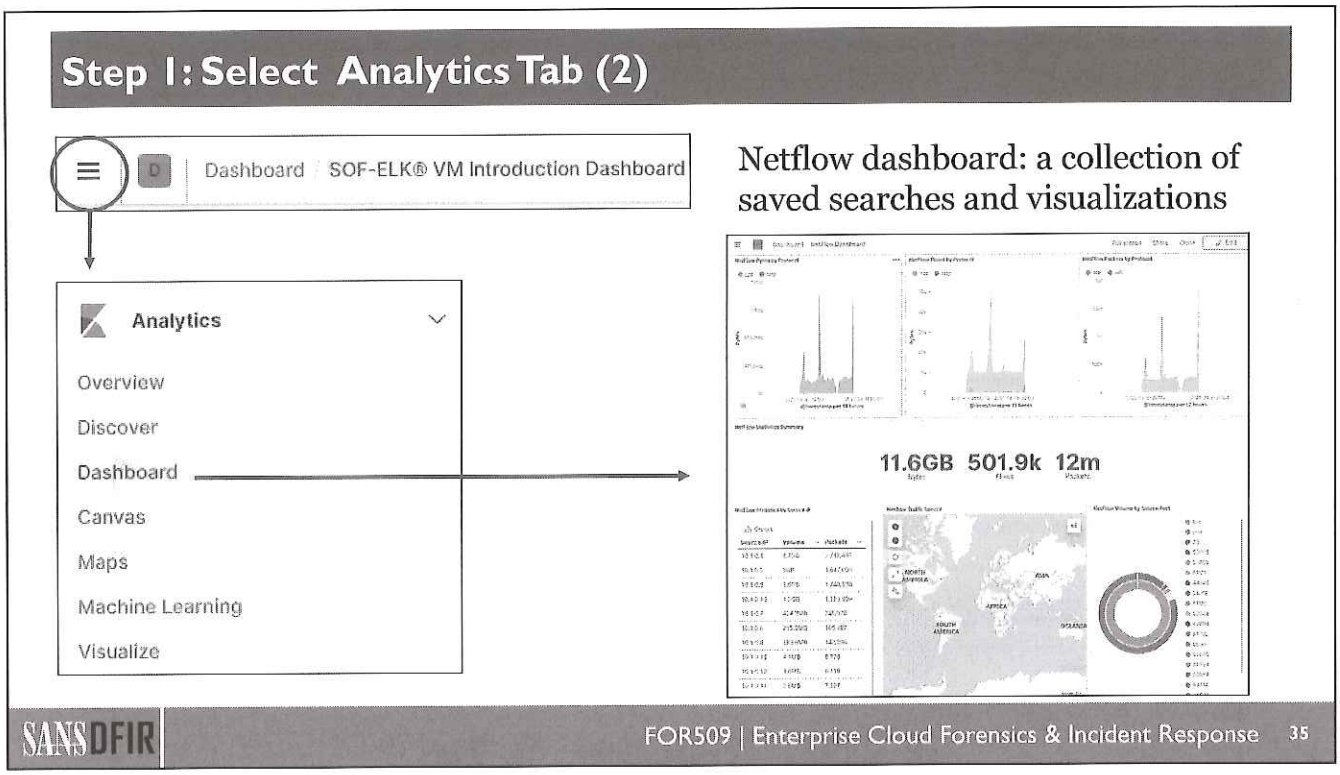
You can now select from a number of chart types.

The best way to learn about all these chart types is to try them out.

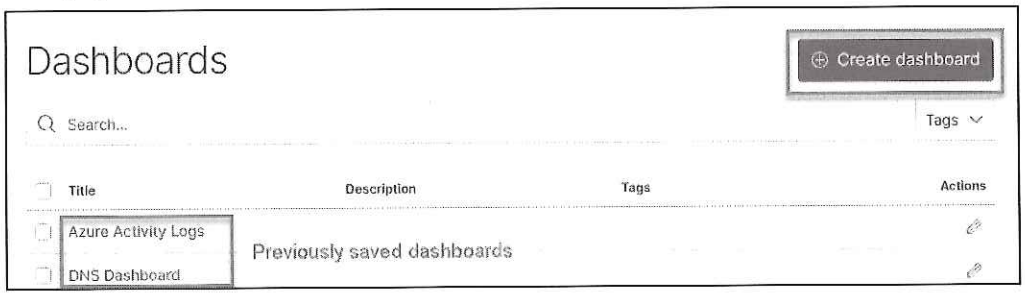
Once you have selected a chart type, you can drag and drop any of the fields.

The menu on the right will let you adjust the parameters for the field you selected.

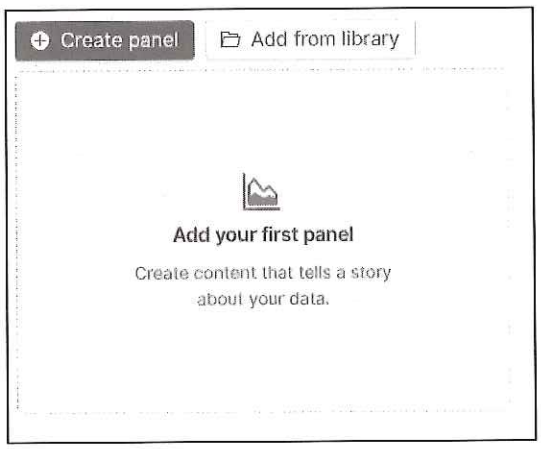




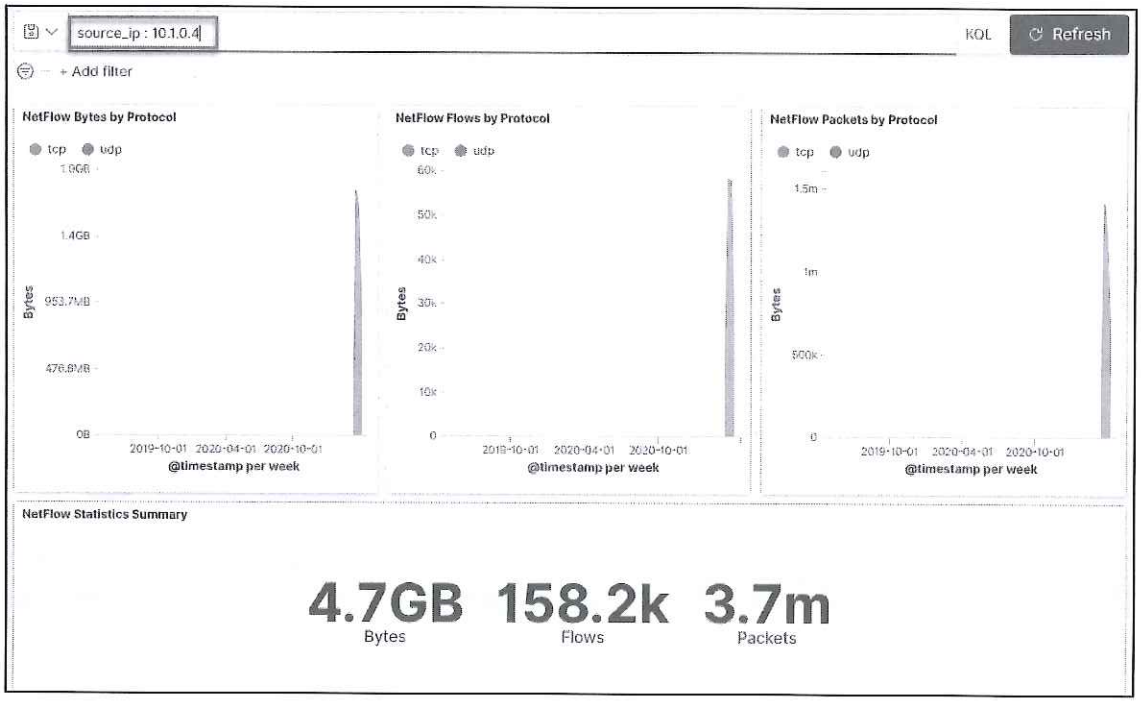
Now that we have created and saved various visualizations, we can combine them in a dashboard. Each visualization will be shown in a panel, and you can organize the panels any way you wish on the page. As an example, this slide shows the Netflow dashboard.



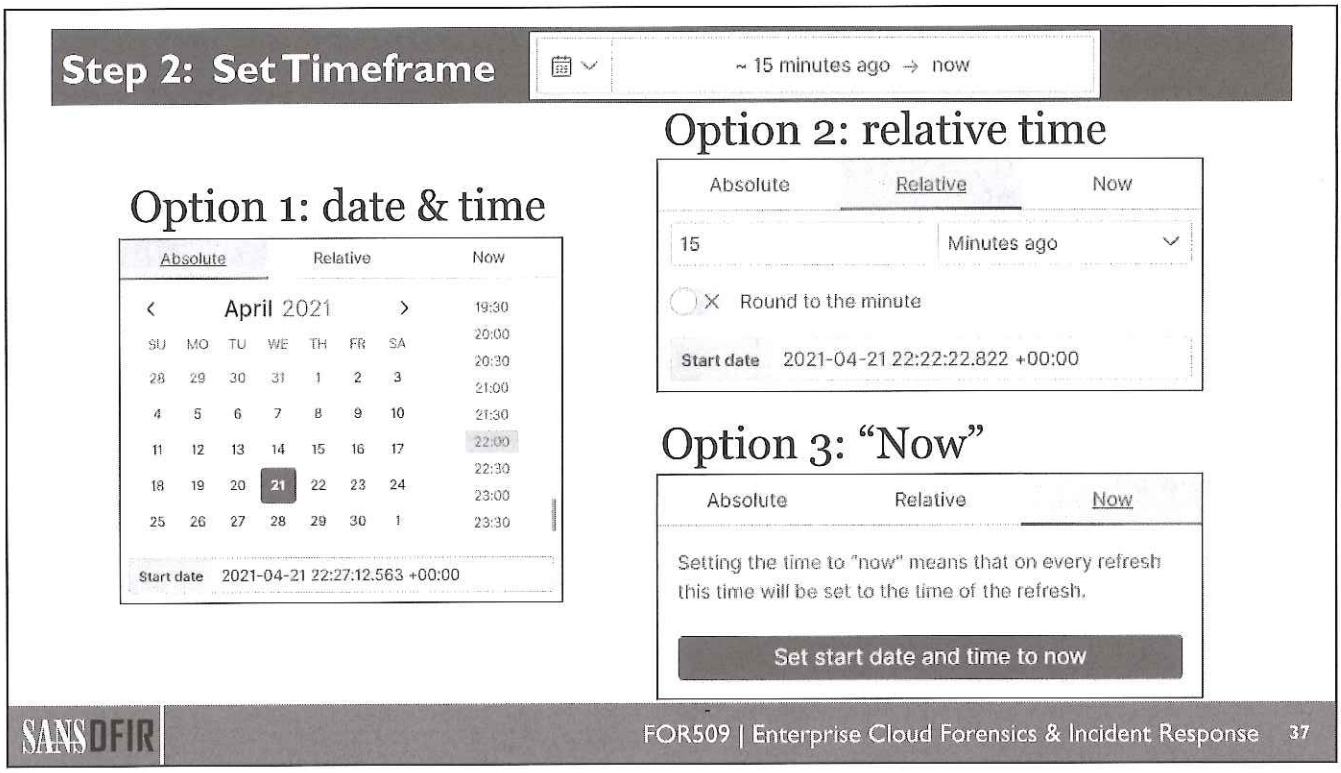
When you create a dashboard, you are presented with a blank page and the options to either "Create panel" or "Add from library". The library refers to previously saved visualizations or searches.



The dashboard tab offers both the search and timeframe options. One very powerful aspect of these dashboards is that all panels will be dynamically updated based on any search or time parameters that you enter.



There are many more features and the best way to discover them is simply to create your own dashboard.

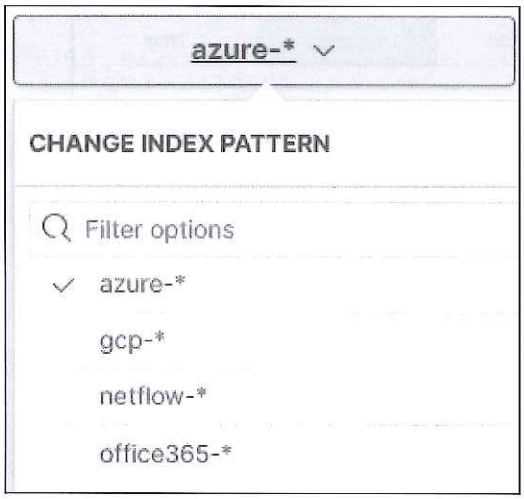


The timeframe bar allows you to select a bracket of time to narrow down your data. There are 3 options to specify date and time:

- Option 1: specify an absolute date and time
- Option 2: use relative time
- Option 3: use "Now"

While these options are pretty self-explanatory, we would like to emphasize to always pay attention to the timeframe bar as it's easy to inadvertently change it and get incorrect results.

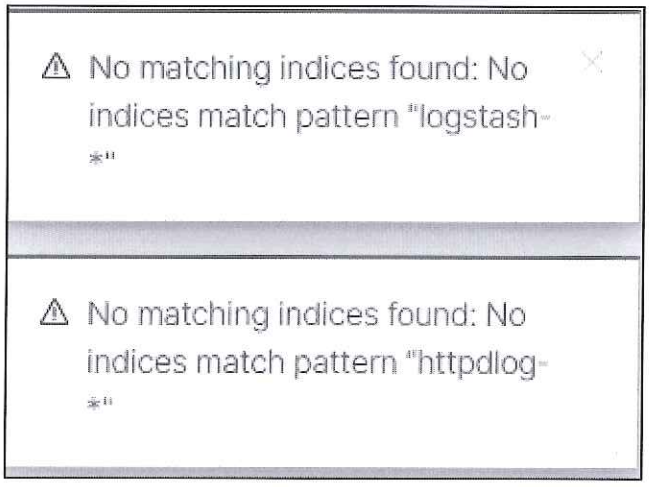
Step 3: Select Index



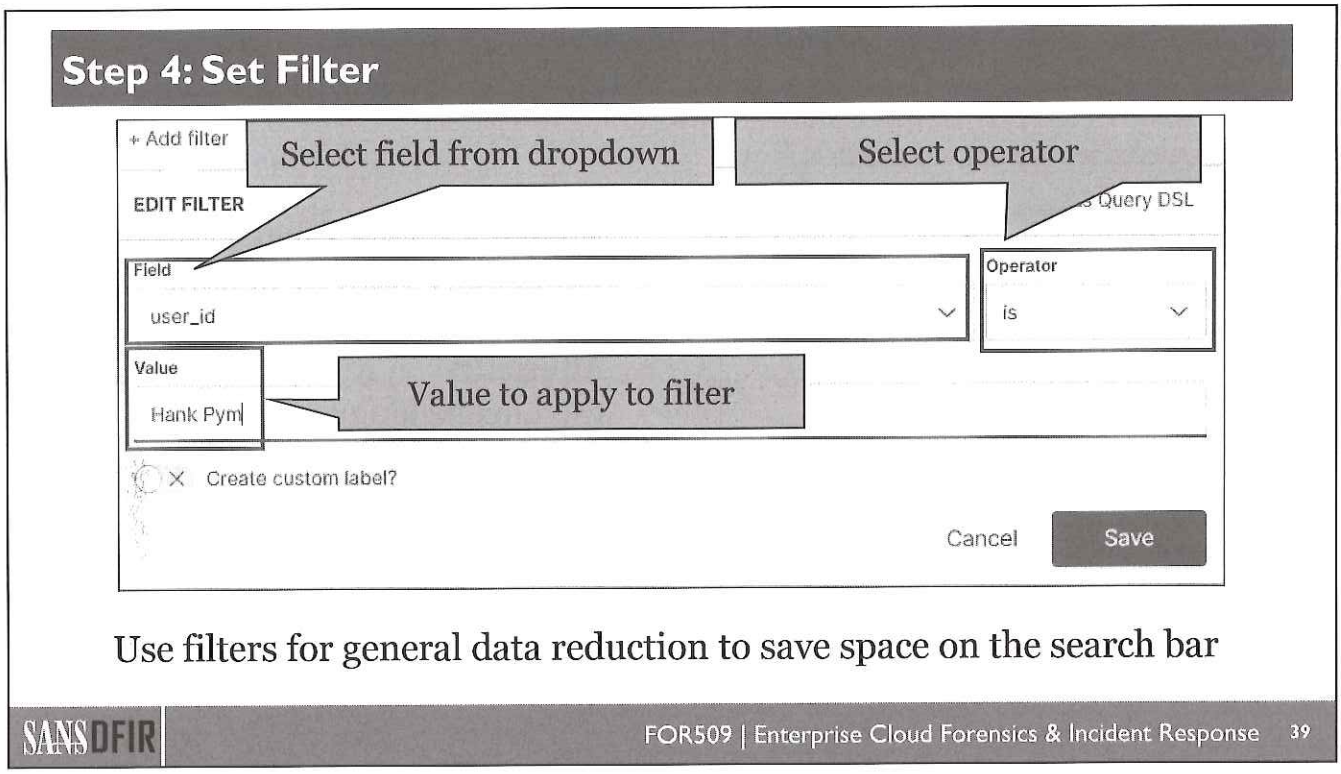
Index	Cloud
aws-*	Amazon AWS
azure-*	Microsoft Azure
gcp-*	Google Cloud Platform
netflow-*	Flow Logs (any clouds)
office365-*	Microsoft 365

When importing logs into SOF-ELK, we must specify an index. This slide shows the indices we will be using for FOR509.

As previously mentioned, the public release of SOF-ELK supports many more indices. Since we don't have data in these other indices, you will see these error messages (simple close the popup window):

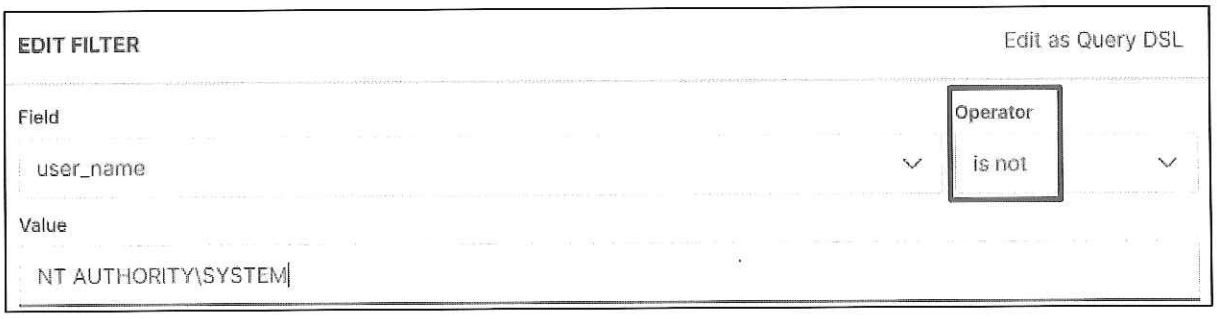


These messages simply indicate that no data has been loaded under that specific index.

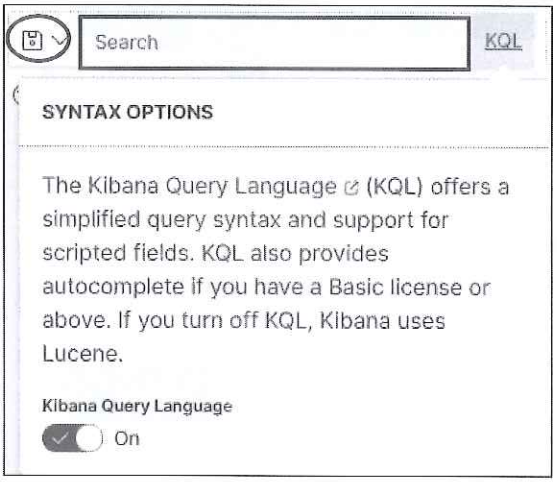


The filter section is great for initial data reduction. It's very user friendly as it presents all available fields in a drop-down menu. Possible operators are presented in the next menu. Finally, you only need to specify the value you wish to look for.

One great way to use filters is to exclude information by using the operator "is not":



Step 5: Perform Search



- Kibana uses the Kibana Query Language (KQL) by default
- KQL uses simple query syntax
- Alternatively, Kibana can use the Lucene language
- Searches can be saved (floppy disk icon on the left of the search bar)

The final step is to search for relevant information. Kibana supports two query languages: KQL (Kibana Query Language) and Lucene. KQL is the default and very easy to use. There are many good tutorials on the internet in addition to the official documentation.[1]

References:

[1] https://for509.com/kql

Search Examples

1	Search for a term across all fields	`pymtechlabs`
2	Search for a term within a specific field	`organization_name : pymtechlabs` `organization_name.keyword : pymtechlabs.com`
3	Boolean query	`user_ids : (luis or slang)`
4	Boolean query with multiple fields	`user_ids : luis AND operation : UserLoginFailed`
5	Exist query (very useful to eliminate records that don't have data in a specific field)	`useragent : *`
6	Negate a value	`not workload : Exchange`

This slide shows a few examples that you will find useful for the FOR509 labs. As you can see the syntax is quite simple.

If you want to search across all fields, you can simply enter your search term as shown in the first example.

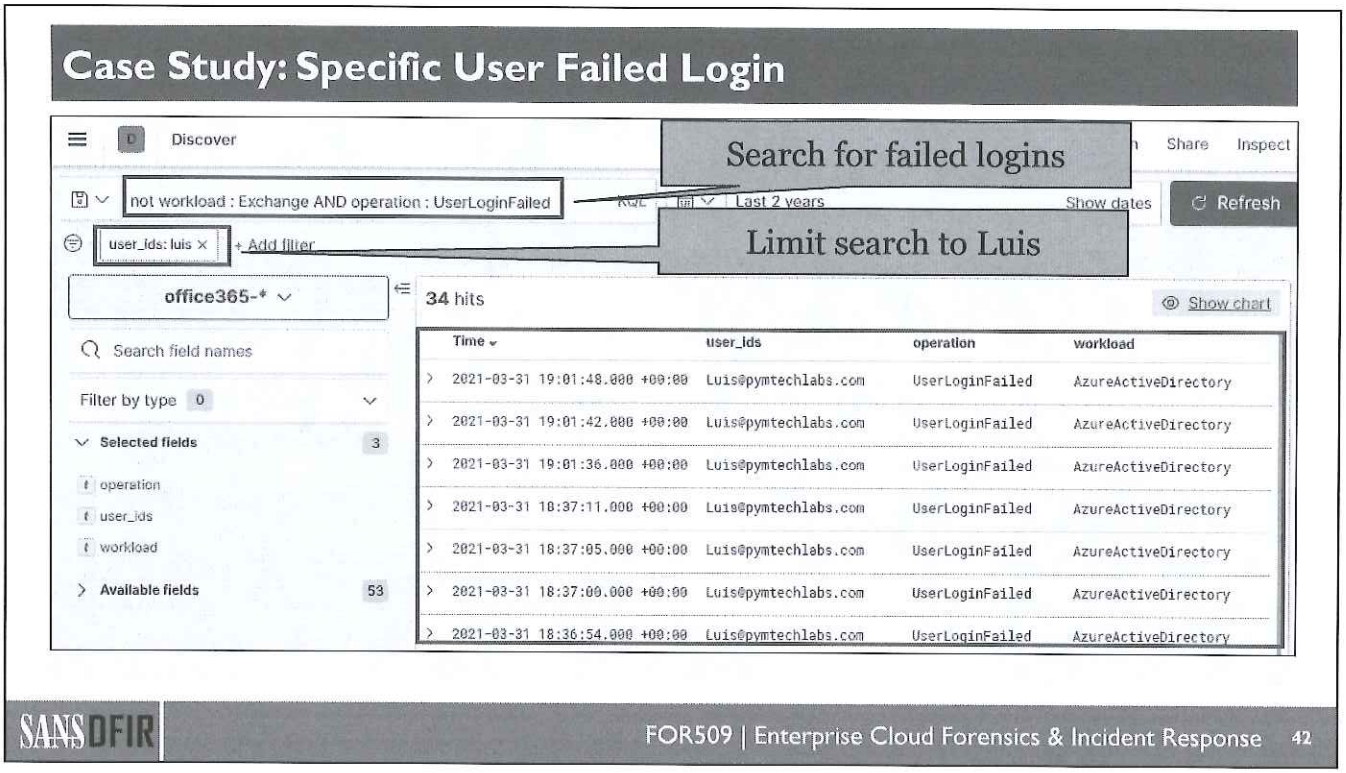
The second example requires a bit more explanation. Elasticsearch will break up text strings on certain delimiters: ".","/","-", whitespace, and many others. Elasticsearch refers to these "tokenized" fields as "analyzed". The benefit is that if you search "`organization_name : pymtechlabs`", you will match "`pymtechlabs`", "`pymtechlabs.com`", "`pymtechlabs.onmicrosoft.com`". However, Elasticsearch also leaves a version of that field intact and adds the term "`.keyword`" to the field to indicate the non-tokenized version. This version is useful if you want to do an exact search.

The third and fourth examples demonstrate various Boolean queries.

The fifth example is very useful and frequently used. Not all log entries contain data in every single field. You are frequently faced with empty fields, and this is the query you will use to limit your search to fields that contain data.

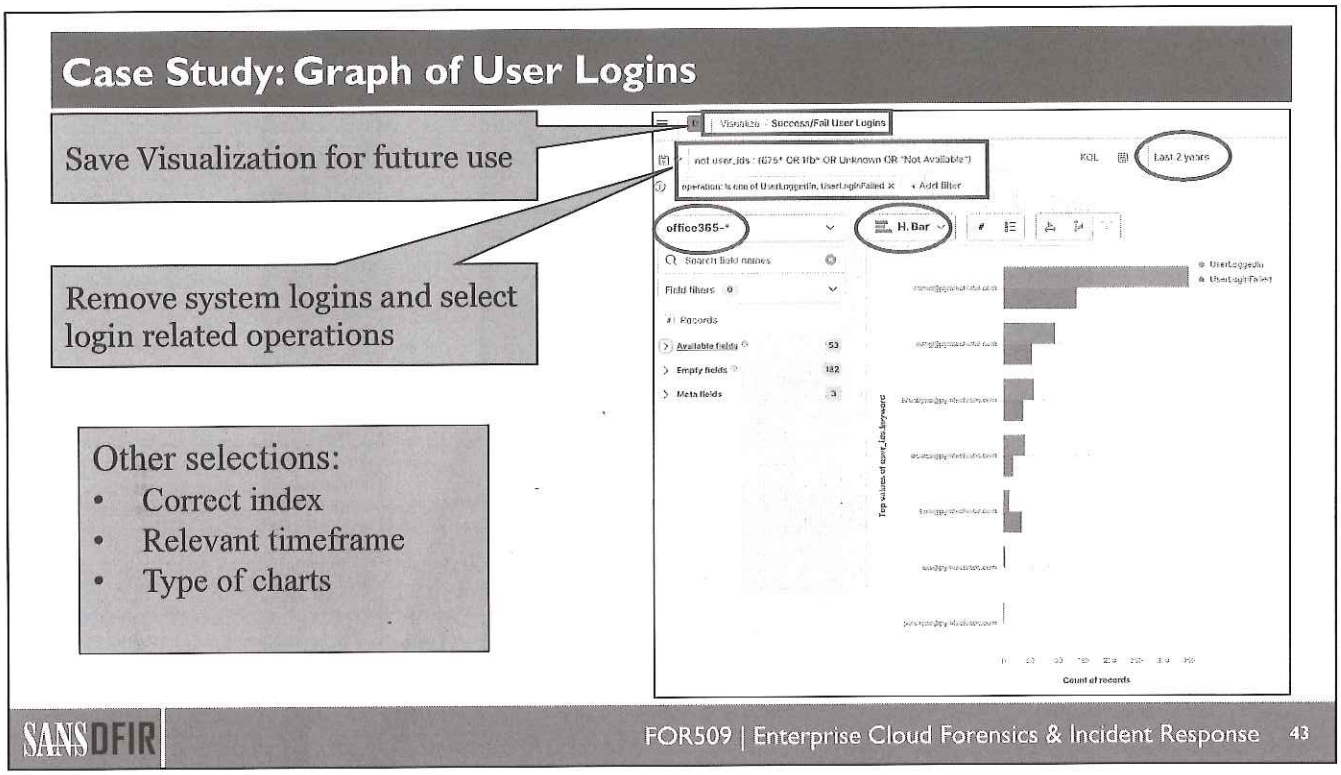
The sixth example comes into play when you have too much data. You may want to eliminate a certain type of data and the NOT operator will do that for you.

Note that you don't have to leave a space before and after the colon (":").



As an example we are searching for instances where Luis@pymtechlabs.com failed to successfully login. We will admit that the "`not workload : Exchange`" is superfluous since Exchange doesn't record successful and failed logins. However, as you build more and more complex queries, you have to go through a trial and error phase to find out which fields are meaningful to your query and which ones are not.

Be sure to verify your results and make sure you didn't accidently eliminate data that was relevant. It's a good idea to try multiple scenarios and test your assumptions.



We previously searched for failed logins for user Luis@pymtechlabs.com. Luis had 34 failed logins. Is this normal in our environment? Is this too many and indicated a possible attack? We can put this information into context by creating a chart of other user's successful and failed logins.

In this example, we use the filter:

```
operation : is one of UserLoggedIn, UserLoginFailed
```

The operator "`is one of`" is very useful to provide a list of items. Since "operation" has numerous possible entries, we need to select the ones related to user logins.

Our initial results show a number of system accounts as well as entries called "Unknown" and "Not Available". For the purposes of this graph, we aren't interested in these entries hence the filter:

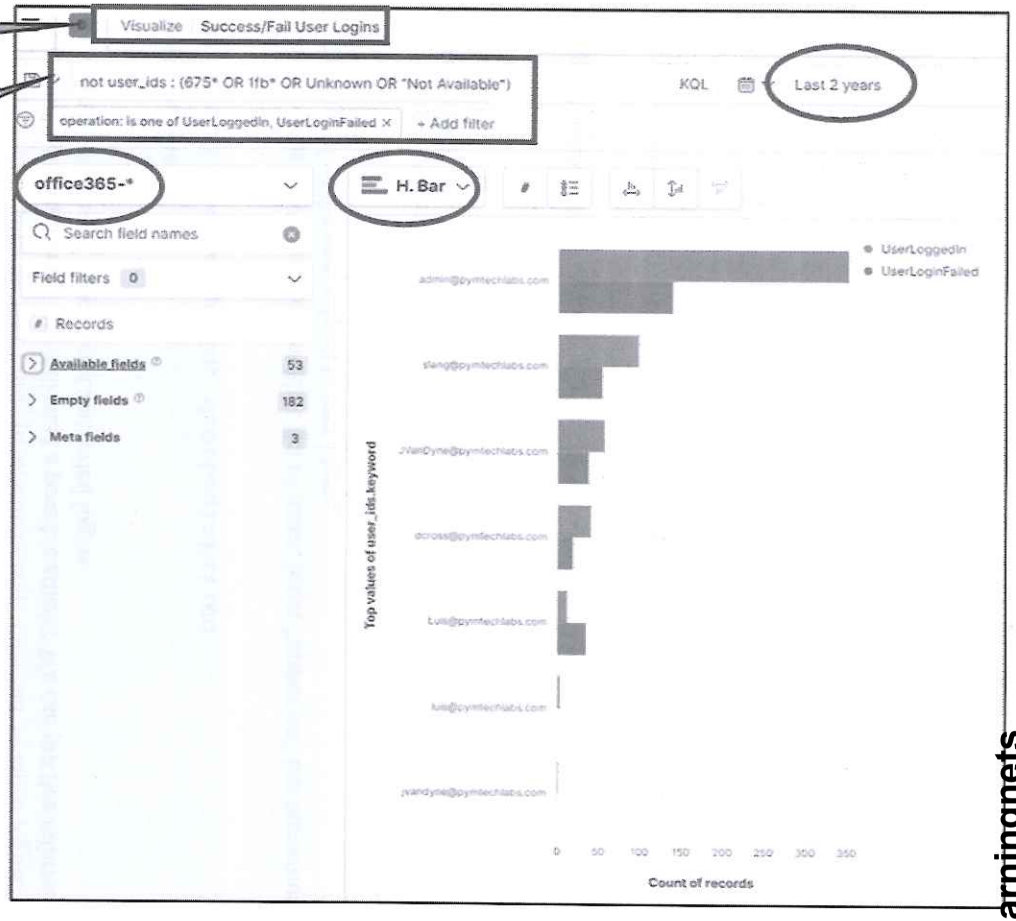
```
not user_ids : (675* OR 1fb* OR Unknown OR "Not Available")
```

After double checking the index and the timeframe, you can select any chart type that you would like.

You will notice that we saved this visualization with the name "Success/Fail User Logins" so we can add it to our dashboard.

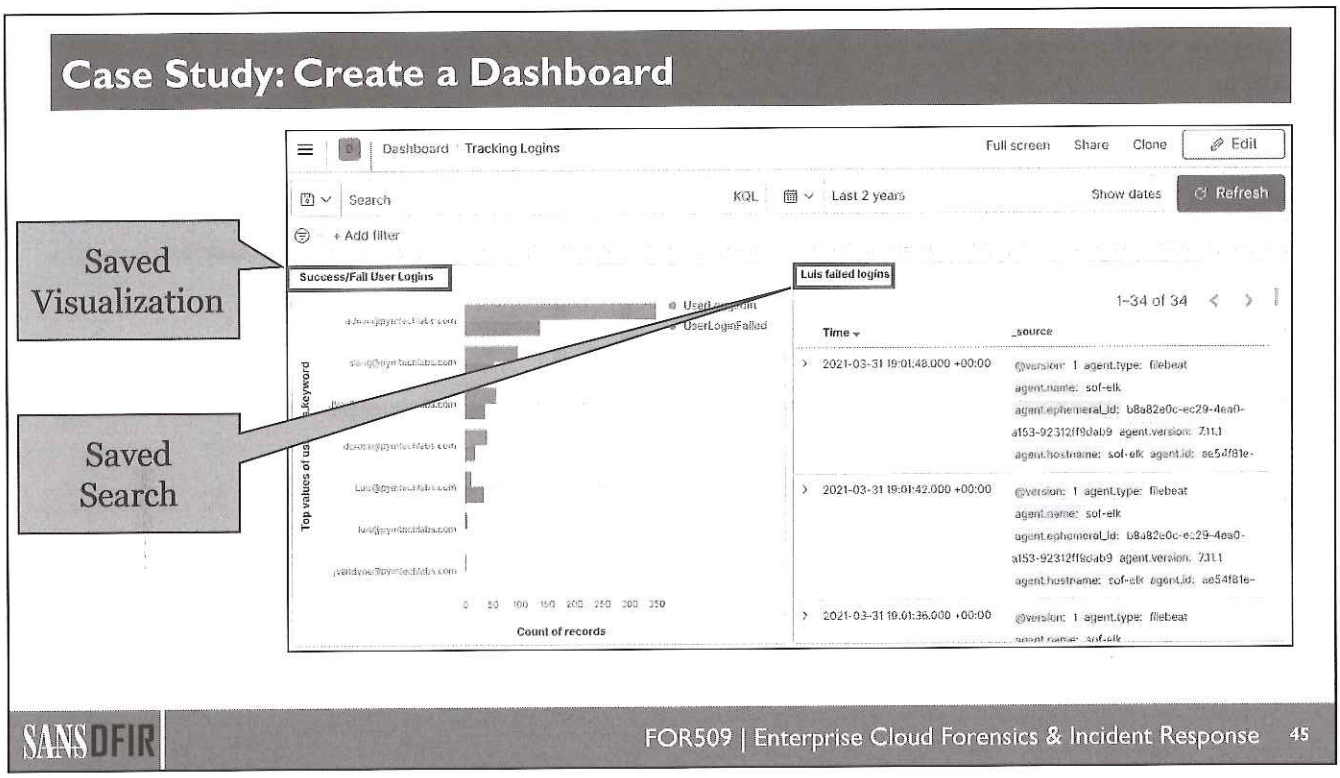
Save Visualization for future use

Remove system logins and select login related operations



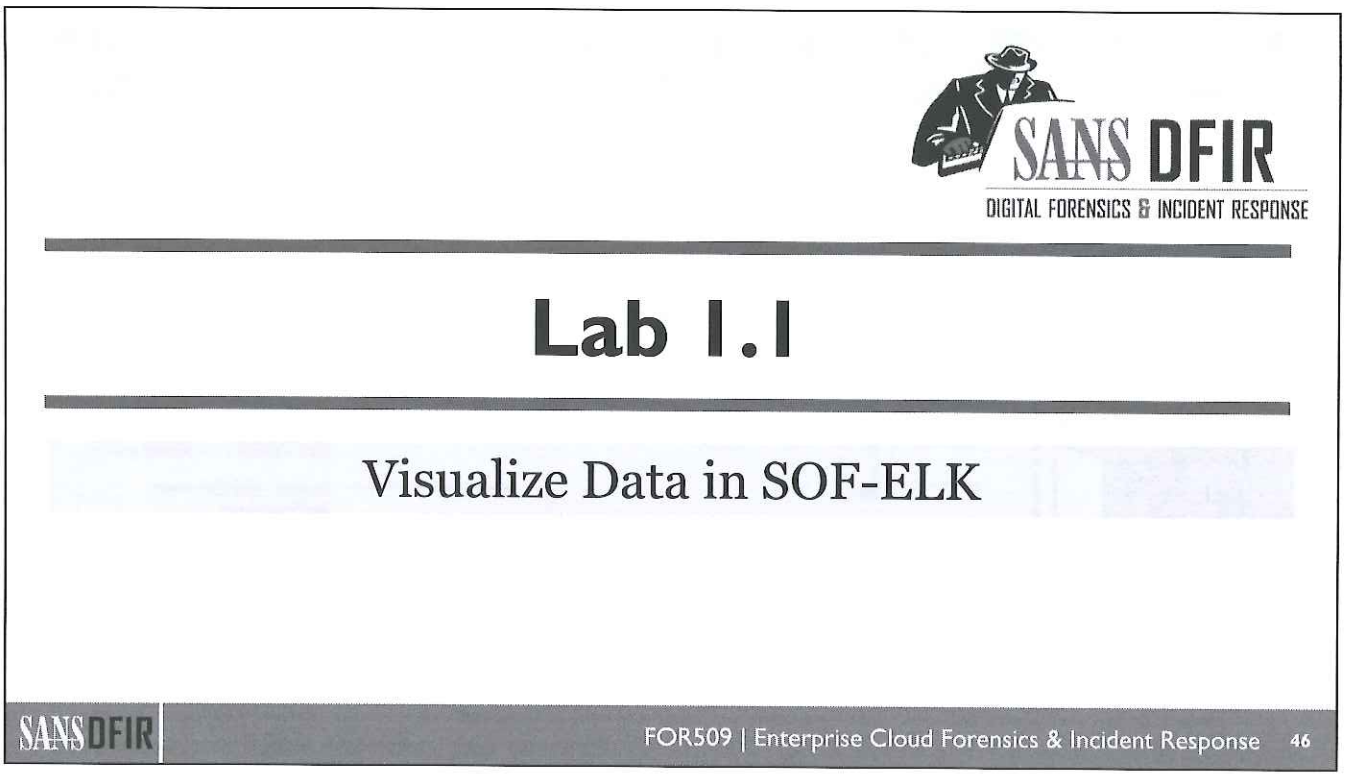
Other selections:

- Correct index
- Relevant timeframe
- Type of charts

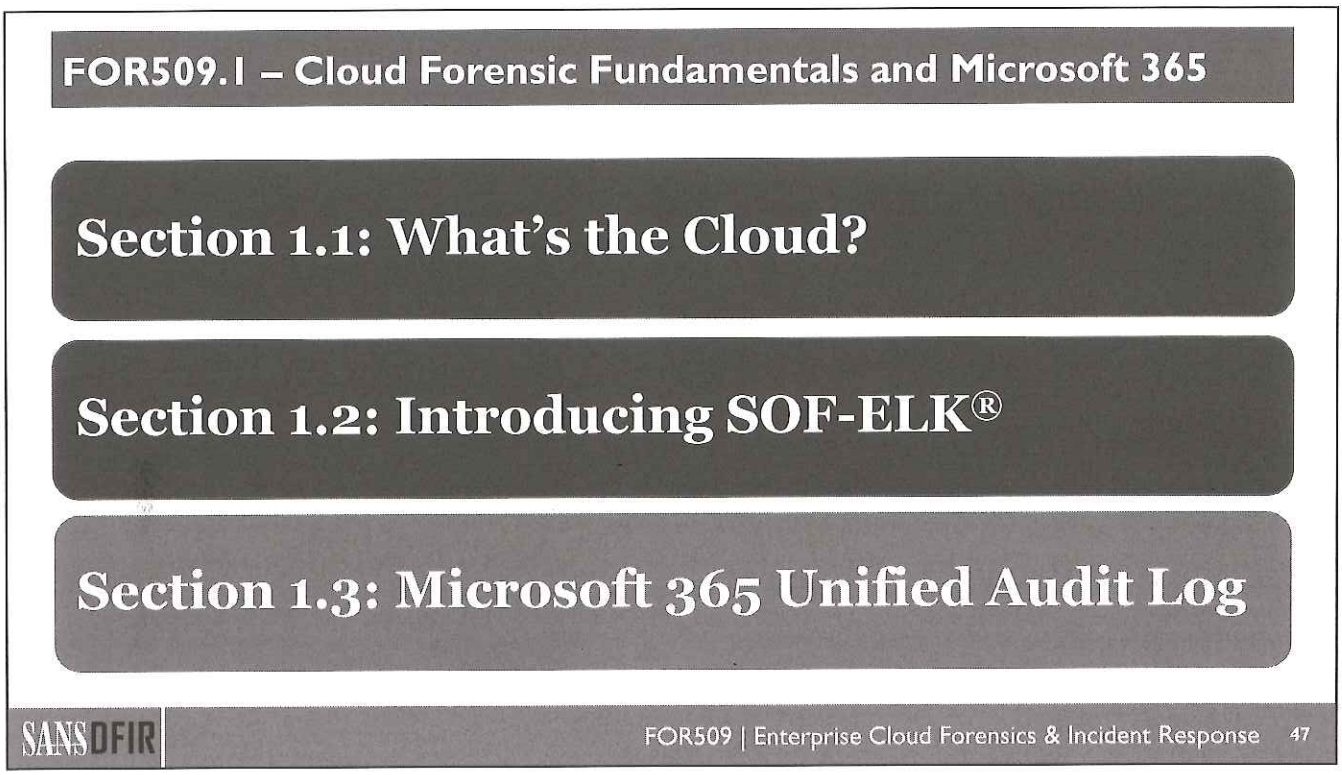


We can now combine our saved visualization and our saved search into a single dashboard. You can create very elaborate dashboards with many panels. A great feature is that any additional search, filter or time change you enter on the dashboard will automatically update every panel.

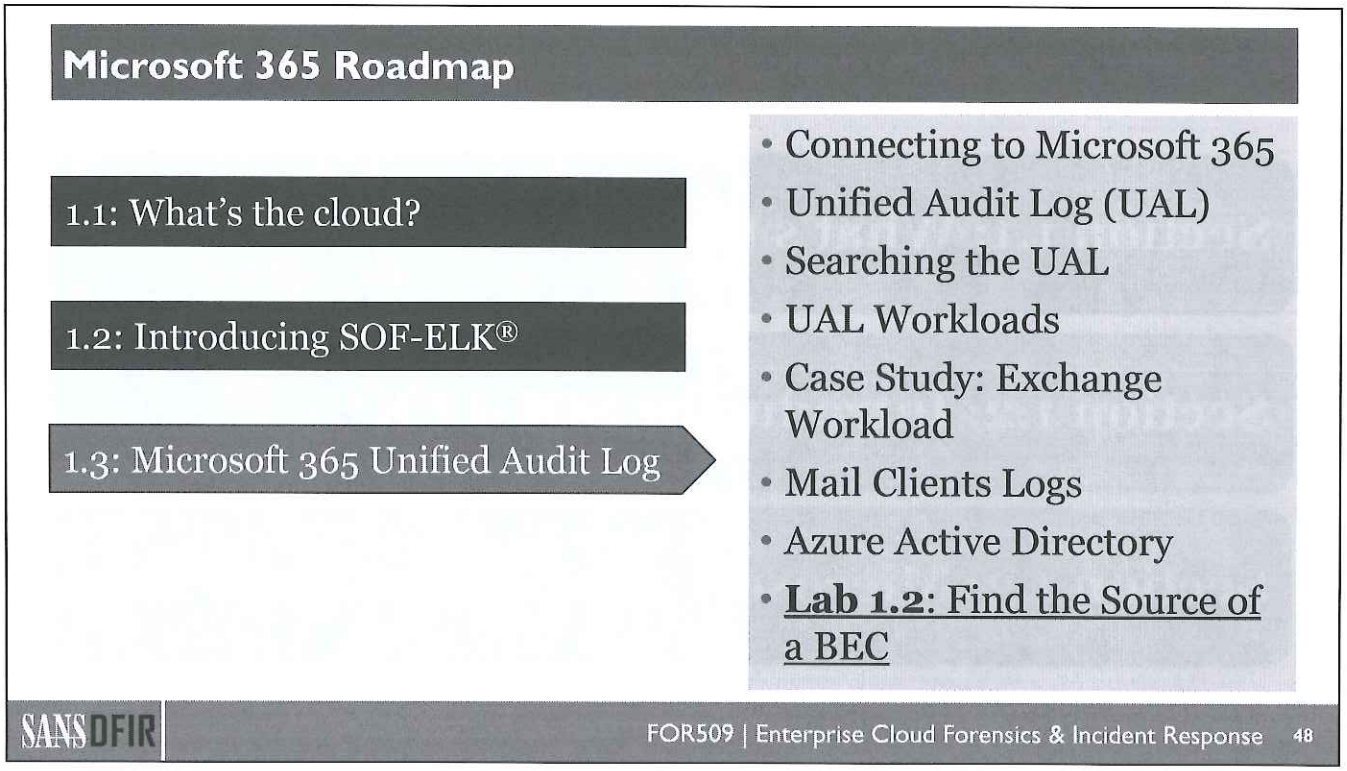
The only way to really appreciate the power of Kibana is to try it for yourself, so let's go to lab 1.1.



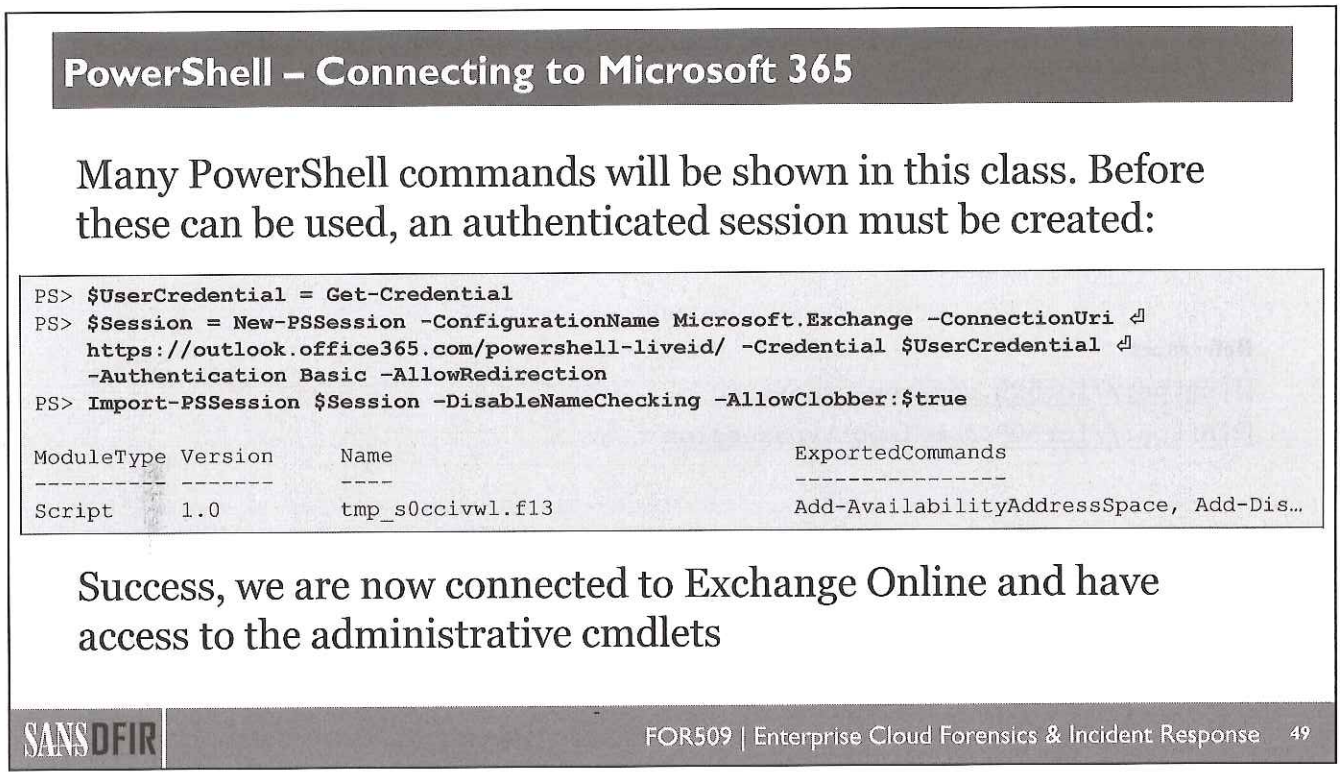
This page intentionally left blank.



This page intentionally left blank.



This page intentionally left blank.



In this section, we will introduce a number of PowerShell commands. Before we can issue any Microsoft 365 PowerShell command, we must establish a session and import the appropriate PowerShell cmdlets in your terminal session.

There are three steps that we must complete:

Step 1: Provide your credentials

```
$UserCredential = Get-Credential
```

Step 2: Create a persistent connection to Office 365 Exchange Online[1]

```
$Session = New-PSSession -ConfigurationName Microsoft.Exchange -
ConnectionUri https://outlook.office365.com/powershell-liveid/ -
Credential $UserCredential -Authentication Basic -AllowRedirection
```

Step 3: Import cmdlets from PSSession into your current session[2]

```
Import-PSSession $Session -DisableNameChecking -AllowClobber:$true
```

These steps are specific to your current terminal session (i.e. your current PowerShell window). If you open a new window, you will have to repeat these steps in that window.

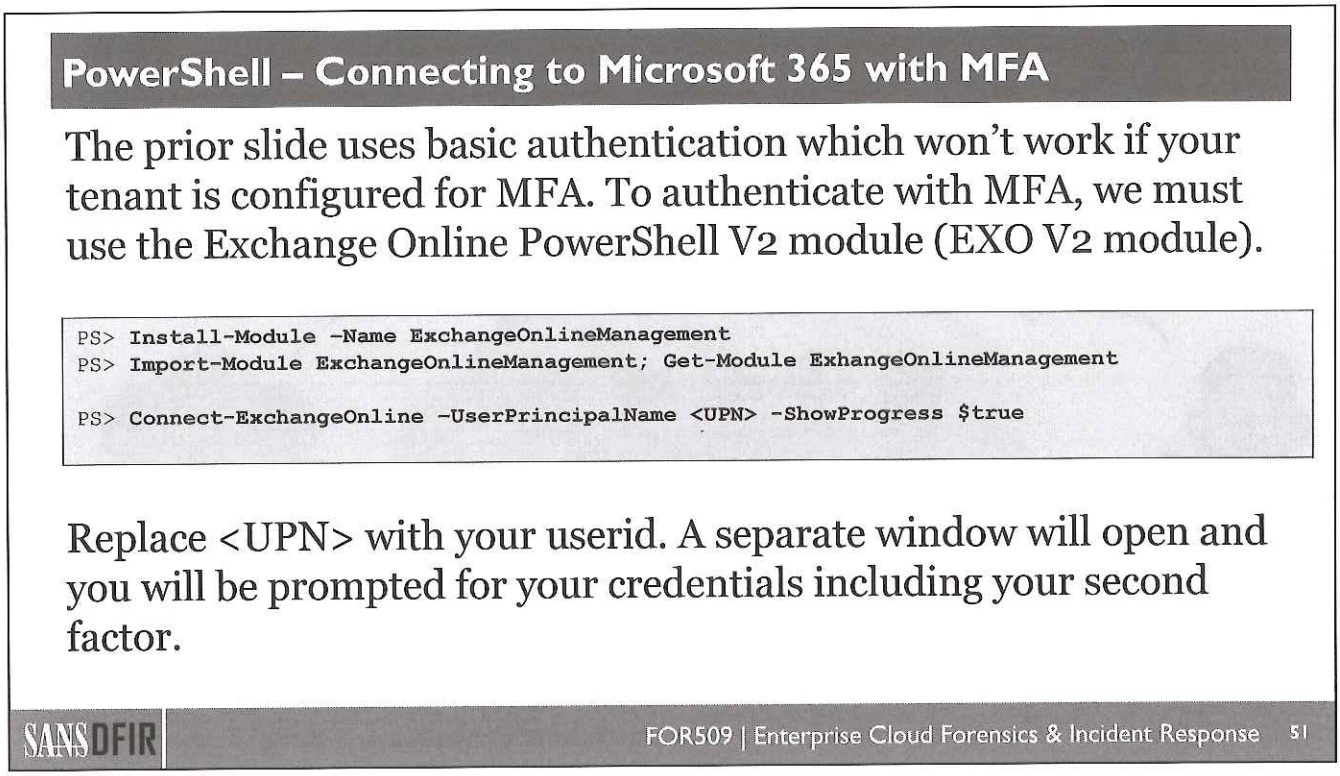
A session will be created based on the privileges of the account provided in step 1. Step 3 will import cmdlets based on the privileges of that account. If the account doesn't have the necessary administrative permissions, you will be missing key cmdlets such as Search-UnifiedAuditLog, Get-Mailbox, etc.

All future slides with PowerShell instructions will assume that you have successfully established a connection to Microsoft 365.

References:

[1] https://for509.com/new-pssession

[2] https://for509.com/import-pssession



The prior slide uses basic authentication which won't work if your tenant is configured for Multi Factor Authentication (MFA). To authenticate with MFA, we must use the Exchange Online PowerShell V2 module (EXO V2 module).[1]

This module also introduces new and optimized cmdlets. However, this class will refer to the older versions of the cmdlets for maximum backwards compatibility.

Step 1: Install the EXO V2 module (if not already installed)
`Install-Module -Name ExchangeOnlineManagement`

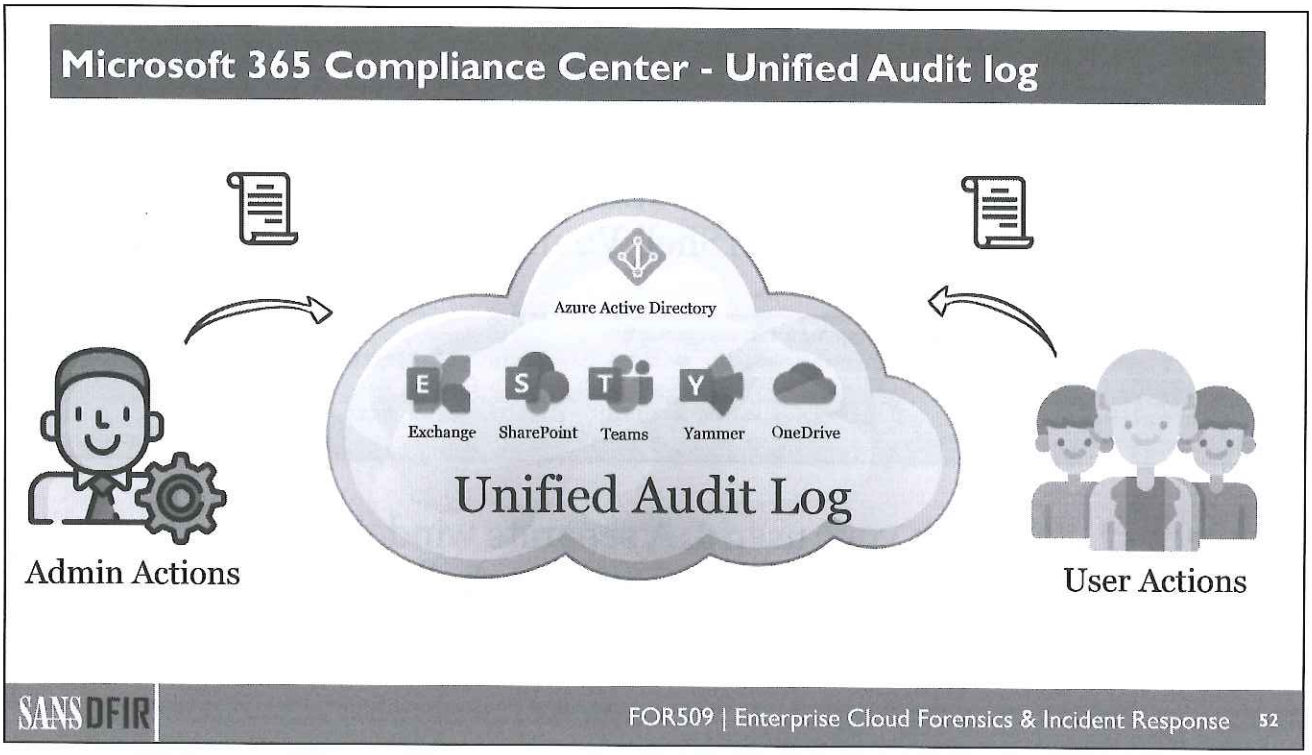
Step 2: Verify the module was installed and check the version
`Import-Module ExchangeOnlineManagement; Get-Module ExchangeOnlineManagement`

Step 3: Connect to Exchange Online using the EXO V2 module
`Connect-ExchangeOnline -UserPrincipalName <UPN> -ShowProgress $true`

Replace <UPN> with your userid (example: admin@pymtechlabs.com)

In both this slide and the last one, we connected to Microsoft Exchange Online so that we may access the Unified Audit Log (UAL). However, this requires administrative permissions to Microsoft Exchange. Some organizations may be reluctant to provide this permission simply to search the UAL. There is a second option. The UAL may also be searched by someone who has eDiscovery permissions. We will discuss eDiscovery in a later section. Note that this option only works with basic authentication.

References:
[1] https://for509.com/exchangeonline



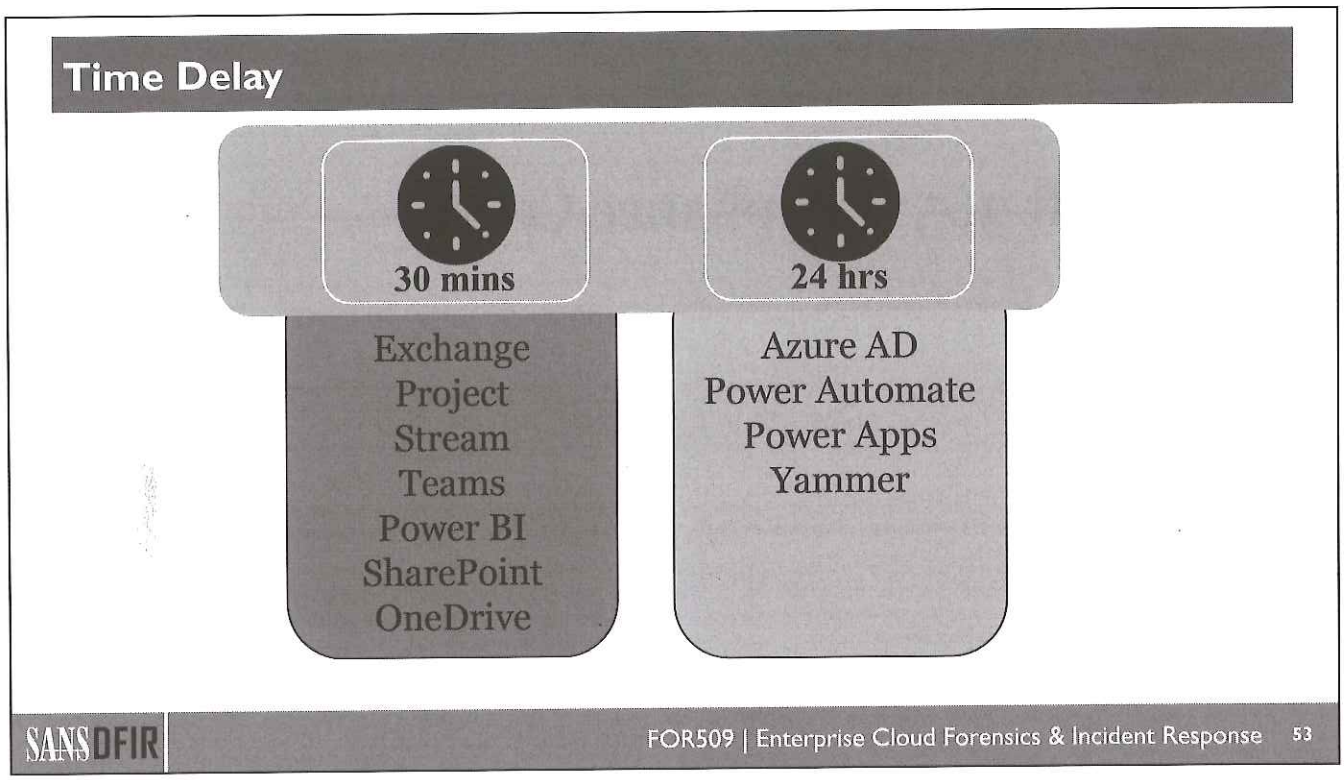
Microsoft 365 offers numerous applications. Fortunately, logs are aggregated in a single location called the Unified Audit Log (UAL).

The UAL will record both user activity and admin activity. The list of Microsoft 365 applications is constantly changing. These are some of the most common:

- Azure Active Directory
- SharePoint Online
- OneDrive for Business
- Exchange Online
- Power BI
- Teams
- Dynamics 365
- Yammer
- Flow
- Stream
- Workplace Analytics
- PowerApps
- Forms

As we will discuss shortly, the UAL can be queried in three different ways:

1. Microsoft 365 Compliance Center
2. PowerShell
3. Microsoft 365 API

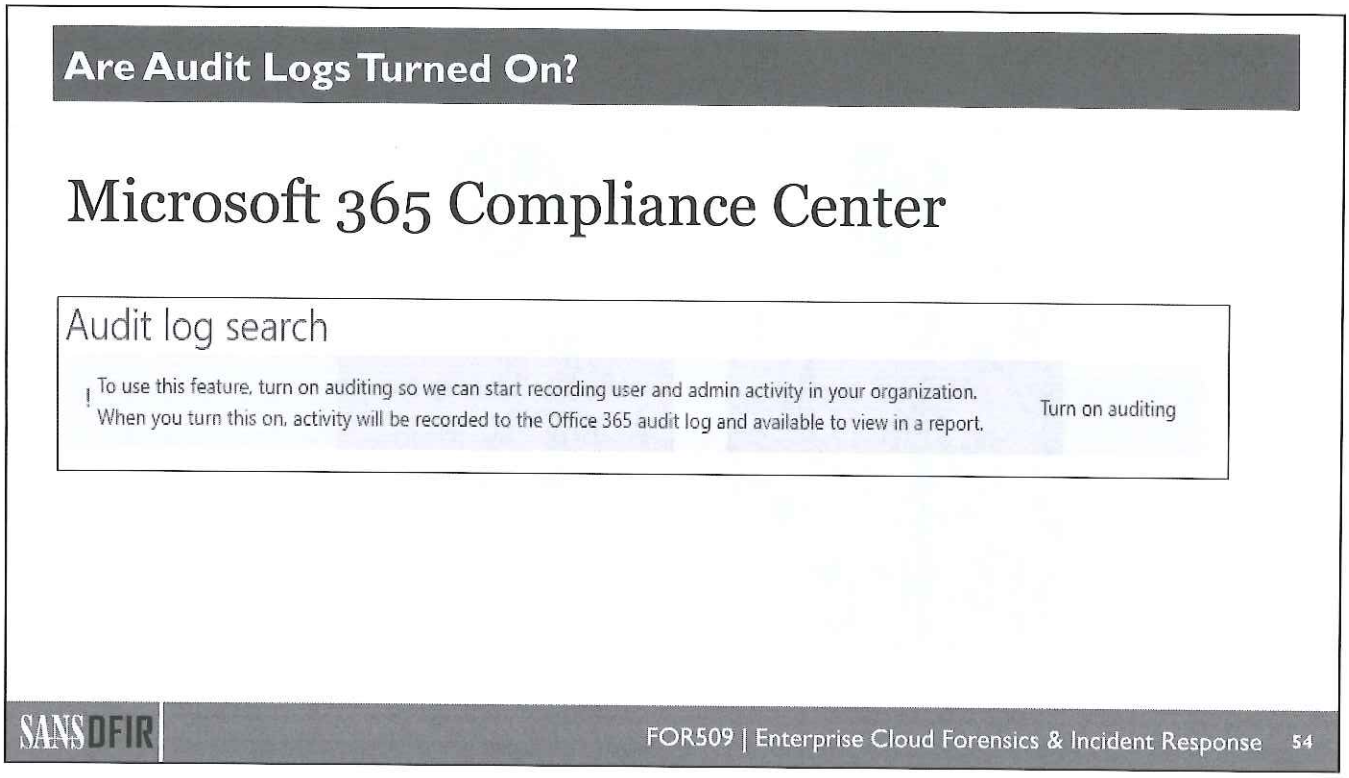


Audit log entries can take between 30 minutes and up to 24 hours before they are displayed in the search results.[1]

In practice, the time is far from exact. It also depends whether the data is being extracted via the Microsoft 365 Compliance Center, PowerShell, or API.

References:

[1] https://for509.com/search-ual



The Unified Audit Log (UAL) is not turned on by default. Typically, mature organizations turn this on during initial deployment, but smaller and less mature ones may not be aware of this requirement. Not having the UAL available will make your investigation significantly more complicated.

The UAL can be enabled in one of two ways:

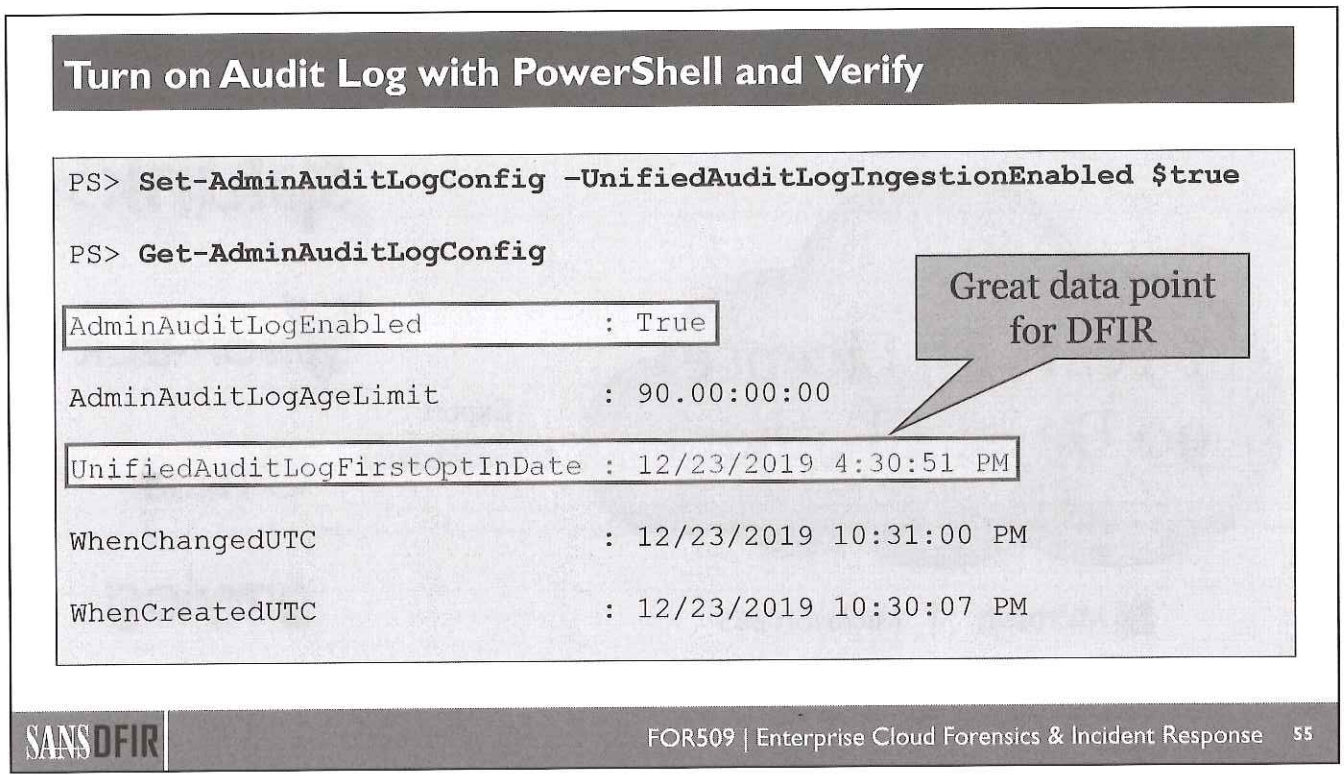
1. In the Microsoft 365 Compliance Center as shown in this slide
2. Using the PowerShell cmdlet `Set-AdminAuditLogConfig` as shown in the next slide

An interesting fact is that all actions taken in the Microsoft 365 Compliance Center are executed as PowerShell cmdlets. As such, to turn on auditing as shown in option 1, you must be assigned the Audit Logs role in Exchange Online in addition to the roles given to you in Microsoft 365 Compliance Center.[1]

In most environments, this administrative task should be handled by someone with Global Admin privilege.

References:

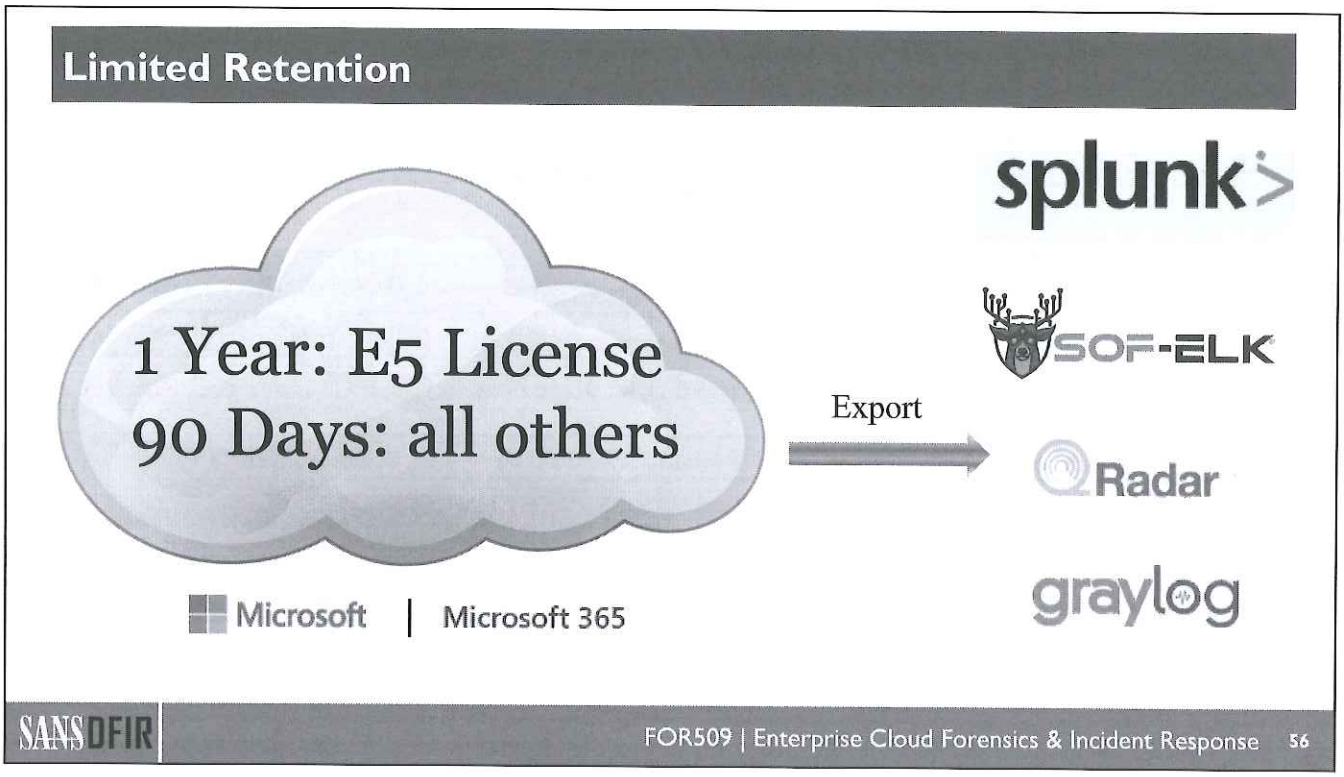
[1] https://for509.com/auditlog



While the audit log can be turned on via the Microsoft 365 Compliance Center, most administrators are likely to accomplish this task with PowerShell. In this slide, we show the PowerShell cmdlet to turn on the audit log and the cmdlet to verify that it's been correctly turned on.

The cmdlet `Get-AdminAuditLogConfig` will confirm that audit logging has been enabled and provide some very interesting information. From a forensics point of view, you should record the date in the field "UnifiedAuditLogFirstOptInDate" and compare that date with the timeframe of any incident you might be investigating.

The dates "WhenChanged" and "WhenCreated" may also be useful if an admin stopped and started the audit log in order to make sure that their misdeed wasn't recorded.



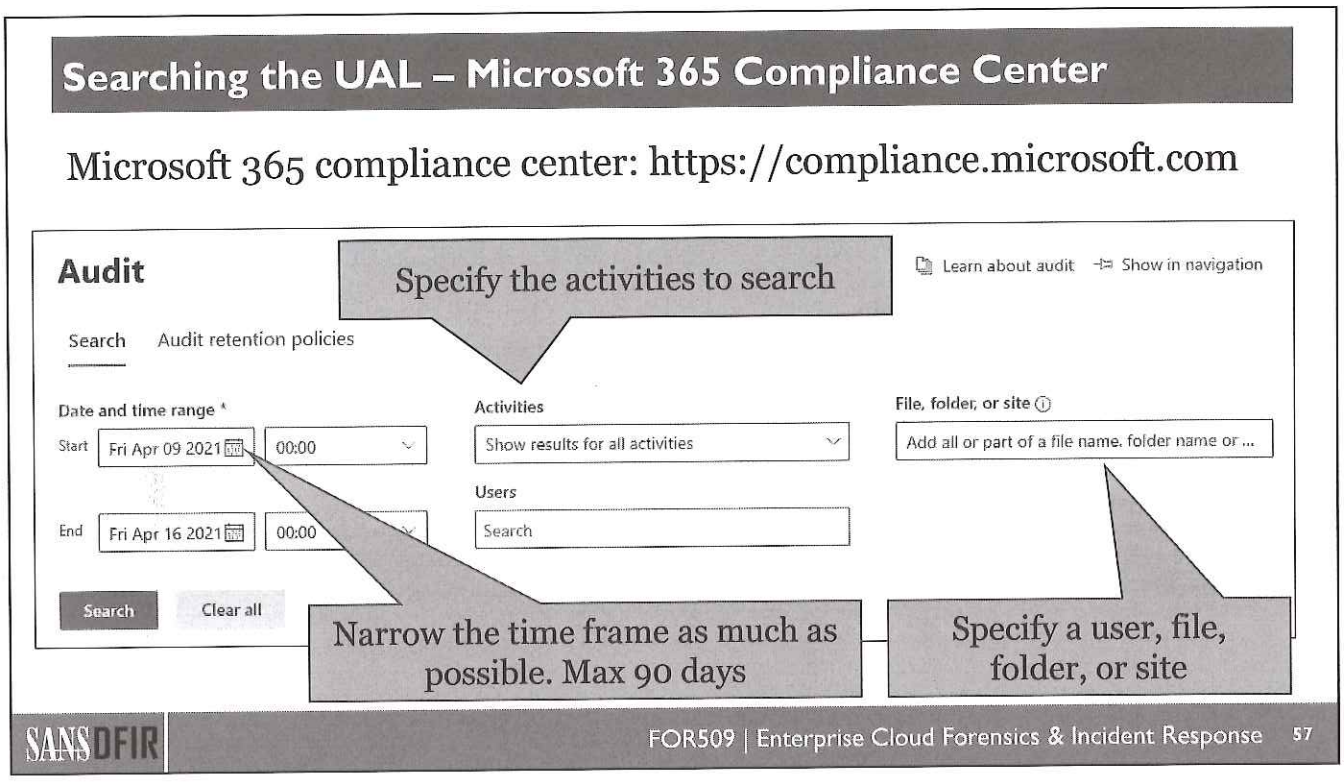
By default, advanced audit in Microsoft 365 will retain all Exchange, SharePoint, and Azure Active Directory audit records for 1 year. This default policy can't be modified. This default policy only applies to certain record types as documented in reference [1].

Further, the default policy retention of 1 year only applies to users who are assigned a Microsoft E5 license. For all other licenses, audit records are only retained for 90 days.

We expect most large organizations to export audit records to a third-party security information and event management (SIEM) application. In later slides, we will show you different methods to export this data. In this course, we will use SOF-ELK to analyze the UAL.

References:

[1] https://for509.com/logretention



As previously mentioned, there are three methods to search the UAL: Microsoft 365 Compliance Center, PowerShell, and API. The easiest is the Microsoft 365 Compliance Center which provides a graphical user interface.

To start a search, provide any of the four pieces of information:

1. Activities to search for
2. Timeframe
3. Specific Users
4. Specific file, folder, or site

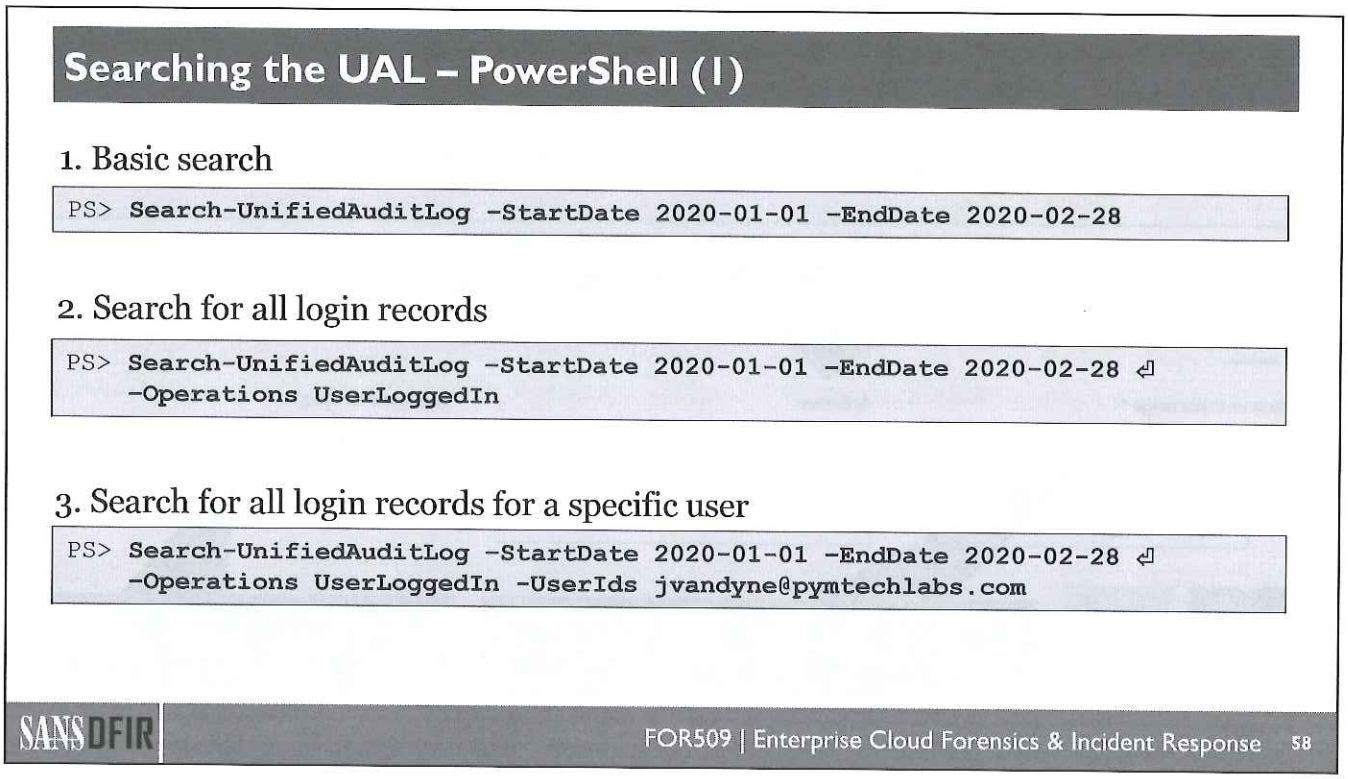
This search can be quite slow, and it's highly recommended to be as specific as possible. Microsoft may also rate limit the data extraction which can result in incomplete information. This method of searching the UAL is only recommended for small data sets.

Unfortunately, the output on the screen is not very useful and therefore the information is best when downloaded as a CSV file. However, the csv file is itself very difficult to use as all the information is contained in a single field called AuditData. This is the reason why a tool like SOF-ELK is needed.

The UAL log contains a large number of user and admin activities for each application. Reference [1] contains an exhaustive list of these activities and their description.

References:

[1] https://for509.com/search-ual



The second method to search the UAL is to use the PowerShell cmdlet `Search-UnifiedAuditLog`.[1] This cmdlet has many options, and we will provide a few examples here.

Example 1

In its simplest form, the UAL can be searched by simply providing a start date and an end date. As we have previously discussed, depending on your tenant's license you may be able to search up to 1 year (E5 license) or 90 days (all other licenses).

Example 2

The first example is likely to provide way too much data. Each audit log has a field called "Operations" which specifies the type of action being recorded. For example, the operation "UserLoggedIn" will record authentication attempts to Azure AD.

Example 3

We may not be interested in every authentication attempt during the specified period. We may only be looking for a specific user. In that case, we would use the option "UserIds" and specify the name of the user we are looking for.

References:

[1] https://for509.com/search-unifiedauditlog

Searching the UAL – PowerShell (2)

4. Search for all failed logins and export to csv (increase to maximum number of results)

```
PS> Search-UnifiedAuditLog -StartDate 2020-01-01 -EndDate 2020-02-28 ⏎
    -Operations UserLoginFailed -ResultSize 5000 -SessionCommand ⏎
    ReturnLargeSet | Export-Csv -Path "c:\data\userloginfailed.csv"
```

5. Search for all events and return results as JSON

```
PS> Search-UnifiedAuditLog -StartDate 2020-01-01 -EndDate 2020-02-28 ⏎
    -SessionCommand ReturnLargeSet -ResultSize 5000 | Select-Object ⏎
    -ExpandProperty AuditData
```


Example 4

In this example, we changed the Operations parameter to "UserLoginFailed" in order to look for failed login attempts. By default, `Search-UnifiedAuditLog` will only return 100 results. The parameter "-ResultSize 5000" will give us the maximum number of records that PowerShell can return. If you don't get as many results as expected, try adding the option "-SessionCommand ReturnLargeSet".

Since we expect a large dataset, we will store it in a csv (comma separated values) file by using the cmdlet `Export-Csv`.

Example 5

One of the advantages of PowerShell is that results are returned as objects. As such, we can choose specific objects from the results and discard everything else. The UAL stores all the relevant information in a field called "AuditData". In this example, we are extracting that field as a JSON object.

These are just 5 examples to show the flexibility of using PowerShell to extract information from the UAL. You may find these type of searches appropriate when conducting a small, highly focused investigation. However, in the normal course of business, the UAL should be exported on a continuous basis to a SIEM via the Microsoft 365 API.

UAL - Import into SOF-ELK

1. Export data to csv file (portal or PowerShell)
   - The Logstash parser was written to import data from a csv file (rather than JSON) to be compatible with both the portal and PowerShell
   - The format is a bit different between the portal and PowerShell, but the Logstash parser will accommodate both versions
2. Copy file to the Logstash folder
   - The Logstash folder for the Microsoft 365 log is `/logstash/office365`
3. Wait a few minutes, csv files takes a bit longer to process than JSON files

The UAL can be exported either through the Microsoft 365 Compliance Center portal or via PowerShell as previously described. The csv file format is a little bit different between the two, but the Logstash parser was written to accommodate both formats.

PowerShell can also export the UAL in JSON format which is more efficient to process. However, we didn't write a Logstash parser for this format at this time.

The export file should then be copied to the /logstash/office365 folder for processing.

Before being known as Microsoft 365, the office suite was called office365, hence the name of the directory and index in Kibana.

You will get the opportunity to practice these steps in lab 1.2.

Searching the UAL – API

Four steps to call the API:

1. Register your application in Azure AD
2. Get Microsoft 365 tenant admin consent
3. Request access tokens from Azure AD
4. Call the Microsoft 365 Management APIs

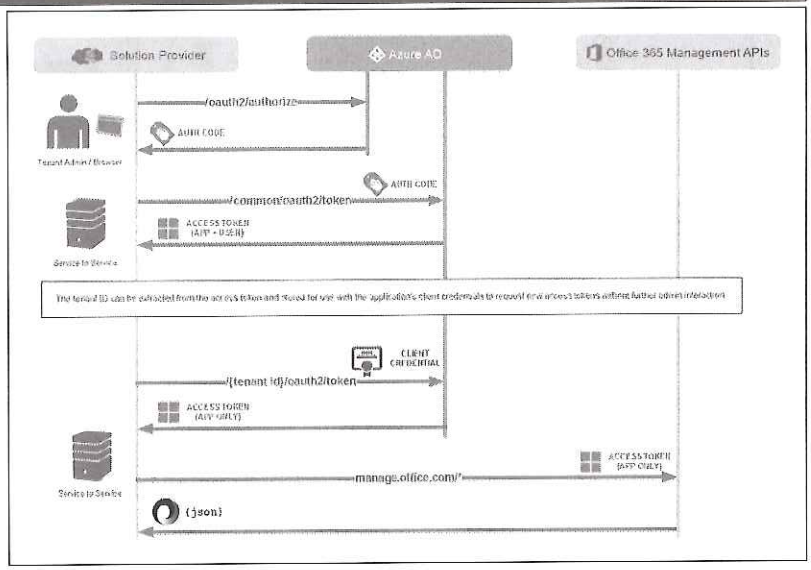


Image Credit: Microsoft (see reference 1 in the notes)

Most companies will want to export the UAL to their SIEM on a continuous basis. This is achieved by using the Microsoft 365 Management APIs.[1]

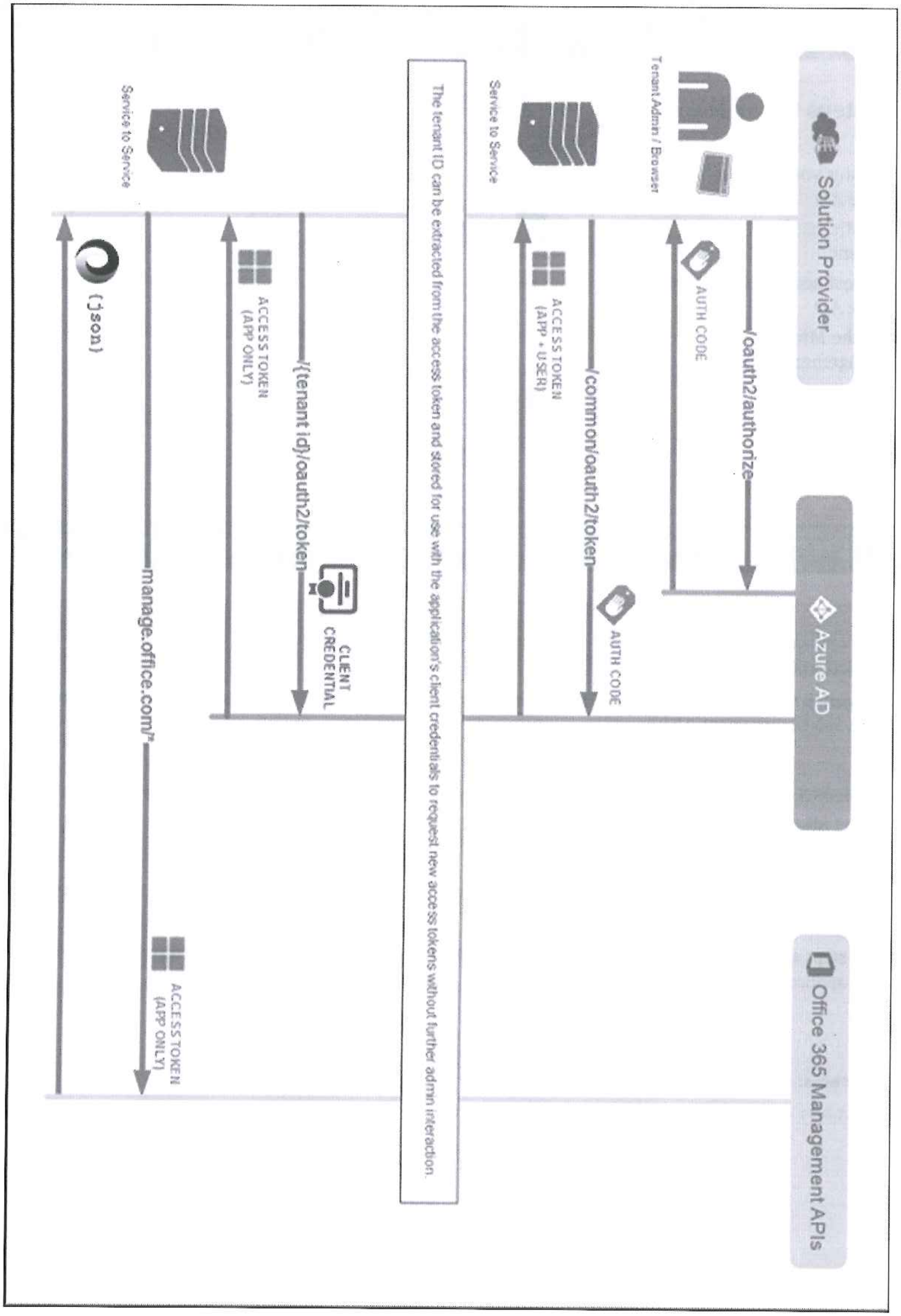
Configuring the API is beyond the scope of this class. At a high level, there are four key steps to enabling the Office 365 Management API:

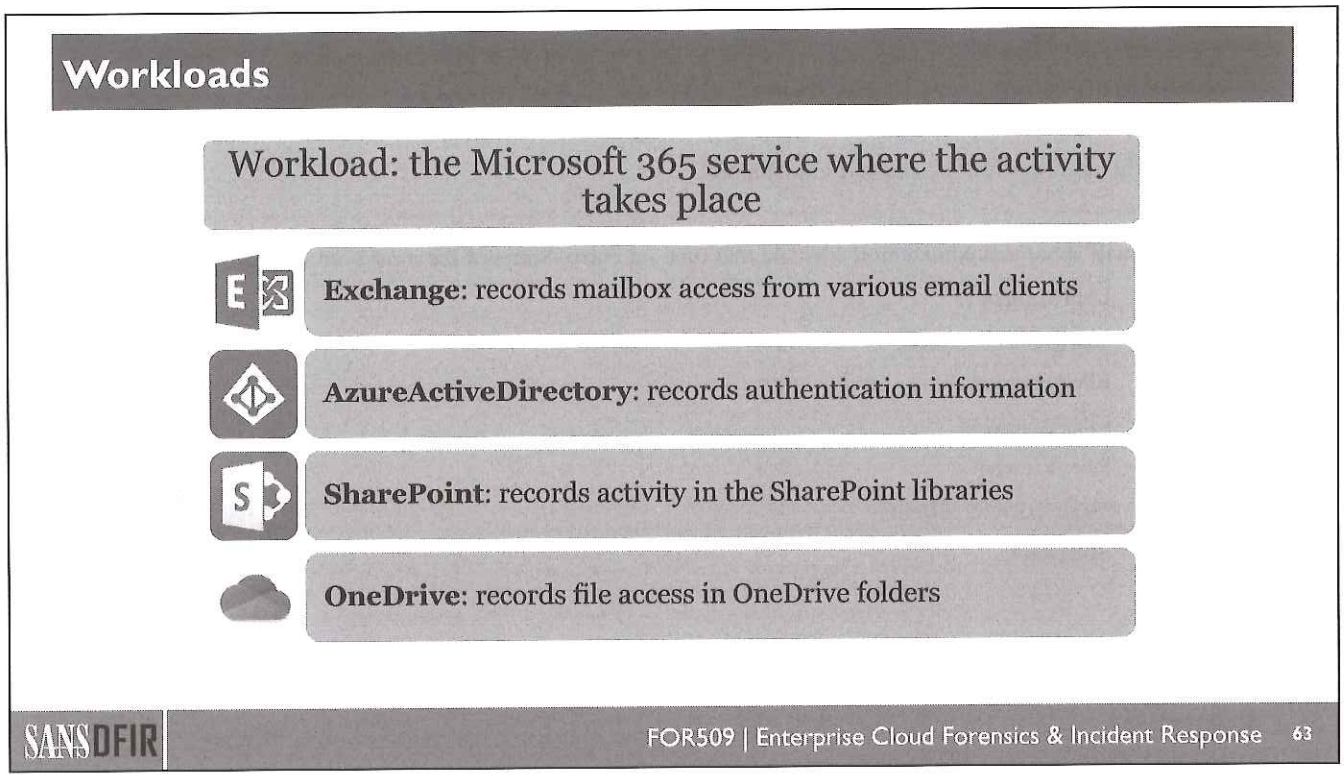
1. Register your application in Azure AD
2. Get Microsoft 365 tenant admin consent
3. Request access tokens from Azure AD
4. Call the Microsoft 365 Management APIs

These steps should be performed by your Global Admin in coordination with the team responsible for the SIEM application.

References:

[1] https://for509.com/ualapi





Each UAL entry contains a wealth of information. As a matter of fact, the quantity of information can be overwhelming.[1]

Each UAL entry contains a large number of fields. The most important field is called "workload". Microsoft uses the term "workload" to describe which Microsoft 365 service wrote the log entry. The primary workloads you will see are:

- AzureActiveDirectory
- Exchange
- SharePoint
- OneDrive
- MicrosoftTeams
- SecurityComplianceCenter

Some fields are found in most log entries while others are unique to specific workloads. The thing to keep in mind during your investigation is that fields can be named differently from one workload to another.

For example, the Exchange workload records the IP address in a field called "ClientIPAddress" while the AzureActiveDirectory workload records the IP address twice in fields called "ActorIPAdress" and "ClientIP". As such, you must be very careful when you filter the data.

When looking at a log for the first time, it's best to focus on a few key fields to narrow down the search. Suggested key fields are: Time, UserId, Workload, Operations, ResultStatus.

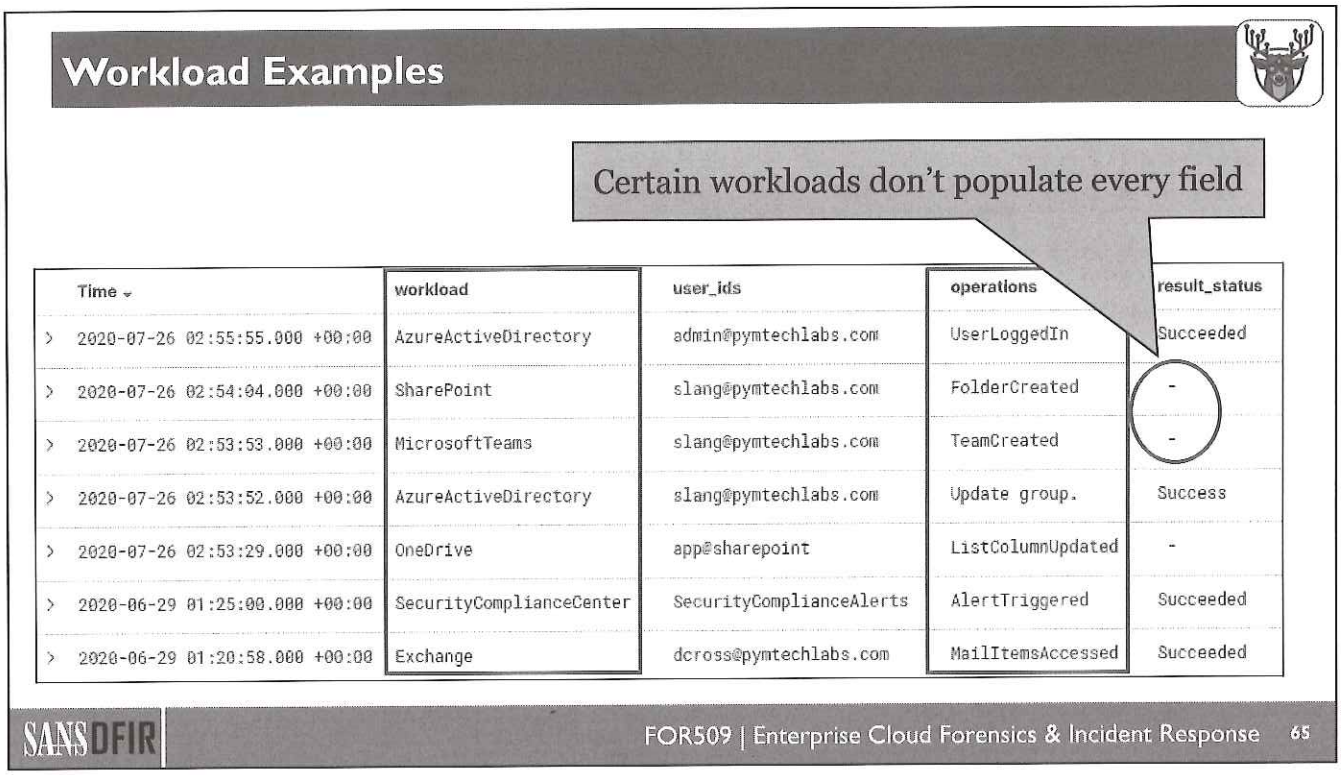
Time, UserId, and ResultStatus are self-explanatory.

Operations will show you what action was taken in that log entry. Some of the most common are:

- UserLoggedIn
- MailItemsAccessed
- FileAccessed
- FilePreviewed
- PageViewed
- MoveToDeletedItems
- SoftDelete

References:

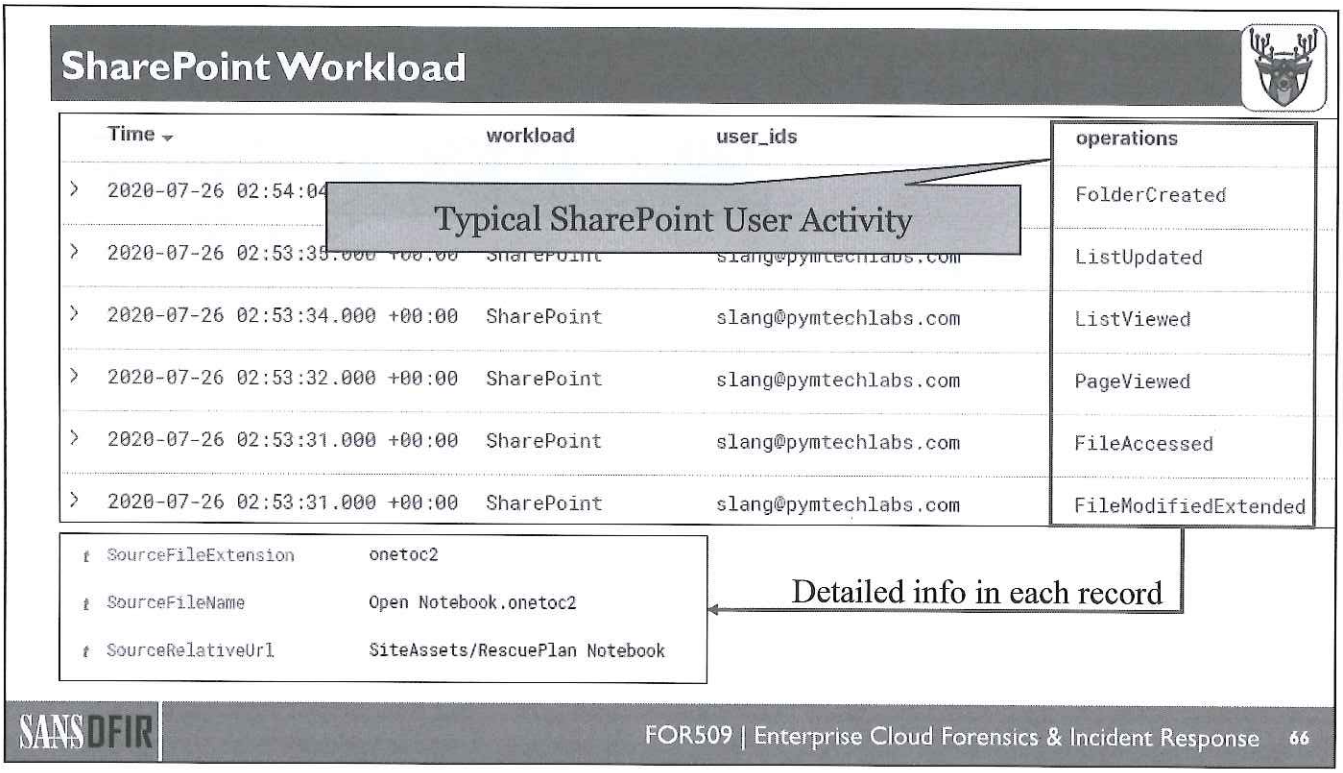
[1] https://for509.com/ual-properties



In this slide, we used SOF-ELK to filter the UAL to show 5 keys fields: time, workload, user_ids, operations, and result_status.

These are random events picked from the UAL to give you an idea of what you may see once you import your data into SOF-ELK. Notice that the different workloads don't always populate every field and may use different terminology, for example success versus succeeded.

In the next few slides, we will show more detailed examples of the SharePoint, Teams, and OneDrive workloads. We will then explore the Exchange workload in more detail and conclude with the Azure Active Directory workload which is relevant to both Microsoft 365 and Azure.

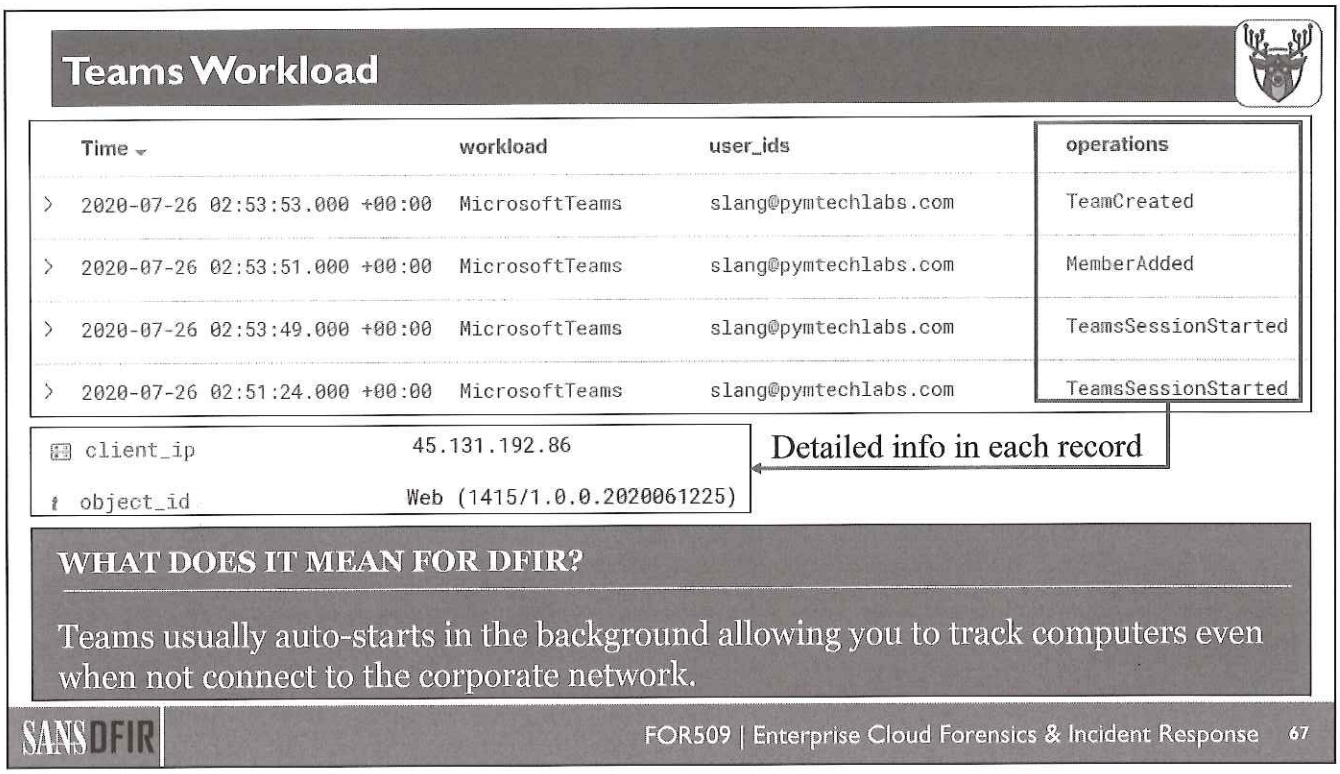


This slide is an example of typical SharePoint user activity. Some of the more frequent operations you will see related to SharePoint are:

Operation	Description
FolderCreated	User creates a folder on a site.
ListUpdated	User updates a SharePoint list by modifying one or more properties.
ListViewed	User views a list on a site.
PageViewed	User views a page on a site.
FileAccessed	User or system account accesses a file.
FileModifiedExtended	User continually modifies a file (up to 3 hours).

The name given to each operation is pretty vague making it difficult to ascertain the actual activity of the user. It's important to review the entire log entry to get a full picture. In this particular example, the "FileAccessed" and "FileModifiedExtended" were generated when Scott Lang opened a OneNote notebook that's embedded in the SharePoint site and modified it.

While not depicted on this slide, you will also see a number of activities where the user id is set to app@sharepoint. This means the system performed the activity on behalf of the user who initiated the action. Unfortunately, such a generic user id makes our job more difficult.



Microsoft Teams is constantly adding new features and the latest list of operations logged are documented in reference [1].

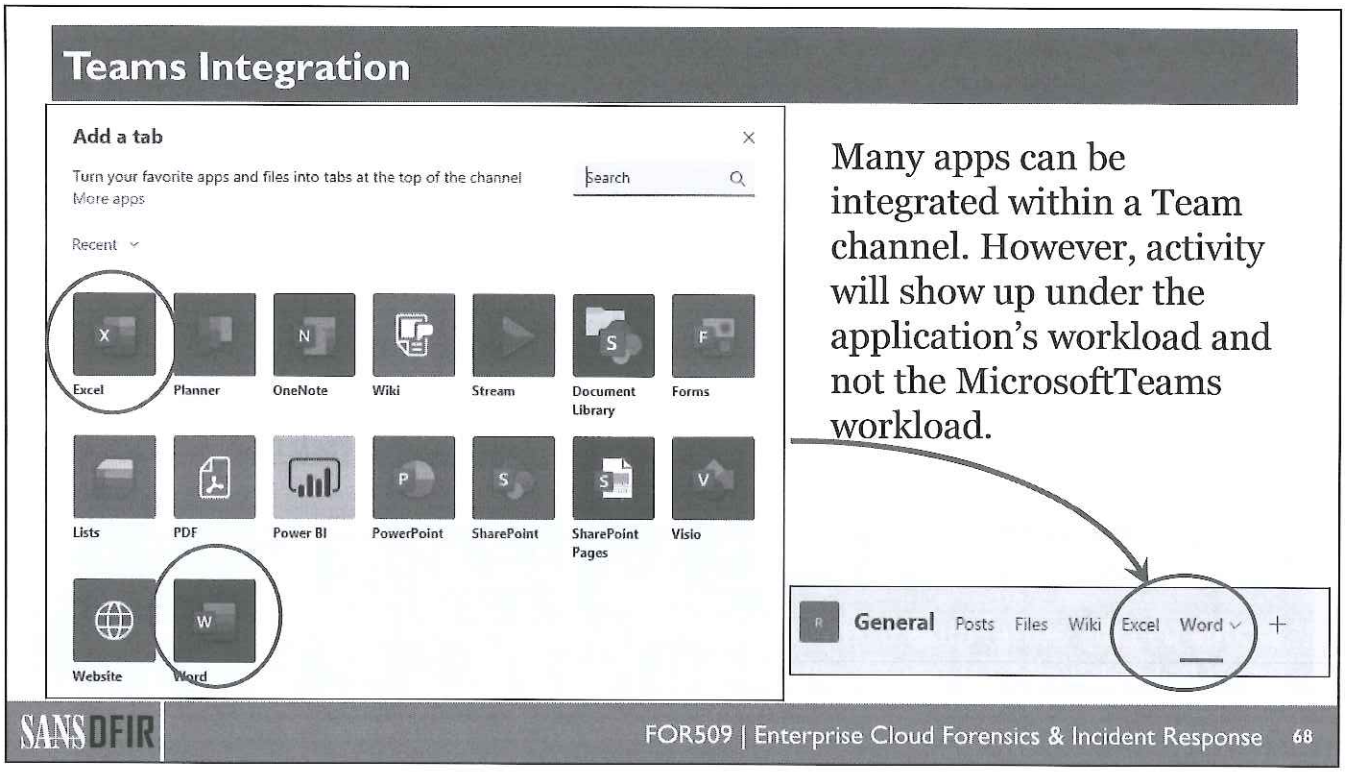
In large environments the most common operation is TeamsSessionStarted. This operation is interesting because it will provide the IP address and client string (called object_id) of the computer connected to the Teams session. Some of the client strings observed are:

Object_id	Description
Web (1415/1.0.0.2020061225)	Web access
Android (1416/1.0.0.2020091301)	Android mobile device
TeamsGraphService (Unknown)	Microsoft Graph API
Unknown (Unknown)	Possibly iPhone?

From a DFIR perspective, remember that Teams is frequently set to auto-start in the background when the computer boots up. As such, you may be able to get IP address information for computers even when they are not inside your corporate network. This could be very useful to track the whereabouts of a computer.

References:

[1] https://for509.com/ual-teams



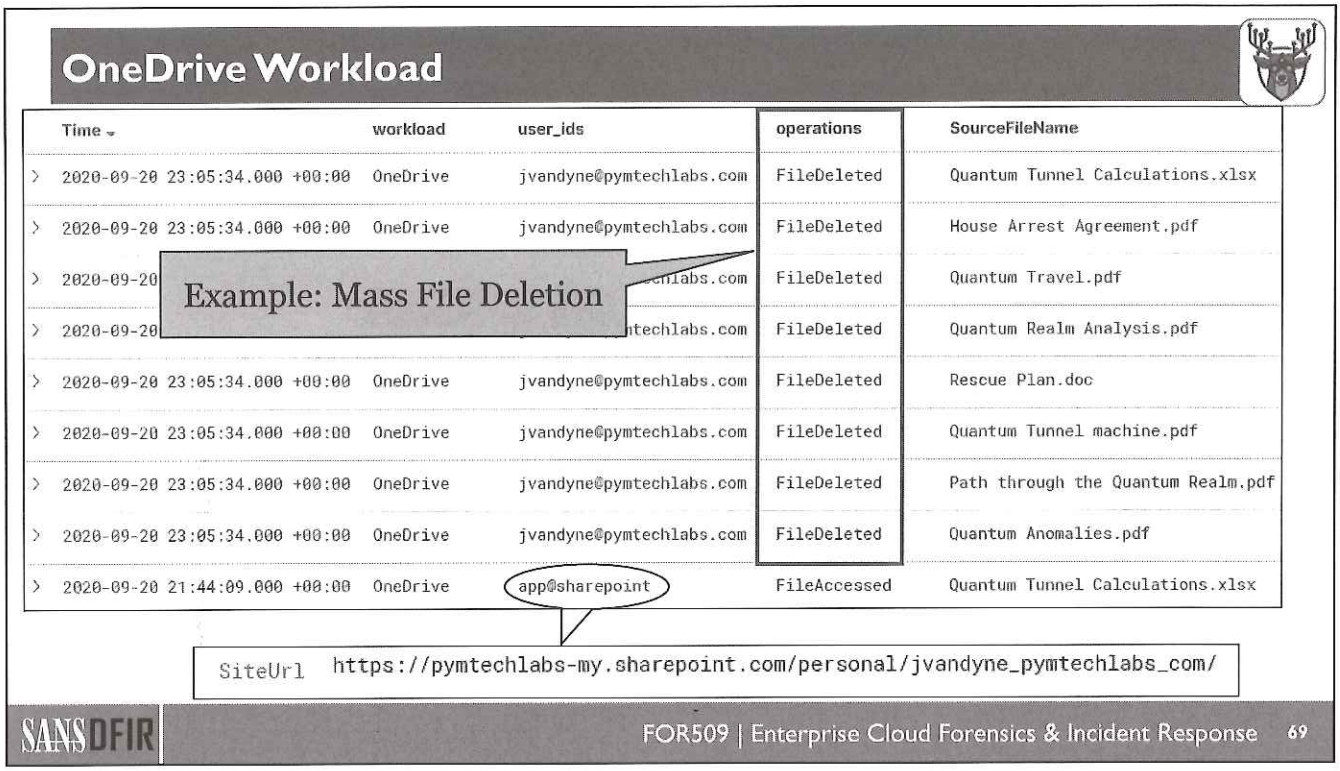
Many Microsoft 365 applications can be integrated within a Team channel and will show up as a new tab in the Teams channel.[1]

One of the consequences of integration is that the log entries show up under that application's workload and not the MicrosoftTeams workload.

Don't be surprised if you interview a user and they don't realize that another Microsoft 365 application was invoked within their team's channel. From their point of view, they were "just using teams".

References:

[1] https://for509.com/teamsintegration



OneDrive for business (simply called OneDrive) allows users to automatically sync their files to the Microsoft 365 cloud. This is a treasure trove of information for our investigation. While the user thinks of their file as being local on their computer, a copy is actually made to the cloud and every access is logged.

The situation shown above may happen when a user decides to leave the company. They may check the content of a file as shown by the FileAccessed operation followed by a mass deletion.

As mentioned in the SharePoint Workload example, you may sometimes see activities where the user id is set to app@sharepoint. This means the system performed the activity on behalf of the user who initiated the action. By looking at the name of the file and the siteurl, you may be able to deduce who performed the operation. Unfortunately, that's not always possible for large sites with a lot of activity.

You will also notice that OneDrive is using SharePoint in the background and the OneDrive folder is represented by a URL.

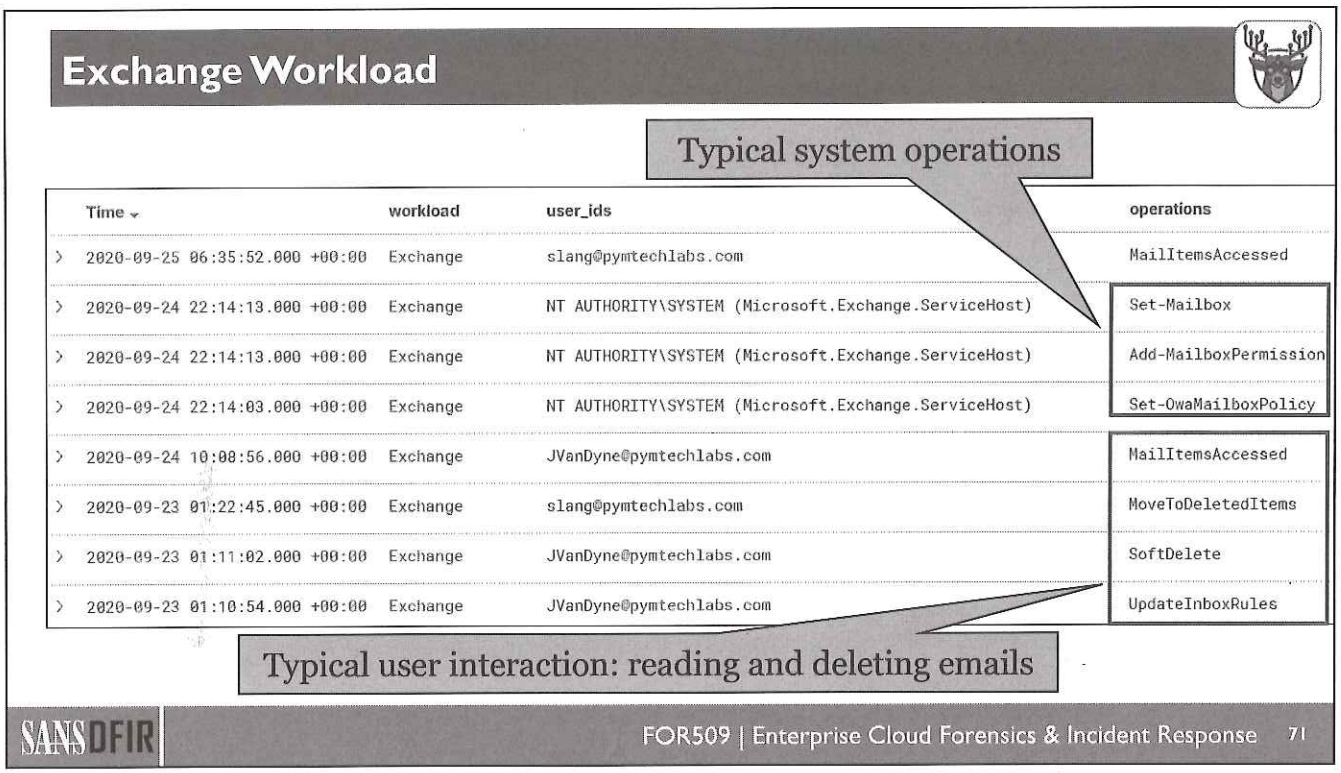
SharePoint Online & OneDrive for Business

- SharePoint Online (SP) and OneDrive for Business (ODfB) are related components of Microsoft 365 with overlapping architecture and features
- User actions in either SP or ODfB can result in entries under both applications
- Many categories of log activities

Activities	SP	ODfB
File and page activities	✓	✓
Folder activities	✓	✓
SharePoint list activities	✓	
Sharing and access request activities	✓	✓
Synchronization activities	✓	✓
Site permissions activities	✓	
Site administration activities	✓	
Sensitivity label activities	✓	

SharePoint Online (SP) and OneDrive for Business (ODfB) are so closely related that it's not always clear under which workload the log entries will show up.

The audit log search feature in the Microsoft 365 Compliance Center provides the options to search for groups of activities (or specific activities). The ones related to SP and ODfB are shown in the table above. However, it doesn't mean that the user performed the activity in that application.

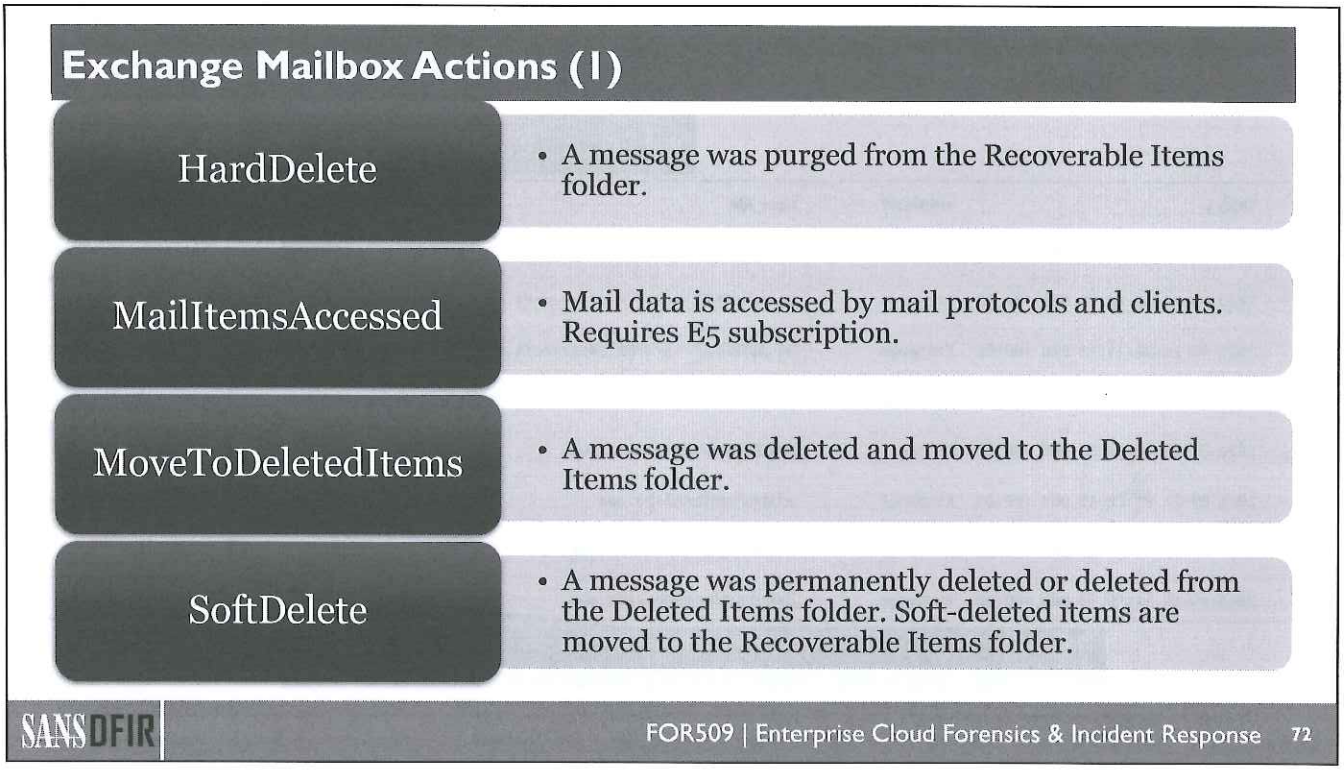


Some typical user and system Exchange operations are:

Operation	Description
MailItemsAccessed	Messages were read or accessed in mailbox. This activity is only logged for users with an E5 license.
Set-Mailbox	Change mailbox configuration parameters.
Add-MailboxPermission	Modify the permissions assigned to a mailbox.
Set-OwaMailboxPolicy	Configure OWA mailbox policies.
MoveToDeletedItems	A message was deleted and moved to the Deleted Items folder.
SoftDelete	A message was permanently deleted or deleted from the Deleted Items folder.
UpdateInboxRules	A mailbox owner modified an inbox rule in the Outlook client.

Under the SharePoint and OneDrive workloads, the Microsoft 365 will log system events under the app@sharepoint user id. For the Exchange workload, the system will use the NT AUTHORITY\SYSTEM user id.

The most important operation is the MailItemsAccessed as it will provide us detailed information about message activity. Unfortunately, that operation is only logged for mailboxes with an E5 license. In the next slides we will explore the information we can obtain from the MailItemsAccessed log entries.



There are a number of mailbox actions that can be logged. These actions may apply to user mailboxes, shared mailboxes, or group mailboxes. In addition, there are 3 logon types:

1. **Owner**: the mailbox owner
2. **Delegate**: a user who's been assigned the SendAs, SendOnBehalf, or FullAccess permission to another mailbox
3. **Admin**: the mailbox is accessed via the Microsoft 365 compliance center

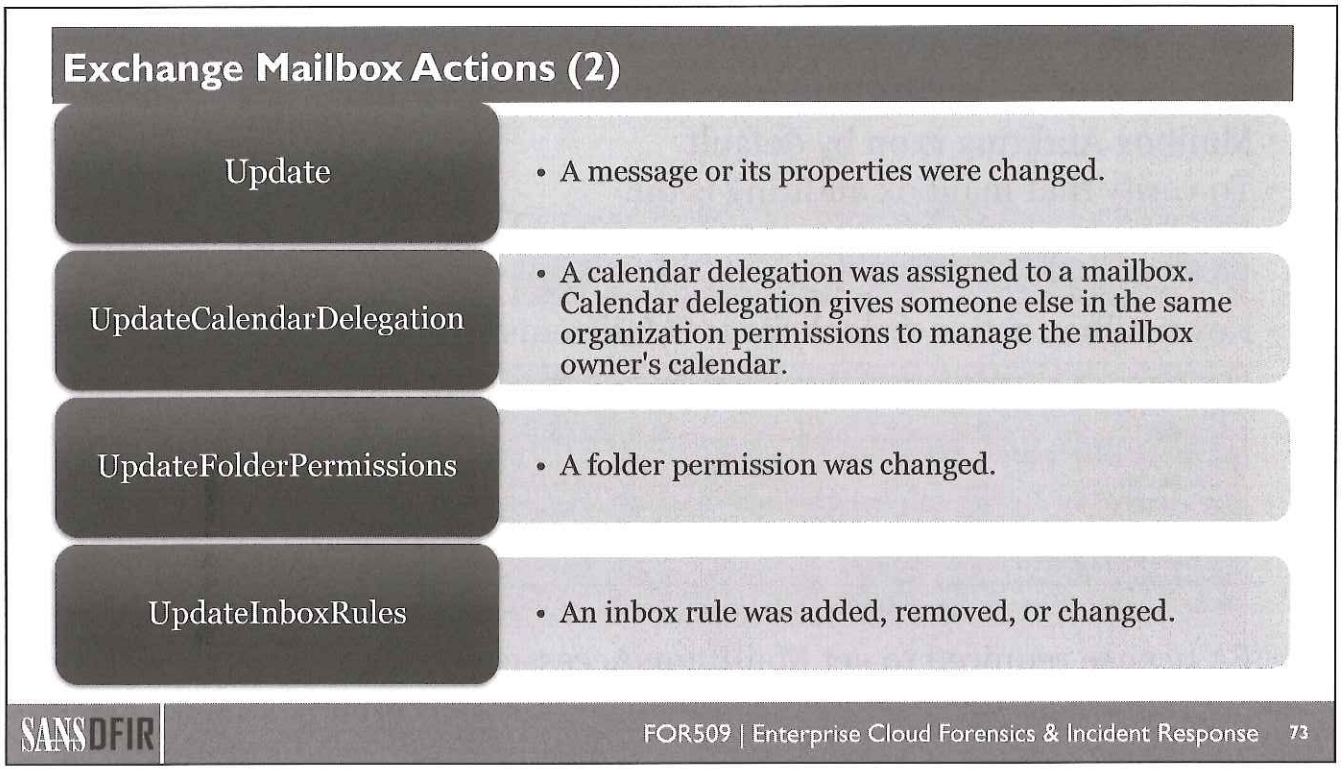
There are nuances as to which mailbox action is logged for which logon type, but we will focus on the **owner** logon. These are the default actions logged for owner logon:

- **HardDelete**: A message was purged from the Recoverable Items folder.
- **MailItemsAccessed**: Mail data is accessed by mail protocols and clients. Requires E5 subscription.
- **MoveToDeletedItems**: A message was deleted and moved to the Deleted Items folder.
- **SoftDelete**: A message was permanently deleted or deleted from the Deleted Items folder. Soft-deleted items are moved to the Recoverable Items folder.

The Microsoft documentation contains a detailed table with the various possible scenarios.[1]

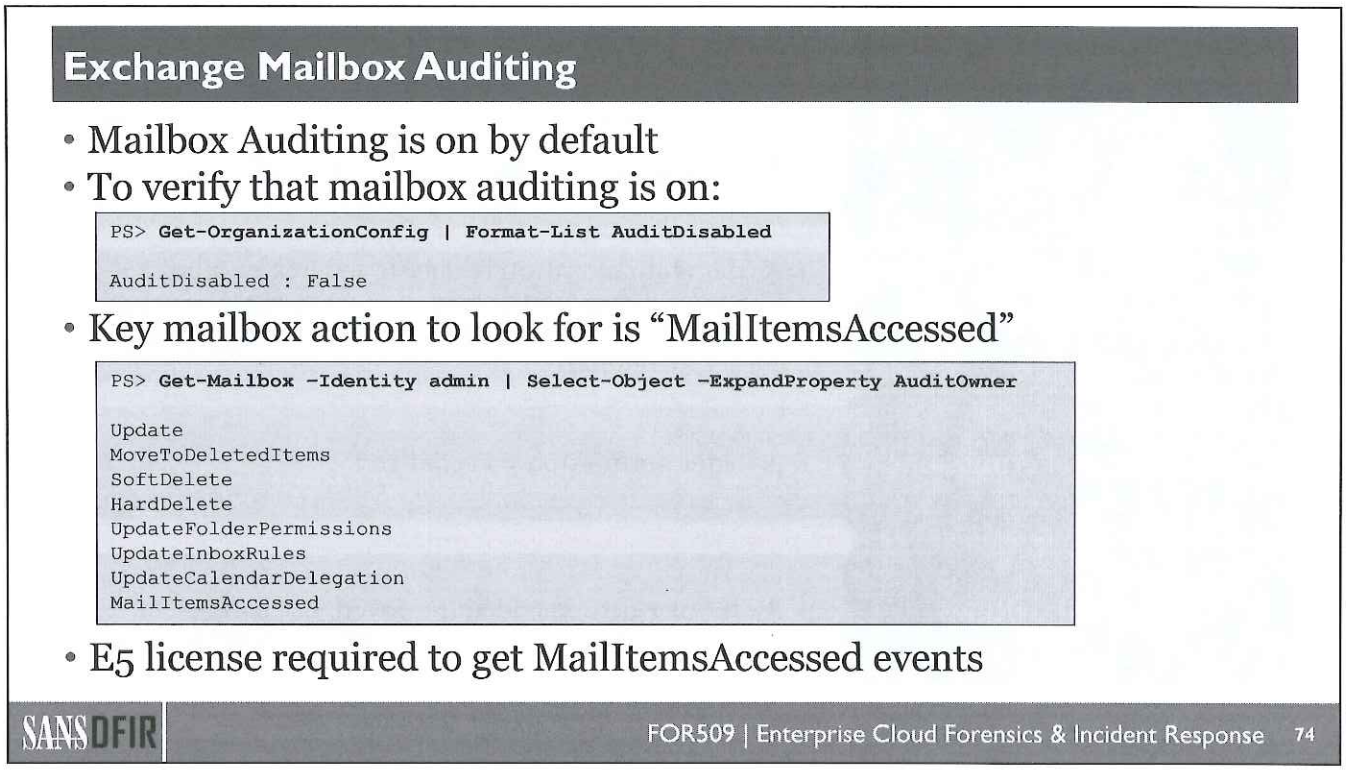
References:

[1] https://for509.com/mailboxauditing



Additional default actions logged for owner logon (continued from previous slide):

- **Update**: A message or its properties were changed.
- **UpdateCalendarDelegation**: A calendar delegation was assigned to a mailbox. Calendar delegation gives someone else in the same organization permissions to manage the mailbox owner's calendar.
- **UpdateFolderPermissions**: A folder permission was changed.
- **UpdateInboxRules**: An inbox rule was added, removed, or changed.



One of the most interesting workloads is Exchange. Many incidents will involve Exchange and as such it's critical to make sure the Exchange logs are enabled.

As of January 2019, the Exchange logs should be turned on by default for newly created tenants.[1] Be sure to confirm the correct configuration for tenants created prior to January 2019.

The PowerShell command `Get-OrganizationConfig | Format-List AuditDisabled` will help you confirm that audit logs are not disabled, hence they are enabled. The double negative can be confusing: False means that auditing is turned on.

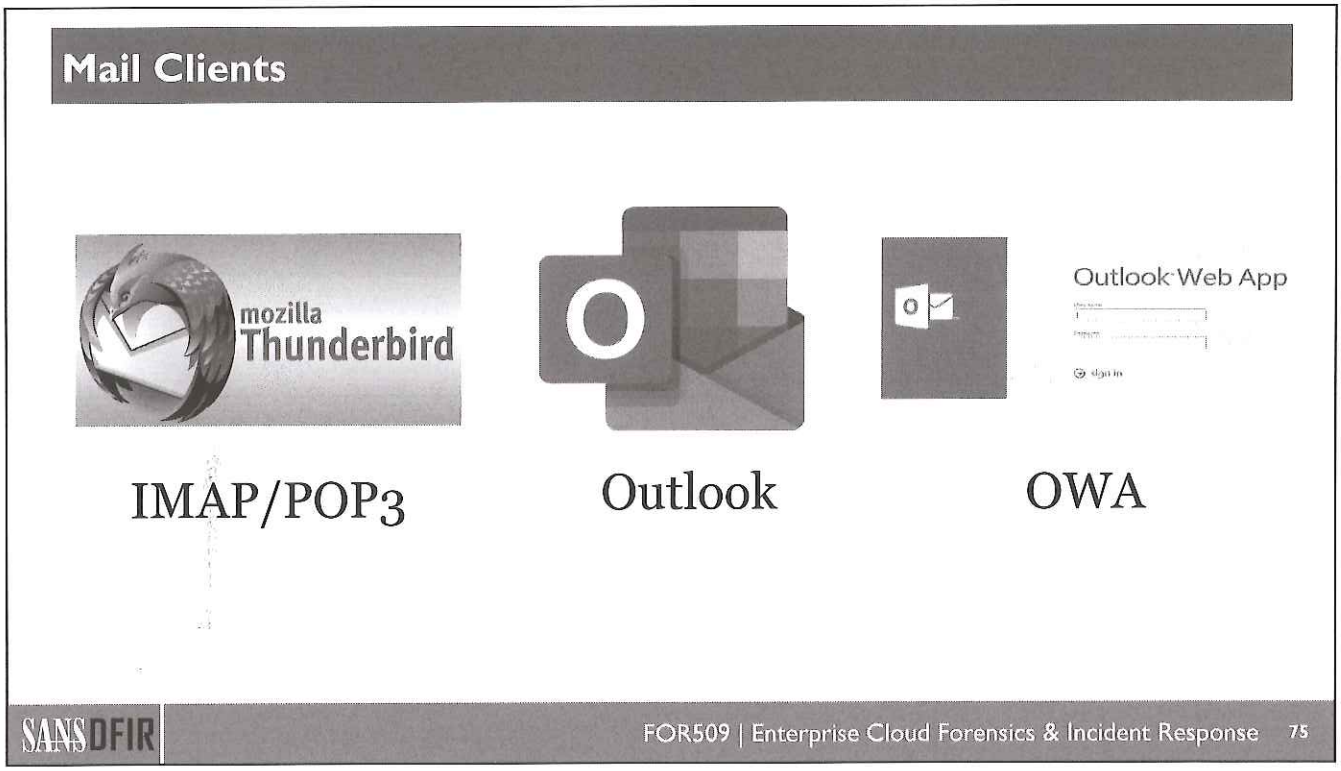
Mailbox auditing can be turned on either at the organization level or the individual mailbox level. If turned on at the organization level, then Microsoft 365 will ignore attempts to turn off auditing at the mailbox level. The only exception is if the mailbox is configured for auditing bypass. This subtlety is important to understand in case you are working a case involving an administrator abusing their privileges. Said administrator could attempt to hide their bad actions by enabling mailbox audit bypass. Hence, it's not sufficient to check mailbox auditing at the organization level.

To conduct an effective investigation involving Exchange, the key field is "MailItemsAccessed". Unfortunately, that field is only available if your tenant is licensed at the E5 level [2].

To make sure that a mailbox is being audited for "MailItemsAccessed", use the PowerShell command:
`PS> Get-Mailbox -Identity <name of mailbox> | Select-Object -ExpandProperty AuditOwner`

References:

[1] https://for509.com/onbydefault
[2] https://for509.com/advancedaudit



Messages can be retrieved from mail servers in various ways. We have two elements to consider: the protocol and the mail client. The main email protocols are POP3 and IMAP. Examples of mail clients are Mozilla Thunderbird and Microsoft Outlook.

- IMAP is the Internet Message Access Protocol, and the specifications are defined in RFC 3501.[1]
  - While POP was designed for a single user to manage a single mailbox, IMAP's design allows for the management of a mailbox by multiple email clients. To accomplish that goal, IMAP will leave emails on the server unless explicitly configured otherwise.
  - The latest version is IMAP4.
  - IMAP servers listen on TCP port 143. When using IMAP over SSL, TCP port 993 is used.

- POP3 is the Post Office Protocol version 3.
  - This protocol is rarely seen today, although still supported by most mail clients.
  - The Post Office Protocol was first defined in RFC 918 (POP1). In 1985, POP2 was defined in RFC 937. The most common version is POP3 which was initially defined in 1988 with RFC 1081. The current versions of POP3 is defined in RFC 1939.[2]
  - Originally designed to connect to a mail server, download all emails, and delete them from the server. Many POP3 clients have implemented the option to leave the mail on the server.
  - POP3 servers listen on TCP port 110. When using SSL, TCP port 995 is used.

- Microsoft Outlook is primarily an email client which is part of the Microsoft Office suite. However, it also provides a calendar, task management, contact management, and many more features. Outlook is best known for connecting to Exchange servers. However, it also has the ability to connect to POP3 and IMAP servers.

- OWA is an acronym for either Outlook Web Access or Outlook Web App. It permits access to your mailbox via the web and removes the need for the desktop client. With the introduction of Microsoft 365, OWA is now part of the entire suite of Microsoft applications and is accessed via the office365.com portal.

Why the history lesson? Because the log entries are going to show different pieces of information depending on the mail client used to access the mailbox.

References:

[1] https://for509.com/rfc3501

[2] https://for509.com/rfc1939

MailItemsAccess – Sync vs. Bind access

- Sync: Outlook
  - Audit event only includes the mail folder being synced
- Bind: OWA, IMAP/POP3
  - Audit event includes each individual email messages
  - All bind operations within a 2-min interval are aggregated in a single audit record
- Throttling
  - Audit records will no longer be generated if more than 1,000 MailItemsAccessed in 24 hours
  - This is per mailbox
  - Only applies to Bind operations
  - OperationProperties will show a value of True under the key "IsThrottled"

There are two types of mailbox access: Sync and Bind.

Sync access is used by Outlook where entire folders are synced between Exchange in the cloud and Outlook on the desktop. In this case, we will not get information about individual emails.

Bind access is used by web clients such as OWA, IMAP and POP3. Each email is recorded in the audit log. In rare cases, if more than 1,000 MailItemsAccessed audit records are generated in less than 24 hours, Exchange Online will stop generating auditing records for MailItemsAccessed activity.

Email messages that were accessed are identified by their **internet message Id**.

Each log entry will have a number of interesting fields (they are subfields of the Auditdata field):

Property	Description
MailAccessType	Whether the access is a bind or a sync operation.
ClientInfoString	Describes protocol, client (includes version).
ClientIPAddress	IP address of the client machine.
SessionId	Session ID helps to differentiate attacker actions vs day-to-day user activities on the same account (in the case of a compromised account).
UserId	UPN of the user reading the message.
ParentFolder	The full folder path of the mail item that was accessed.
Logon_type	The logon type of the user who performed the action. The logon types (and their corresponding Enum value) are Owner (0), Admin (1), or Delegate (2).
MailboxUPN	The UPN of the mailbox where the message being read is located.

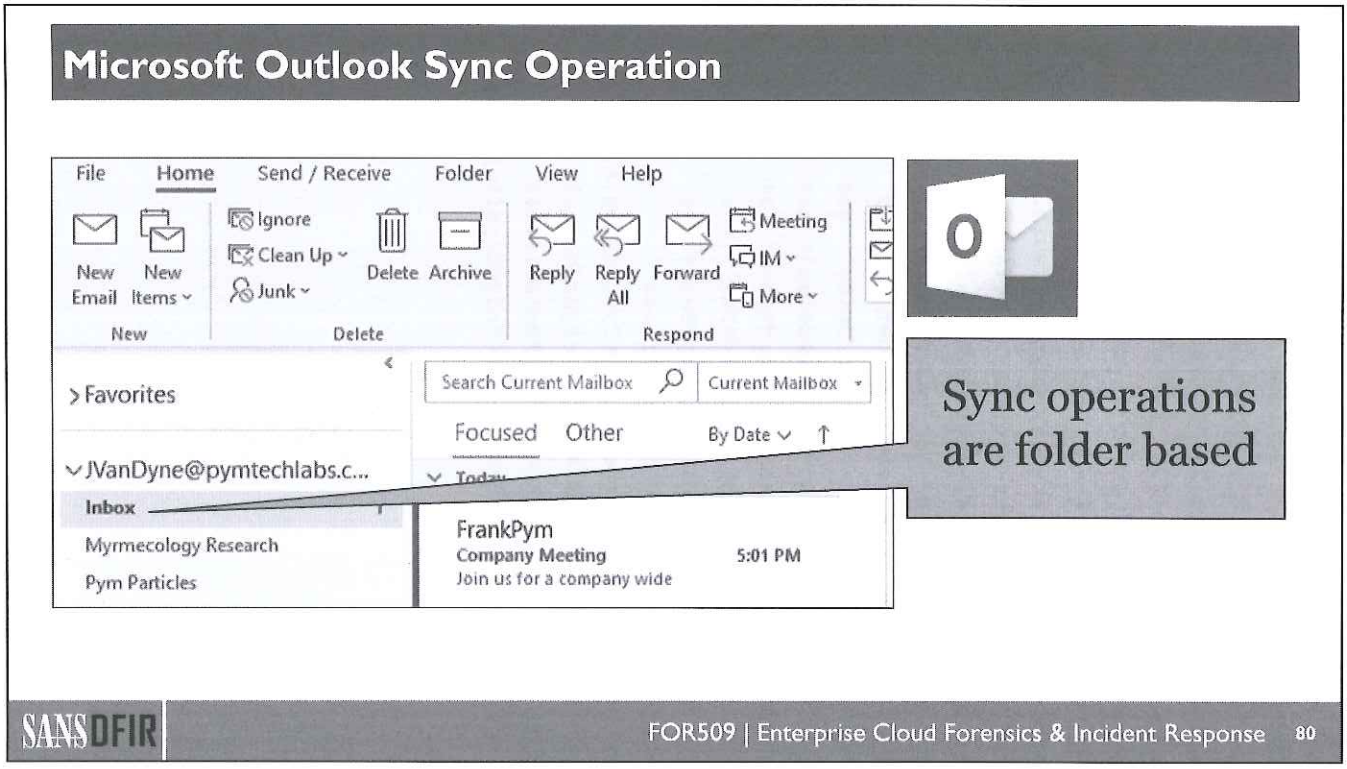
UPN stands for User Principal Name. For example, admin@pymtechlabs.com is a UPN where admin is the username and pymtechlabs.com is the domain.

ClientInfoString can provide some unique information about the type of device used to access the mail server. Rachel Moorehead has compiled an interesting list on Github[1] as shown on the next page.

References:

[1] https://for509.com/clientinfostring

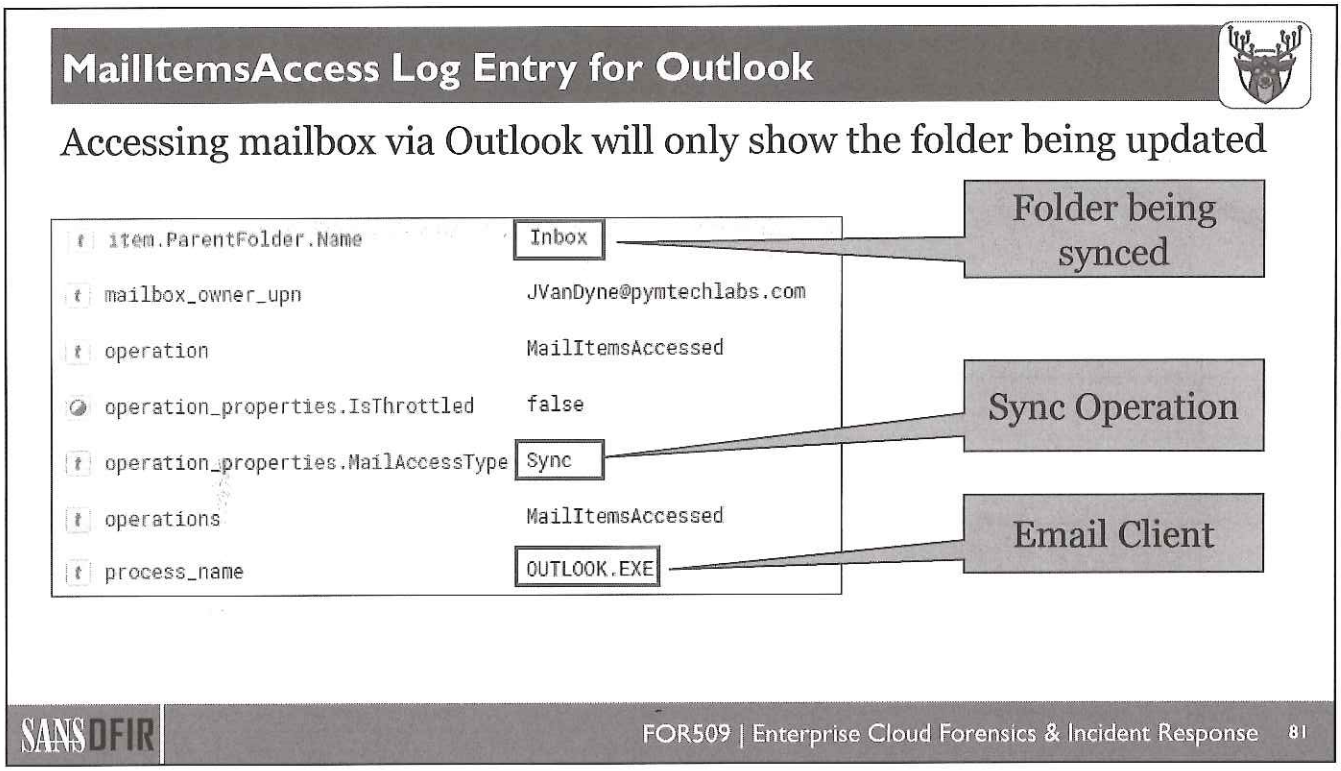
Outlook 2011	Client=WebServices;UserAgent=MacOutlook/14.3.2.130206 (Intel Mac OS X 10.8.3)
Outlook MAPI	Client=WebServices;UserAgent=Microsoft Office/14.0 (Windows NT 6.1; Microsoft Outlook 14.0.6129; Pro)
Thunderbird?	Client=WebServices;UserAgent=ExchangeWebServicesProxy/CrossSite/EXCH/14.16.0287.008/Mozilla/4.0
MacMail	Client=WebServices;UserAgent=ExchangeWebServicesProxy/CrossSite/EXCH/14.15.0129.007/Mac OS X/10.6.8
	Client=WebServices;UserAgent=Mac OS X/10.8.2 (12C2034); ExchangeWebServices/3.0 (157); Mail/6.2 (1499)
	Client=WebServices;UserAgent=Mac OS X/10.8.3 (12D78); ExchangeWebServices/3.0 (157); Mail/6.3 (1503)
iPhone 3	Client=ActiveSync;UserAgent=Apple-iPhone3C3/1002.329;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=Apple-iPhone3C1/1001.523;Action=/Microsoft-Server-ActiveSync/default.eas
iPhone 4	Client=ActiveSync;UserAgent=Apple-iPhone4C1/1002.142;Action=/Microsoft-Server-ActiveSync/default.eas
iPhone 5	Client=ActiveSync;UserAgent=Apple-iPhone5C1/1002.143;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=Apple-iPhone5C2/1002.329;Action=/Microsoft-Server-ActiveSync/default.eas
iPad 1	Client=ActiveSync;UserAgent=Apple-iPad1C1/902.206;Action=/Microsoft-Server-ActiveSync/default.eas
iPad 2	Client=ActiveSync;UserAgent=Apple-iPad2C1/1002.146;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=Apple-iPad2C2/1002.329;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=Apple-iPad2C5/1002.329;Action=/Microsoft-Server-ActiveSync/default.eas
Android	Client=ActiveSync;UserAgent=Android/4.2.1-EAS-1.3;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=Android/4.2.2-EAS-1.3;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=Android-EAS/0.1;Action=/Microsoft-Server-ActiveSync/default.eas
TouchDown	Client=ActiveSync;UserAgent=TouchDown(MSRPC)/8.1.00020/;Action=/Microsoft-Server-ActiveSync/default.eas
	Client=ActiveSync;UserAgent=TouchDown(MSRPC)/7.2.00016/;Action=/Microsoft-Server-ActiveSync/default.eas
DroidRazr	Client=ActiveSync;UserAgent=motorola-DROIDRAZR/1.0;Action=/Microsoft-Server-ActiveSync/default.eas
Samsung	Client=ActiveSync;UserAgent=SAMSUNGSCHS720C/2.3.6-EAS-1.2;Action=/Microsoft-Server-ActiveSync/default.eas
Outlook RPC	Client=MSExchangeRPC
Passive DAG	Client=CI (Content Indexing)
OWA	Client=OWA;Action=ViaProxy
	Client=WebServices;UserAgent=OwaProxy
IMAP	Client=IMAP4



Users will normally have multiple folders in their mailbox. In this example, JVanDyne has 3 different folders: "Inbox", "Myrmecology Research", and "Pym Particles". Since Outlook does a Sync operation, each folder will be synced in their entirety. Therefore, the log entries will only show the name of the folder being synced, not each individual messages. Our investigation is impacted by this sync behavior, as we won't be able to prove which individual messages were read solely based on the UAL. Other forensic techniques will need to be used.

You may also get failure events in the log for folders that fail to sync.

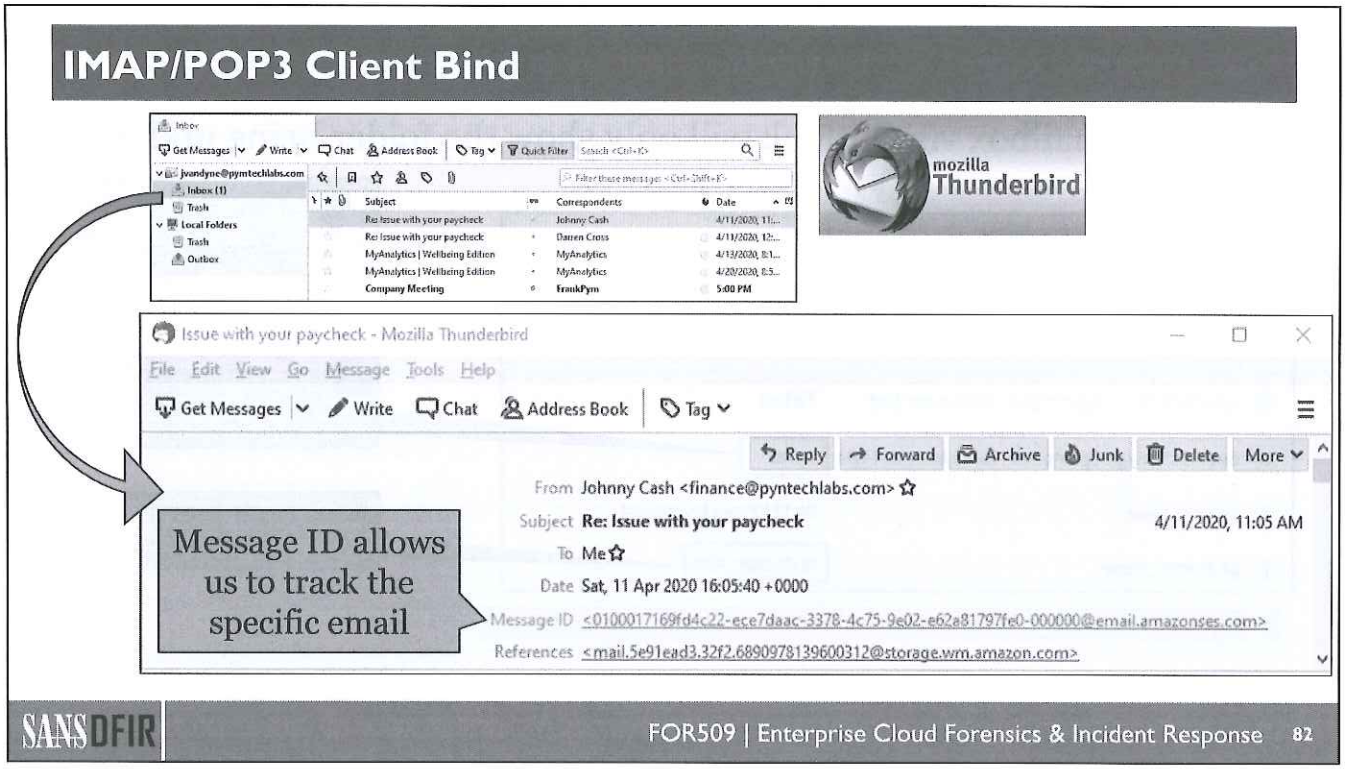
In the next slide, we will show a log entry for the Inbox folder.



This is an example log entry of a "Sync" access as seen in SOF-ELK.

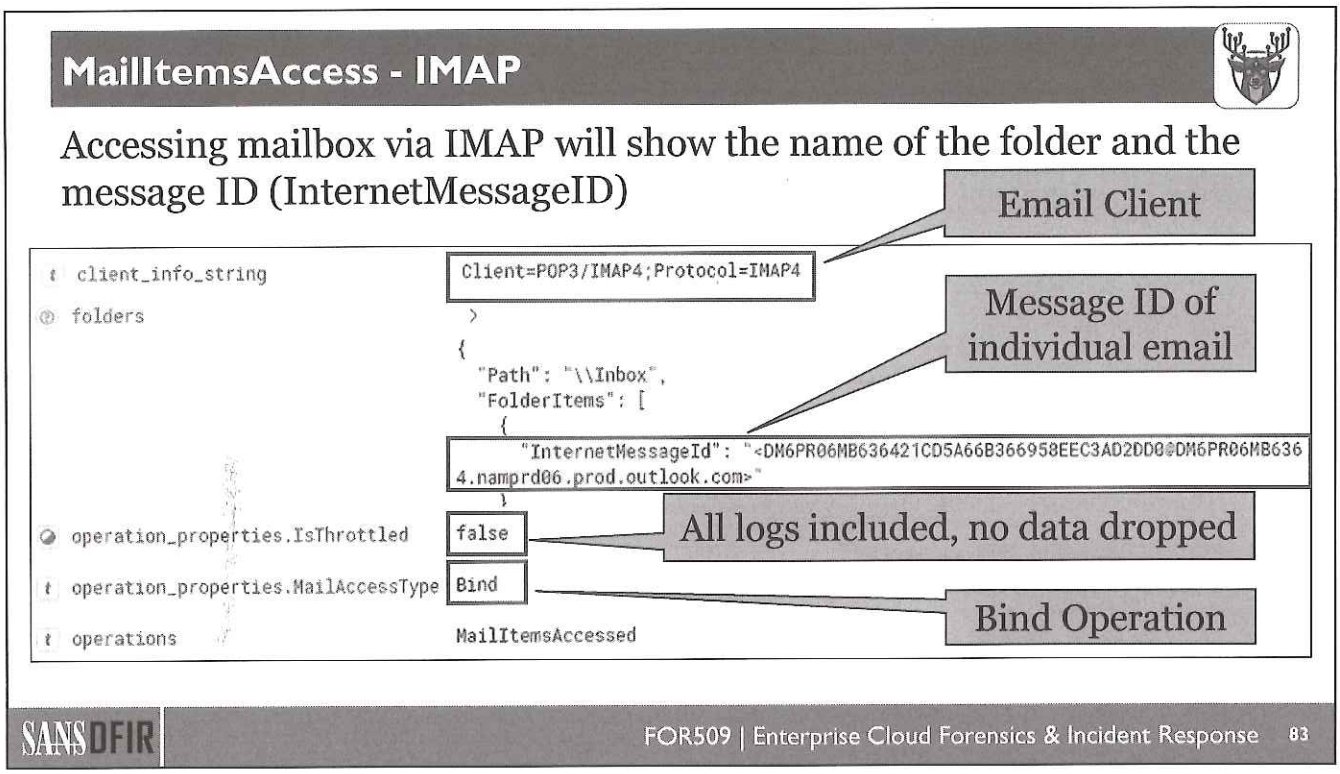
In this example, the `Inbox` folder is being synced between the Exchange server and the Outlook client.

As previously mentioned, since the Sync operations is at the folder level, the log entry only contains the name of the folder being synced and not the individual email message information.



This example shows the Thunderbird email client downloading a single email using the IMAP protocol.

IMAP (and POP3) clients will download individual emails using a Bind operation. This is an advantage as we can track individual emails by following the Message ID field through the various log entries.



This is an example of a "Bind" access via an IMAP/POP3 client as seen in SOF-ELK.

There is a lot of interesting information in this log entry:

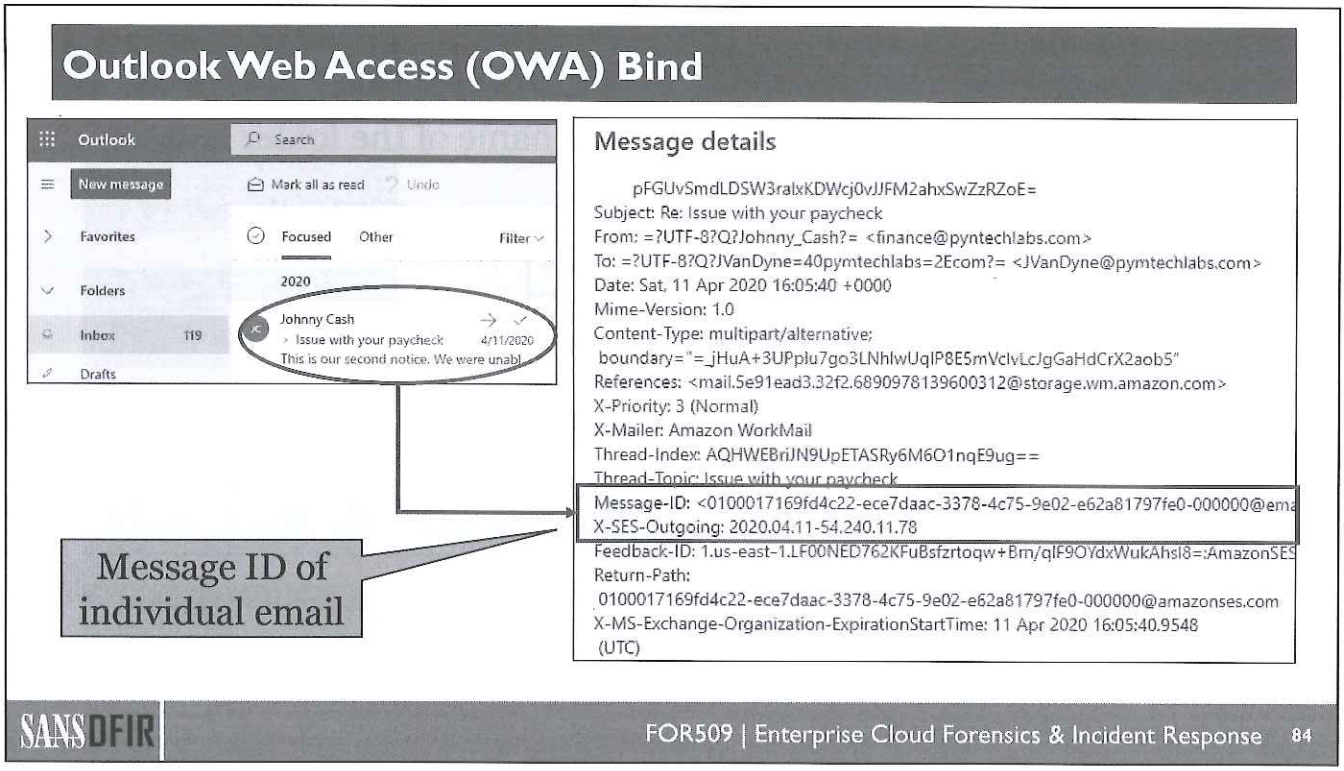
1. Client=POP3/IMAP4 – does the company authorize the use of POP3/IMAP? Many don't, so this kind of access may be very suspicious
2. The log entry shows not only the folder, but the exact email being accessed via the "InternetMessageId"
3. "IsThrottled" is set to False, so we know that all email accesses have been recorded

The MessageID header field remains constant as the message travels throughout the Exchange organization. This property is named InternetMessageId.[1]

Note: the actual string in the InternetMessageId shown in the slide is an example only.

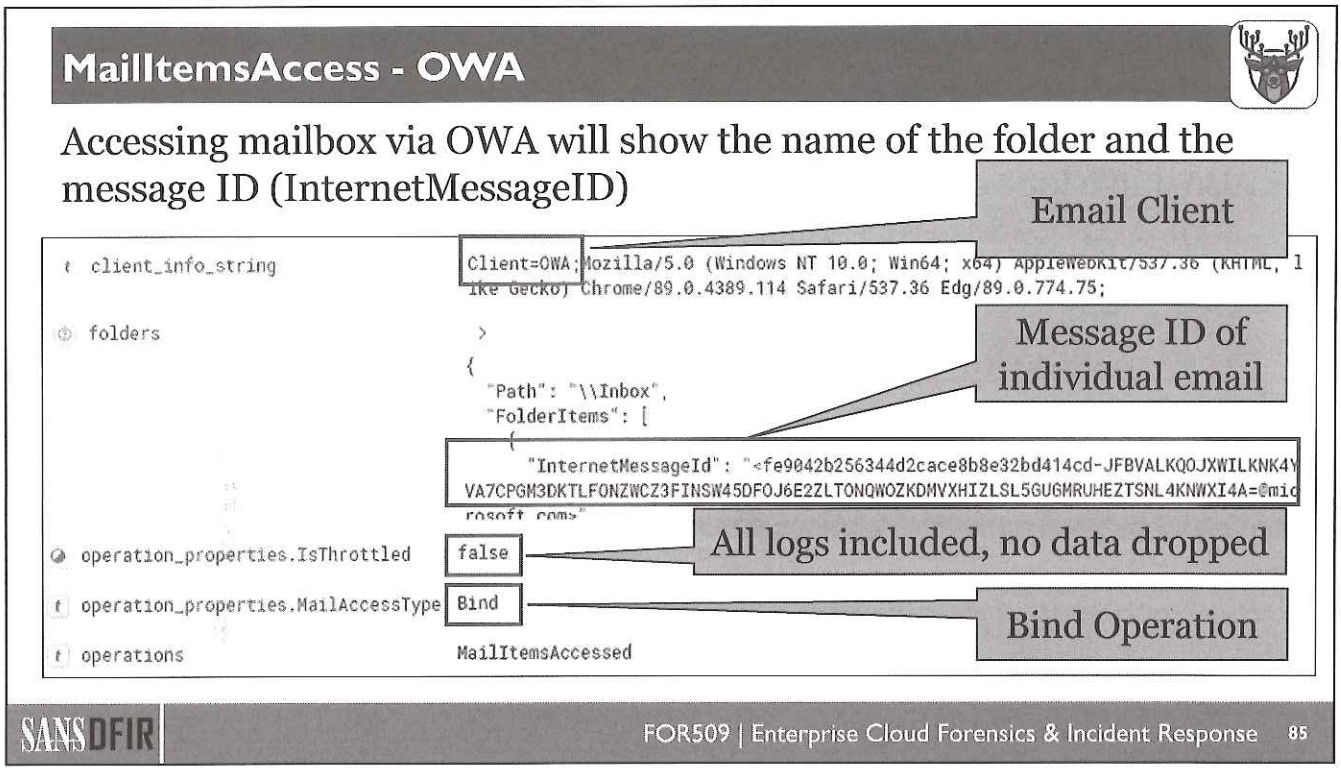
References:

[1] https://for509.com/internetmessageid



Many users will access their mailbox with Outlook Web Access (OWA). Like IMAP, OWA records a Bind operation for each email viewed by the user.

This is very advantageous for our investigations are the logs will contain a unique Message-ID for each email. As described in the previous slide, the property for this Message-ID is called InternetMessageId in the Microsoft 365 logs.



This is an example of a "Bind" access via an OWA (Outlook Web Access) as seen through the logs in SOF-ELK.

There is a lot of interesting information in this log entry:

1. Client=OWA – This is a common way to access email in most companies.
2. The log entry shows not only the folder, but the exact email being accessed via the "InternetMessageId".
3. "IsThrottled" is set to False, so we know that all email accesses have been recorded.

Note: the actual string in the InternetMessageId shown in the slide is an example only.

ForwardingSMTPAddress

- Auto-forward emails with `ForwardingSMTPAddress`
- Also watch for `DeliverToMailboxAndForward` or `ForwardingAddresss` rules
- Solution: block at domain level:

```
PS> Set-AutoForwardEnabled $false
AutoForwardEnabled : False
```

- Forwarded emails are dropped even if forward is set by the user at the mailbox level

MESSAGE EVENTS

DATE (UTC)	EVENT	DETAIL
7/16/2020 3:08:04 PM	Receive	Message received by: HE1PR0402MB3354
7/16/2020 3:08:04 PM	Receive	Message received by: HE1PR04MB3113 using TLS1.2 with AES256
7/16/2020 3:08:05 PM	Submit	The message was submitted.
7/16/2020 3:08:05 PM	Drop	Reason: [{LED=250 2.1.5 RESOLVER.MSGTYPE.AF; handled AutoForward addressed to external recipient};{MSG=};{FQDN=};{IP=};{LRT=}]
Reason: [{LED=250 2.1.5 RESOLVER.MSGTYPE.AF; handled AutoForward addressed to external recipient};{MSG=};{FQDN=};{IP=};{LRT=}]		
7/16/2020 3:08:05 PM	Spam Diagnostics	

Once a mailbox has been compromised, bad actors will frequently set a mail forwarding rule in order to get a copy of each email. This is a discreet way to obtain information that can be later used against the company. This is commonly used to commit financial fraud and is often referred to as Business Email Compromise (BEC).

The best solution is to disabled auto-forwarding at the domain level with the PowerShell command `Set-AutoForwardEnabled $false`

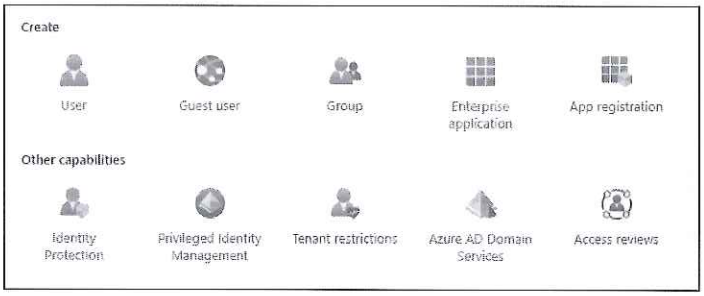
This will prevent emails from being forwarded even if a user sets a forwarding rule at the mailbox level.

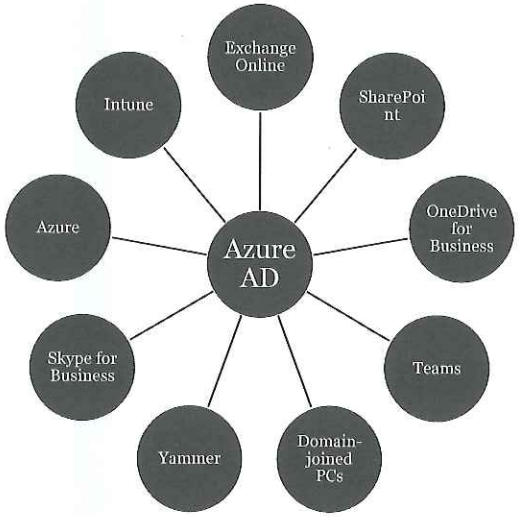
MESSAGE EVENTS

DATE (UTC)	EVENT	DETAIL
7/16/2020 3:08:04 PM	Receive	Message received by: HE1PR0402MB3354
7/16/2020 3:08:04 PM	Receive	Message received by: HE1PR04MB3113 using TLS1.2 with AES256
7/16/2020 3:08:05 PM	Submit	The message was submitted.
7/16/2020 3:08:05 PM	Drop	Reason: [{LED=250 2.1.5 RESOLVER.MSGTYPE.AF; handled AutoForward addressed to external recipient};{MSG=};{FQDN=};{IP=};{LRT=}]
Reason: [{LED=250 2.1.5 RESOLVER.MSGTYPE.AF; handled AutoForward addressed to external recipient};{MSG=};{FQDN=};{IP=};{LRT=}]		
7/16/2020 3:08:05 PM	Spam Diagnostics	

Azure Active Directory (Azure AD)

- Azure AD is Microsoft's Identity and Access Management (IAM) solution
- User agents can be cross referenced with Azure AD logs





Azure Active Directory is Microsoft's Identity and Access Management (IAM) solution. It contains a large number of features that a single slide can't do justice. Azure AD is used to manage authentication not only from Microsoft 365 applications, but also from Microsoft Azure. There is a trust relationship between the Azure subscription and Azure AD. Another key feature of Azure AD is multi factor authentication (MFA). Azure AD supports different MFA methods and in today's world there is no excuse for not turning on MFA.[1]

Azure AD is also used to authenticate user to domain-joined PCs.

For our purposes, we will look at Azure AD's log entries to understand who logged in.

References:

[1] https://for509.com/aad-intro

AzureAD Log Example

Example of a user login

```
@timestamp 2020-04-11 17:55:29.000 +00:00
workload   AzureActiveDirectory
operation  UserLoggedIn
client_ip  104.238.59.218
user_ids   dcross@pymtechlabs.com
useragent  Mozilla/5.0 (Windows NT 10.0;
Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/80.0.3987.163
Safari/537.36
```

- For regular login the "useragent" can be helpful in certain investigations, example: password spraying

Example of a system generated AD event

```
@timestamp  2020-04-11 17:40:37.000 +00:00
workload    AzureActiveDirectory
object_id   dcross@pymtechlabs.com
operation   Add member to group.
client_ip   40.126.6.52
modified_properties     {
  "Name": "AccountEnabled",
  "OldValue": "[]",
  "NewValue": "[\r\n  true\r\n]"},
result_status           Failure
user_ids    ServicePrincipal_87b15a38-add8-47ec-aaff-
0a98e8b42edb
```

- Notice that the IP address belongs to Microsoft
- The UserID is a service principal
- The attempt to add dcross to a group failed

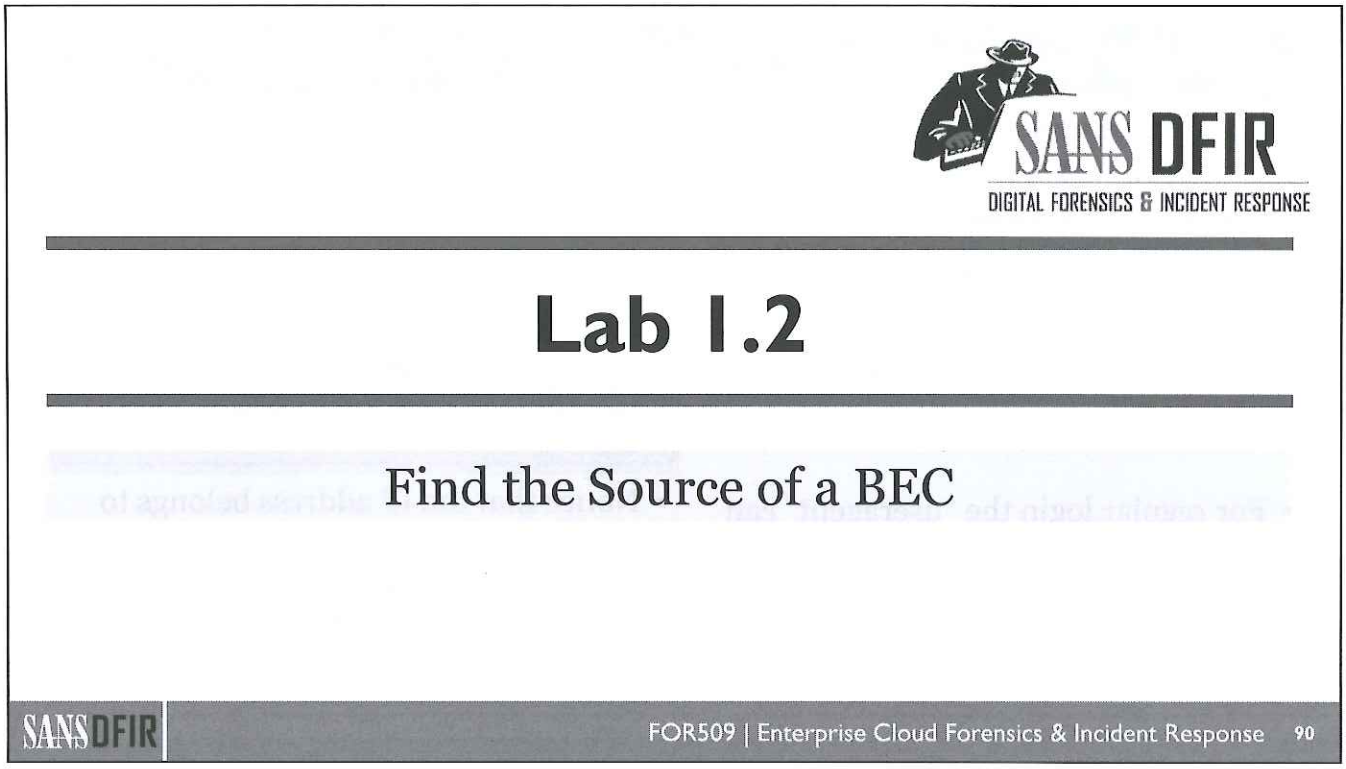
In a large environment Azure AD can be very verbose. Regular user login entries are straightforward. Others, not so much.

The example on the left shows a regular user login. As expected the workload is from AzureActiveDirectory since Azure AD performs the authentication. The other interesting fields are the `operation` and `user_ids` fields which indicate which user is authenticating. The `client_ip` and `useragent` fields may also be useful fields in your investigation.

Applications like SharePoint are constantly "doing things" which creates numerous log entries. They use service principals to perform these operations. Service principals are similar to service accounts.

The example on the right shows a system generated entry. The key fields are: `object ID`, `operation`, `modified_properties` and `result_status`. The `modified_properties` field will provide information about the change being made.

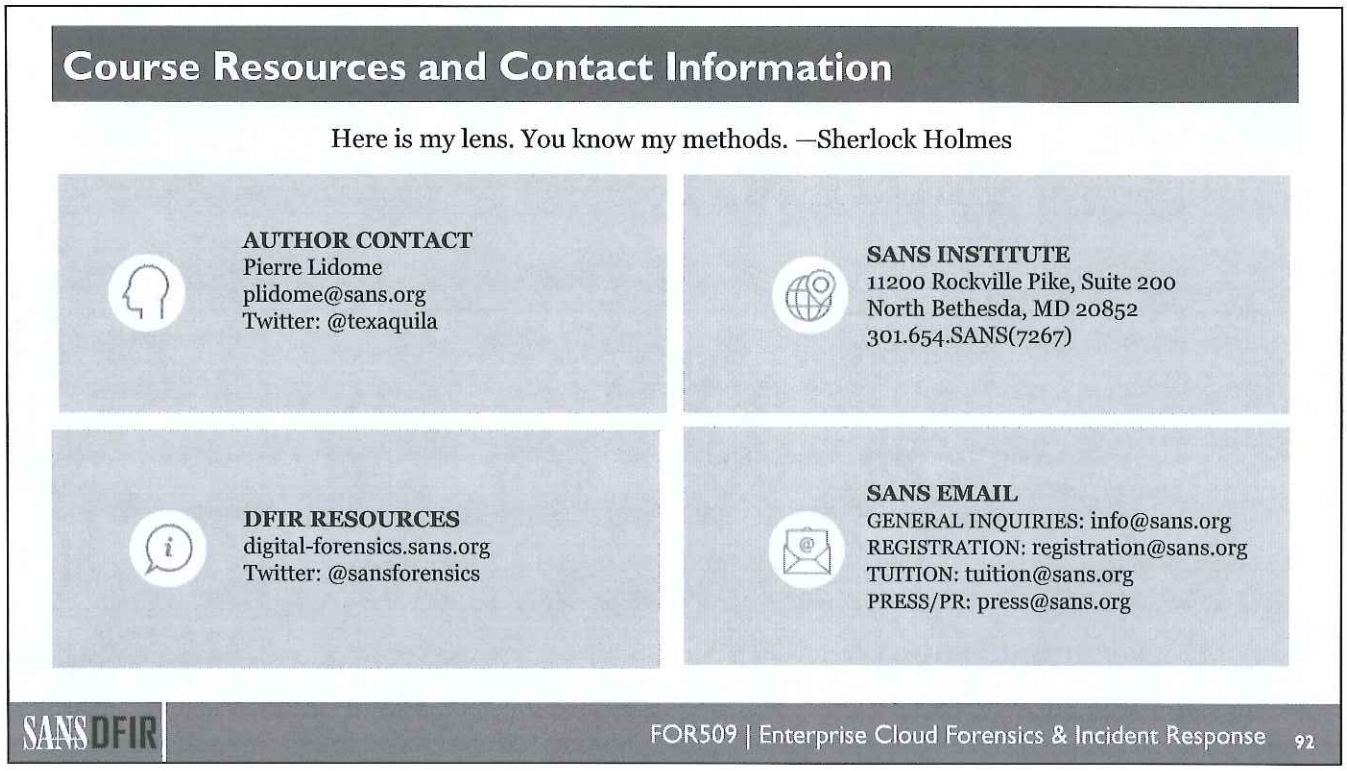
In this example, the account dcross@pymtechlabs.com was being added to a group. However, the operation failed. In this case, it failed because the account was already enabled (meaning it was already part of the group).



If you would like to read more about Business Email Compromise (BEC) investigations, we highly recommend reading PwC's Business Email Compromise Guide available on their GitHub:

`https://github.com/PwC-IR/Business-Email-Compromise-Guide`

This page intentionally left blank.



Author: Pierre Lidome

Email: plidome@sans.org

Twitter: @texaquila

"As usual, SANS courses pay for themselves by Day 2. By Day 3, you are itching to get back to the office to use what you've learned."
Ken Evans, Hewlett Packard Enterprise - Digital Investigation Services

SANS Programs
sans.org/programs

GIAC Certifications
Graduate Degree Programs
NetWars & CyberCity Ranges
Cyber Guardian
Security Awareness Training
CyberTalent Management
Group/Enterprise Purchase Arrangements
DoDD 8140
Community of Interest for NetSec
Cybersecurity Innovation Awards

Search SANSInstitute

SANS Free Resources
sans.org/security-resources

- E-Newsletters
  NewsBites: Bi-weekly digest of top news
  OUCH!: Monthly security awareness newsletter
  @RISK: Weekly summary of threats & mitigations
- Internet Storm Center
- CIS Critical Security Controls
- Blogs
- Security Posters
- Webcasts
- InfoSec Reading Room
- Top 25 Software Errors
- Security Policies
- Intrusion Detection FAQ
- Tip of the Day
- 20 Coolest Careers
- Security Glossary

SANS Institute
8120 Woodmont Avenue | Suite 310
Bethesda, MD 20814
301.654.SANS(7267)
info@sans.org