Optum CareData Platform Security

# Background

Many IT executive are still uncomfortable with the cloud adoption. Their main concern is around cloud security, which is causing them hesitation to move to the cloud. That concern may not be justified - studies have shown that cloud environments are more secure than their legacy counterparts. It may be due lack of understanding of how the cloud security works; where the division of responsibility lies between customer and cloud provider.

The main point to note is that the security in the cloud is a shared responsibility between customer and cloud provider. The diagram below shows the division of responsibility between the customer and Microsoft.



*Cloud shared responsibility model*

For all cloud deployment types, customer owns data and identities. Customer is responsible for protecting its data and identities, on-premises resources, and the cloud components customer controls (varies by service type). Regardless of the type of deployment, the following responsibilities are always retained by the customer:

* Data
* Endpoints
* Account
* Access management

The security is a big area with many dimensions. In this document however, the security is covered in context of Optum CareData platform (OCDP) with focus on Data protection, Identity and access management, Networking and Logging/Monitoring.

Optum CareData platform security is a broad set of policies, technologies, applications, and controls utilized to protect endpoints, data, applications, services, and the associated infrastructure. It is a sub-domain of computer security, network security, and information security. These measures ensure authentication of users and devices, access control for data and resources, and protection of data privacy. It is the foundation of building clients' confidence in the cloud computing; preserve reputation, avert financial losses and support data regulatory compliance. Some of the main benefits are: Protection against DDoS; Data security; and Regulatory compliance.

# Best practices

To ensure proper asset protection and appropriate access controls are enabled, time tested, industry best practices are developed. They are categorized into two main groups: Asset protection best practices and Access control best practices.

## Asset protection best practices

* Network protection
* Identity management
* Posture and Vulnerability Management
* Enable threat detection for IAM
* Incident response
* Automation
* Data protection
* Logging and threat detection

### Network protection

* All services are created within virtual networks (vnets) and subnets.
* Each subnet is protected by a network security group (NSG).
* No anonymous public access is allowed. Instead, private endpoints are used.
* Azure VNETs should follow an enterprise segmentation principle and align to the business risks. Higher risks for the organization should be isolated within their own virtual networks and sufficiently secured with either a network security group (NSG) and/or Azure Firewall.
* Network access is limited to specific networks.
* Restrict/allow traffic between internal resources based on network security group rules. Adopt "deny by default, permit by exception" approach. Use Azure Firewall where central management is required.
* Use Adaptive Network Hardening to recommend network security group configurations that limit ports and source IPs.
* Use Azure Sentinel to discover the use of legacy insecure protocols such as SSL/TLSv1, SMBv1, LM/NTLMv1, wDigest, Unsigned LDAP Binds, and weak ciphers in Kerberos.

### Identity management

* Use strong authentication controls for all Azure Active Directory based access
  + Azure AD supports strong authentication controls through multi-factor authentication (MFA) and strong passwordless methods.
    - Enable Azure AD MFA and follow Azure Security Center identity and access management recommendations for MFA setup.
    - Passwordless authentication options are: Windows Hello for Business, Microsoft Authenticator app, and on-premises authentication methods such as smart cards.
    - For administrator and privileged users, use the highest level of the strong authentication method possible.
* For third-party applications and marketplace services, change default password during initial service setup.

### Posture and Vulnerability Management

* Preserve secure configurations for Azure services
  + Use Azure Security Center to monitor the configuration baseline and use Azure Policy [deny] and [deploy if not exist] rule to enforce secure configuration across Azure compute resources, e.g., VMs, containers and so on.
* Persist secure configurations for compute resources
  + Use Azure Security Center and Azure Policy to regularly assess and remediate configuration risks on Azure compute resources. Use Azure Resource Manager templates, custom operating system images, or Azure Automation State Configuration to maintain the security configuration of the operating systems.
* Quickly and automatically remediate software vulnerabilities
  + Deploy software updates quickly to remediate software vulnerabilities in operating systems and applications.
  + Use a risk scoring program or risk ratings provided by a third-party scanning tool and tailor to OCDP environment.
  + Use Azure Automation Update Management or a third-party solution to ensure that the most recent security updates are installed.
* Backup & recovery
  + Mitigate risk of lost resource keys - Ensure you have measures in place to prevent and recover from loss of keys. Enable soft delete and purge protection in Azure Key Vault to protect keys against accidental or malicious deletion.

### Enable threat detection for IAM

* Azure AD provides the following user logs that can be viewed in Azure AD reporting or integrated with Azure Monitor, Azure Sentinel or other monitoring tools:
  + Sign-ins - The sign-ins report provides information about the usage of managed applications and user sign-in activities.
  + Audit logs - Provides traceability through logs for all changes done by various features within Azure AD.
  + Risky sign-ins - A risky sign-in is an indicator for a sign-in attempt that might have been performed by someone who is not the legitimate owner of a user account.
  + Users flagged for risk - A risky user is an indicator for a user account that might have been compromised.
* Enable logging for Azure resources
  + Enable logging for Azure resources to meet the requirements for compliance, threat detection, hunting, and incident investigation.
  + Use Azure Security Center and Azure Policy to enable resource logs and log data collecting on Azure resources for access to audit, security, and resource logs. Activity logs, which are automatically available as well.

### Incident response

* Detection and analysis - create incidents based on high quality alerts
  + Create high quality alerts and measure the quality of alerts to learn from the past.
  + High quality alerts can be built based on past incidents, validated community sources, and tools to generate and refine alerts by combining diverse signal sources.
  + Azure Security Center provides high quality alerts across many Azure assets. You can use the ASC data connector to stream the alerts to Azure Sentinel or any other log aggregation tool of choice.
  + Export Azure Security Center alerts and recommendations using the export feature to identify risks to Azure resources.
* Detection and analysis - prioritize incidents
  + Determine which incidents to focus on first based on alert severity and asset sensitivity.
  + Azure Security Center assigns a severity to each alert to help you prioritize which alerts should be investigated first.
  + Use resource tags and create a naming system to identify and categorize Azure resources, especially those processing sensitive data.

### Automation

* Automation scripting is highly encouraged to promote predictability and repeatability.

### Data protection

* Only secure data transfer (HTTPS) is permitted; data is encrypted while data is in-flight and at-rest; data protection policies are used.
* To complement access controls, data at rest should be protected against ‘out of band’ attacks, e.g., accessing underlying storage using encryption.
* Azure provides encryption for data at rest by default. You can implement additional encryption as well.
* Azure Defender for Azure Storage is used to provide an additional layer of security intelligence to detect unusual and potentially harmful activities or exploits.
* Shared access signature (SAS) tokens are limited to HTTPS connections only.

### Logging and threat detection

* Enable threat detection for Azure resources
  + Monitor different types of Azure assets for potential threats and anomalies. Alerts can be sourced from log data, agents, or other data. Focus on high quality alerts.
  + Use the Azure Security Center built-in threat detection capability, which is based on monitoring Azure service telemetry and analyzing service logs.
  + Use Azure Sentinel to build analytics rules, which hunt threats that match specific criteria across the environment.

## Access control best practices

* Access control
* Authentication and authorization
* Audit control

### Access control

Modern access control systems leverage unified access management, which offers access controls for on-premises and cloud environments. Appropriate access controls are of utmost importance for any successful cloud adoption.

* Azure Active Directory is leveraged to authorize resource access.
* Principal of least privilege is exercised.
* Security certificates are used for mutual authentication.
* For service-to-service authentication, System-assigned Managed Identity, user-assigned Managed Identity or Service Principal is used.
* The industry best practice for access control is to use federated authentication where internal (integrated) and external (non-integrated) users are interacting with a platform/system. It is a way of a user's single authentication ticket/token, is trusted across multiple IT systems and organizations. It enables seamless user experience, averts vulnerability risks in a timely and cost-effective manner.
* Establish a trust between organizations. User authentication occurs at the user's own organization, alleviating the burden of user management in platform/system's organization.

### Authentication and authorization

Access control is a security technique that regulates who or what can view or use resources in a computing environment. It is a fundamental concept in security that minimizes risk to the business or organization. It limits access on need to know basis to computer networks, system files and data.

The goal of access control is to minimize the security risk of unauthorized access to physical and logical systems. It ensures that security technology and access control policies are in place to protect confidential information, such as patient healthcare data. The following are some of the recommended best practices:

* Create infrastructure and procedures that limit access to networks, computer systems, applications, files and sensitive data, such as personally identifiable information (PII) and intellectual property.
* Perform identification authentication and authorization of users and entities by evaluating required login credentials that can include passwords, personal identification numbers (PINs), biometric scans, security tokens or other authentication factors.
* Use Multifactor authentication (MFA), which requires two or more authentication factors to provide a layered defense to protect access control systems.
* Minimize use of user id and password. Instead, use user security certificates. The user id and password are used only for client facing parts of a system.
* For service-to-service communication and authentication use system-assigned managed identity, user-assigned managed identity or service principal.

### Audit control

* User data access audits (trust but verify) are implemented.
* The way requests are authorized is tracked.
* Alert notifications are set up for enabling faster response to adverse events.
* Azure recommended security best practices are used; vulnerability scans are conducted.
* Just-In-Time access to appropriate resources when needed and for the least amount of time required is exercised.

# OCDP current state

* Services and components are being developed, configured and deployed on on-going basis.
* The security policies are defined and deployed.
* The security concerns are treated as first citizens.
* Automation is in full swing.
* However, still quite a few components, services and storage accounts that are only partially compliant with defined security policies.
* The security setup work is in-progress, but not finished yet.
* Azure Data Factory v2.0 has been chosen to run data pipelines, process all types of data ingestion use cases, leverage native integrations into Azure services such as, Azure Active Directory, managed identities and so on.

## Gaps

Despite best intentions and efforts, there are still security gaps that need to be filled. The list below is a partial list of gaps between the current state and the target state:

* Network security
  + All subnets do not have network security groups (NSG) assigned to them
  + Public access to All networks is enabled
  + Private endpoints are not used
  + Network security groups are too generic
* Identity Management
  + Still no viable federated authentication solution
  + Solutions being devised are subpar and keep getting rejected by the EIS group
  + CDO user identities are managed as local users in OCDP Azure Active Directory
* Data protection
  + Replication Data is set as Local Redundant Storage (LRS)
  + Data backups are stored for 8 hours only
  + Geo-redundant backup storage is disabled
* Logging and threat detection
  + Alerts not enabled for faster response to adverse events
  + Logs are not collected from all services
  + Incident response is delayed
* Posture and Vulnerability Management is not defined
* Backup & recovery process of resource keys are not in place
* DDoS attack prevention is not configured
* Resource Locks are not setup to prevent inadvertent changes and/or deletions
* AKS cluster security recommendations are not followed

## Mitigative steps and strategy

To ensure proper data and asset protection, Azure Security Benchmark (v1) is closely followed. It is important to identify entry points into the Optum CareData Platform (OCDP); identify who will access the platform, which services and for what purpose.

### Entry points into OCDP

* Kafka brokers/topics
* On-premise data upload (Kafka Connect)
* Azure Data Factory (ADF v2.0)
* CDO datacenter (Kafka brokers/topics, schema registry)
* Snowflake data platform
* FHIR server
* Tableau server
* CMAK, Grafana
* Browser based service management

User profiles interacting with the Optum CareData platform

* Operational teams
  + CMAK, Grafana, monitoring, deployment, configuration management and log management
* Business integrated (internal) users will access the following services:
  + Snowflake, FHIR server, Tableau server and ADF v20.0
* External non-integrated (external) users will access the following services:
  + Snowflake, FHIR server, Tableau server and ADF v20.0
* Managers and Executives:
  + Snowflake, FHIR server, Tableau server and Dashboards
* Data Scientists will access the following services:
  + Machine learning, data engineers, AI enablement, Snowflake, and possibly even FHIR server

### Access control

* User authentication and authorization are done using Azure Active Directory.
* The Role-based access controls (RBAC) are defined at the Azure Active Directory (AAD) level and the fine grain access controls are implemented at the Snowflake and FHIR permissions level.

### Data security

Data transportation is secured using TLS v.1.2 security and authentication is based on mutual security certificates. Data at-rest is secured using Snowflake and FHIR provided encryption capabilities. The Risk level 1 fields are also tokenized before storing.

### Services in the Optum CareData platform that deserve close attention:

* Storage accounts
* Azure Kubernetes Service
* Azure Data Factory
* Grafana and Domo
* Informatica
* Downstream data consumption
* Rest API
* Browser based administration
* Threat vulnerability scanning, detection and alerting

#### Storage accounts

The following set of best practices are used to ensure storage accounts are properly created, protected and accessed:

* Soft delete for blobs and containers is enabled to prevent accidental deletion
* Locks are setup for storage account to prevent accidental changes/deletions
* Business-critical data is stored in immutable blobs
* Only secure transfer (HTTPS) to the storage accounts is allowed
* Azure Defender for Azure Storage is used to provides an additional layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit storage accounts
* Shared access signature (SAS) tokens are limited to HTTPS connections only
* The principal of least privilege is exercised
* Account access keys are secured with Azure Key Vault
* Anonymous public read access to containers and blobs is disabled
* Public access to All networks is disabled
* Shared Key authorization is not allowed
* For service-to-storage account authentication System-assigned Managed Identity, user-assigned Managed Identity or Service Principal is used
* Only TLS ver. 1.2 or higher is allowed for a storage account
* Firewall rules are used to limit access to the storage account to requests that originate from specified IP addresses or ranges, or from a list of subnets in a Azure Virtual Network (VNet).
* Trusted Microsoft services can access the storage account
* Use of private endpoints is preferred
* Azure Storage logging is enabled to track how each request made
* Log alerts are enabled for faster response to negative events

|  |
| --- |
| High-level overview of the features of Azure Defender for Storage |

*Optum CareData storage attack mitigation framework*

* Data replication ZRS or RA-GRS option is used
* Geo-redundant backup storage enabled
* Data backups are stored for 7 days
* Storage encryption should use Microsoft-managed keys
* All subnets have network security groups (NSG) assigned to them
* Private Endpoint Connections are used
* Firewall is used along with Azure Front Door
* Network security groups are purpose specific
* Prevent volumetric DDoS attacks is configured
* Create a playbook and use it If/when attacked to stop it
* Resource Locks are created to prevent changes and/or inadvertent deletion
* Azure Advisor recommendations are heeded

#### Azure Kubernetes Service

Security Center continuously monitors the configuration of kubernetes services to identify

potential security vulnerabilities and recommends actions to mitigate them. To be aware of and to be notified of anomalous events, Azure Defender for Kubernetes should be turned on. The following are the Azure Kubernetes Service vulnerability mitigation recommendations:

|  |  |
| --- | --- |
| **Description** | **Severity** |
| Kubernetes clusters should be accessible only over HTTPS | High |
| Running containers as root user should be avoided | High |
| Container images should be deployed from trusted registries only | High |
| Overriding or disabling of containers AppArmor profile should be restricted | High |
| Usage of pod HostPath volume mounts should be restricted to a known list to restrict node access from compromised containers | Medium |
| Services should listen on allowed ports only | Medium |
| Containers should listen on allowed ports only | Medium |
| Container with privilege escalation should be avoided | Medium |
| Container CPU and memory limits should be enforced | Medium |
| Immutable (read-only) root filesystem should be enforced for containers | Medium |

#### Azure Data Factory

* Azure Data Factory management resources are built on Azure security infrastructure and use all possible security measures offered by Azure to protect data movement.
* Azure Data Factory (ADF v2.0) including Azure Integration Runtime and Self-hosted Integration Runtime does not store any temporary data, cache data or logs except for linked service credentials for cloud data stores, which are encrypted using certificates.
* ADF has native integration with Trusted Microsoft Services’ such as, Azure Active Directory, Azure Key Vault, managed identities, storage accounts, Managed Virtual Network (VNET) and private endpoints. Credentials or secret values are stored in an Azure Key Vault and use them during pipeline execution to pass to the activities.
* Create and use managed identity for ADF and use it to interact with the Azure Key Vault.
* It is strongly suggested to create custom roles such as, Data Factory Pipeline Executor, Data Factory Reader and so on, because the security model and available roles for Data Factory are very basic.

##### Azure Data Factory v2.0 built-in security capabilities

* **Secure data store credentials -** Azure Data Factory protects data store credentials by encrypting them by using certificates managed by Microsoft. The encrypted credentials are securely stored in an Azure Storage managed by Azure Data Factory management services.
* **Data encryption in-transit -** If the cloud data store supports HTTPS or TLS, all data transfers between data movement services in Data Factory and a cloud data store are done via secure channel HTTPS or TLS.
* **Data encryption at-rest -** If data stores support encryption of data at rest. It is strongly recommended that data encryption mechanism be enabled for those data stores.
* **Azure Synapse Analytics -** Transparent Data Encryption (TDE) in Azure Synapse Analytics helps with protecting against the threat of malicious activity by performing real-time encryption and decryption of your data at rest. This behavior is transparent to the client.
* **Azure Data Lake Store -** Azure Data Lake store also provides encryption for data stored in the account. When enabled, Data Lake store automatically encrypts data before persisting and decrypts before retrieval, making it transparent to the client accessing the data.
* **Azure Blob Storage and Azure Table Storage -** Azure Blob Storage and Azure Table storage supports Storage Service Encryption (SSE), which automatically encrypts your data before persisting to storage and decrypts before retrieval.

#### Grafana and Domo

#### Grafana

Grafana is a multi-platform open-source analytics and interactive visualization web application. It provides charts, graphs, and alerts for the web when connected to supported data sources.

* Grafana is secured with an SSL certificate and an Application Gateway Ingress Controller proxy. It supports many ways to authenticate users such as, OAuth 2.0, SAML, LDAP. But at OCDP we are using OAuthv2.0.
* it is highly recommended to change the login information as soon as possible because, every Grafana installation uses the same administrative credentials by default. So, for default we have a random password for admin which changes at every deployment and disabled the admin login to prevent it from unwanted login or user creation.
* Non-Grafana web services running on Grafana server might be vulnerable to exploitations through the Grafana data source proxy or other methods. To prevent such exploitations, the following are the recommended best practices:
  + Limit IP addresses/hostnames for data source URL -Configure Grafana to only allow certain IP addresses or hostnames to be used as data source URLs and proxied through the Grafana data source proxy. For our use case we have only allowed optumIP to access Grafana.
  + Firewall rules -Configure a firewall to restrict Grafana from making network requests to sensitive internal web services. Application Gateway is configured with firewall rules which can be made stricter depending on requirement.
  + Proxy server -Require all network requests being made by Grafana to go through a proxy server. All requests go through the Application gateway.
  + Limit Viewer query permissions: Users with the Viewer role can enter any possible query in any of the data sources available in the organization. Restrict data source query access as shown below:
* Create multiple data sources with some restrictions added in data source configuration that restrict access (such as, database name or credentials). Then use the Data Source Permissions Enterprise feature to restrict user access to the data source in Grafana.
* Create a separate Grafana organization, and in that organization, create a separate data source. Make sure the data source has some option/user/credentials setting that limits access to a subset of the data. Be aware, not all data sources have an option to limit access.
* At OCDP, we have used Azure Active Directory to create three different Roles as Admin, Creator and Users.
* Disable anonymous access to dashboards -Anonymous access to a dashboard is publicly available. The list below shows the security implications of enabling Anonymous access:
* Anyone with the URL can access the dashboard
* Anyone can make view calls to the API and list all folders, dashboards, and data sources
* Anyone can query any data source that the Grafana instance is configured with.
* Anonymous login is disabled at OCDP and no one without the Azure Active Directory Roles attributes can view the dashboard.

#### Domo

Domo is used for BI & Analytics to visualize and analyze data. Intelligent Apps to build apps for new data experiences. Embedded & Extended Analytics to share data outside the organization. It can connect data with hundreds of connectors, ability to access information and report from single platform, customized solutions.

Domo is the **preferred tool for dashboarding** at Optum CareData platform.

Domo is in the same space as Tableau. However, Tableau is a Data Visualization champion with a self-service data discovery feature.; whereas Domo is a cloud-based dashboard that provides a variety of visualization at a lower price point.

* The Domo Trust program is created to live up to its security commitment. Compliance and privacy controls that are in place to protect customers’ most sensitive data and is available upon request.
* Domo enables our customers to always be in control. Multiple logical and physical security layers. Least privilege and separation of duties access model. Threat assessments of each new feature, Transport layer encryption and encryption at rest, that allows customers to manage their own encryption keys and extensive logging and monitoring of network, system and application events;
* Domo provides extensive self-service security features that enable customers to stay in control and have full transparency into their data. These include SAML-based SSO, Multi-Factor Authentication, managed encryption keys, IP address restrictions, security profiles, and more.

#### Informatica

Informatica Intelligent Data Management cloud has some of the compelling capabilities such as, data catalog, data integration, data prep, data quality, master data management, customer and business 360 and governance & privacy

Secure the Informatica domain to protect from threats from inside and outside the network. It includes the following types of security:

* **Infrastructure Security** – it protects the Informatica domain against unauthorized access to or modification of services and resources in the Informatica domain.
* **Operational security** - controls access to the data and services in the Informatica domain. Operational security includes the following aspects:
* Setting restrictions to user access to data and metadata based on the role of the user in the organization
* Setting restrictions to user ability to perform operations within the Informatica domain based on the user role in the organization
* Informatica stores the domain configuration information and the list of authorized users. It also contains the groups, roles, privileges, and permissions that are assigned to each user in the Informatica domain.
* Informatica organizes the list of users by security domains.

##### Informatica shortcomings

* It requires a remote desktop which is not preferred at OCDP
* It does not do version control very well
* TCO is way too high in comparison to other vendors' offers
* ETL module is antiquated
* Data Quality module is also not recommended at OCDP
* No web-based client, causing access issues
* The Web Services Consumer transformation and the REST Web Services Consumer transformation do not support the Timestamp with Time Zone data type.
* Performance with ODBC drivers is comparatively slow which ends up in taking a lot of time
* When you try to delete a parameter set that is part of an application or workflow, the Developer tool generates a null pointer exception and does not delete the parameter set.
* A mapping fails when the following circumstances are true:
* The mapping is designed for a flat file source and parameterize the source to be a relational source.
* The mapping is deployed with a parameter set.
* The mapping has a resource parameter and either a port list parameter or a sort list parameter.
* When you run multiple concurrent mappings from infacmd command line for a long time, the mapping run might fail with an error.
* When Data Transformation cannot process JSON or XML input files, the profile run fails.
* **Developer perspective -** it has the following shortcomings:
* It is not the most developer friendly tool you've ever came across. For example, Sorting of dates and numbers are based on string values.
* Lack of proper sorting in Workflow Monitor, e.g., you can't order sessions by Name, Start Time, Completion Time. Reading execution logs is a challenge, especially when a workflow has many sessions.
* In Workflow monitor, there's no option to filter out irrelevant folders.
* Web-based workflow execution monitoring tool is not available

**Note*:*** *Informatica is* ***discouraged on PADU*** *and is**discouraged at OCDP*

*Azure Data Factory (ADF v2.0) is the recommended alternative.*

#### Downstream data consumption

The downstream data consumption refers to users, machines, processes and services getting data from the Optum CareData Platform to consume it for their use cases. Many different user profile types could be consuming data. For example, reports and dashboards services, downstream data jobs, individual users could be using for analytical processes/jobs, data scientists could be using it to make predictions and recommendations, and so on.

To provide proper guardrails around the data and the platform, users will only be allowed to access Azure services provisioned for them. It is accomplished by identifying who will access the Optum CareData platform, which services, for what purpose, which format and what frequency. OCDP users will access the following platforms and services based on their roles and responsibilities:

* Snowflake CareData platform
* FHIR server
* Tableau server
* Azure Data Factory (ADF) v2.0
* CMAK
* Grafana
* Browser based service management

The following user profiles will be interacting with the Optum CareData platform:

* Operational teams
  + CMAK, Grafana, monitoring, deployment, configuration management and log management
* Business integrated (internal) users will access the following services:
  + Snowflake, FHIR server, Tableau server and ADF v20.0
* External non-integrated (external) users will access the following services:
  + Snowflake, FHIR server, Tableau server and ADF v20.0
* Managers and Executives:
  + Snowflake, FHIR server and Tableau server
* Data Scientists will access the following services:
  + Machine learning, data engineers, AI enablement, Snowflake, and possibly even FHIR server

The Role-based access controls (RBAC) are defined at the Azure Active Directory (AAD) level and the fine-grained access controls are implemented at the Snowflake and FHIR permissions level.

The approach is to abstract implementation details from the downstream data consumers. The standardized data will be served via standardized protocols, e.g., Rest based API calls, data services, Tableau reports and dashboards.

**Note: *Wild card searches are not supported*** *at this time.*

##### Best practices for downstream data consumption

* **Data preparation** – Pertains to the synchronization of demand data.  To synchronize and use the data, it must be cleaned, harmonized, enriched and standardized based upon a carefully crafted data architecture.
* **Design with the end in mind** - Every organization has many use cases**.** Each use case requires a different frequency of data e.g., daily, weekly, etc. and data enrichment schema.  As a result, data must be held at the lowest level of granularity.
* **Change is hard** - In order to make real progress, an organization must embrace new concepts and think about the art of the possible.  The following change management issues dealt upfront could ease that pain:
  + Separate data ingestion from data consumption
  + Be prepared to serve raw data
  + Be open to change. You can innovate/create new products and services only if you give yourself the chance to change your focus from inside-out processes to outside-in processes.
  + Reward inspiration, innovation and perspiration. They need leadership support, encouragement and financial backing.
  + Think like a big organization and operate like a start-up. It is a different focus and requires a different mindset.
* **Build the right governance structure** - It should be a cross functional team. Determine who would own the data, who would access the data, for what purpose, in which format and at what frequency.
* **Determine the right technology for the right job** - For example, a distributed data visibility component to disseminate information across the organization; alerting solutions to enable stakeholders to react to events quickly and effectively; and automated applications can provide you the data at the aggregation level and in the shape consumer demands.

#### Rest API

##### A primary advantage of REST over HTTP is that it uses open standards and does not bind the implementation of the API or the client applications to any specific language. A REST web service could be written in ASP.NET, and client applications can use any language of choice.

##### Design Principals of RESTful APIs using HTTP

* REST APIs are designed around resources, which are any kind of object, data, or service that can be accessed by the client.
* A resource has an identifier*,* which is a URI that uniquely identifies that resource, e.g., the URI for a customer order might be: https://adventure-works.com/orders/1
* Clients interact with a service by exchanging representations of resources. Many web APIs use JSON as the exchange format, e.g., a GET request to the URI listed above might return this response body: {"orderId":1,"orderValue":99.90,"productId":1,"quantity":1}
* REST APIs use a uniform interface, which helps to decouple the client and service implementations. The most common operations verbs are GET, POST, PUT, PATCH, and DELETE:
  + **GET** retrieves a representation of the resource at the specified URI
  + **POST** creates a new resource at the specified URI
  + **PUT** either creates or replaces the resource at the specified URI
  + **PATCH** performs a partial update of a resource
  + **DELETE** removes the resource at the specified URI
* REST APIs use a stateless request model. HTTP requests should be independent and may occur in any order, so keeping transient state information between requests is not feasible. Each request should be an atomic operation to be highly scalable.
* REST APIs are driven by hypermedia links that are contained in the representation.
* Avoid dependencies between the web API and the underlying data sources. For example, if your data is stored in a relational database, the web API doesn't need to expose each table as a collection of resources.
* When you can’t map every operation implemented by a web API to a specific resource. You can handle suchnon-resource scenarios through HTTP requests that invoke a function and return the results as an HTTP response message.

##### A well-designed web API should aim to support

* **Platform independence** - Any client should be able to call the API, regardless of how the API is implemented internally.
* **Service evolution** - The web API should be able to evolve and add functionality independently from client applications. Existing client applications should continue to function without modification.
* Organize the API around resources
* Focus on the business entities that the web API exposes
* A resource doesn't have to be based on a single physical data item
* Group Entities into collections (patients), e.g., <https://examplehealthnet.com/patients>
* The API must guarantee that it is always available to respond to calls and that once it begins execution on the call, that it can finish handling the received message immediately without losing data.

##### Security best practices

* A secure API must guarantee the confidentiality of the information it processes by making it visible only to the Users, Apps, and Servers that are authorized to consume it.
* It must be able to guarantee the integrity of the information it receives from Clients and Servers that it collaborates with, so that it will only process information, if it knows that it has not been modified by a third-party.
* You must be able to recognize the apps that consume your API, users of the same API, and the Servers that your API calls. The API should be able to identify itself to both apps and servers.

###### Identity

* You need an Identity Store that you can refer to verify your recognition of Apps and Users.
* An Identity Provider is dedicated to managing the interaction with the Identity Store(s) for authentication and authorization purposes.
* Identity is the core of the security.

###### User and app authentication

When an App Id or a User’s username (claim) in a call is presented to the API, the API must be able to verify the authenticity of the claim. This is done with some form of a shared secret.

Username/password is a common practice to present as credentials. But it is not a preferred method due to their predictability and password maintenance burdens. Maximum randomness is preferred. Instead, Multi-factor authentication (MFA) and Token-based credentials are desired. Tokens are issued with an expiration period and can be revoked. Since they are issued uniquely to each App, when you choose to revoke a token or if it expires, all the other Apps can continue to use their tokens independently.

###### API and server authentication

The API must be able to authenticate itself to the Apps which consume it. Likewise, when your API interacts with Servers, they must authenticate themselves to the API. Avoid man-in-the-middle attacks, which may pretend to be a Server or your API.

###### User and app authorization

Role-based access control (RBAC) - represents a simple access control mechanism. An App need not keep a record of each User’s level of access to its functions and data. Rather it uses roles to abstract away from those details and assign degrees of access to groups of Users which the role represents.

Attribute-based access control (ABAC)

It enables dynamic determination of access control based on some sort of circumstantial information available at the time of the API call, e.g., the time of day, the role, the location of the API, the location of the App, and combinations of conditions to determine the level of access. Assumption is that these conditions may change from call to call. XACML is a standard which defines the rules that must be executed to evaluate the level of access at the time of the API call.

Delegated access control with OAuth 2.0

The HTTP-based OAuth 2.0 framework allows an App to obtain access to a resource exposed by the API either on its own behalf or on behalf of someone elsewho owns the resource. Thus, it allows Usersto delegate access control to third-party Apps. To facilitate this, your API must collaborate with an OAuth 2.0 authorization Server, checking each incoming call for an access token which it must validate with the authorization Server.

###### Federated identity

Federated Identity Providers collaborate to facilitate the authentication and authorization of users who

belong to different security contexts. For example, user A signs into her App. The App authenticates her

and requests a token from the service B Identity Provider. This authenticates her with the service B Identity Store and then responds to the App with a token.

Single Sign-on multi-experience

The Security Assertion Markup Language (SAML) is an industry standard that has become a defacto standard for Enterprise level Identity Federation. A SAML Assertion can be issued by an Identity Provider in one security context and be inherently understandable by an Identity Provider in another context in a standardized way.

###### Confidentiality, integrity, and availability

Message Integrity goes beyond the authentication of the app and the user; it includes the verification that the message was not compromised mid-flight by a malicious actor. Digital signatures and message safety further ensure the message integrity.

###### Message confidentiality

The integrity of a message sent by a known App is assuring, but it is also necessary to hide message details from the point of delivery by the App to the reception by the Server. An agreement is needed between the App and API to be able to hide the details of the message in a way that allows only the API to uncover them and vice versa. You can preserve the confidentiality further by using techniques such as, Public key cryptography, Digital Certificates, Mutual authentication with Digital Certificates, and HTTPS.

#### Threat vulnerability scanning, detection and alerting

### Threat Vulnerability scanners are automated tools that allow organizations to check if their networks, systems and applications have security weaknesses that could expose them to attacks. It is necessary to understand the different types of vulnerability scanning and how it works with pen testing. There are many tools and products in the vulnerability scanning space that cover different types of assets and offer additional features that help companies implement a complete vulnerability management program - the combined processes related to identifying, classifying and mitigating vulnerabilities.

## External and internal vulnerability scans

## Vulnerability scans can be performed from outside or inside the network or the network segment being evaluated. You can run external scans from outside the network perimeter to determine the exposure to attacks of servers and applications that are accessible directly from the internet and internal vulnerability scans to identify flaws that hackers could exploit to move laterally to different systems and servers if they gain access to the local network.

## Authenticated and unauthenticated vulnerability scans

## Vulnerability scans can be authenticated and unauthenticated, or credentialed and non-credentialed.

## Unauthenticated scans

## Discover services that are open on a computer over the network; based on which, the scanner searches a vulnerability database and lists what vulnerabilities are likely to exist on those systems.

Authenticated scans

They use login credentials to collect more detailed and accurate information about the operating system and the software installed on the scanned machines. Some programs might not be accessible over the network but can still have vulnerabilities that are exposed to other attack vectors such as, opening maliciously crafted files or accessing malicious web pages.

Penetration (pen) testing

It can be used to validate flaws and determine actual risk much better than relying on vulnerability databases. The following are some of the questions that should be asked when evaluating vulnerability scan results:

* Is this vulnerability a true or false positive?
* Could someone directly exploit this vulnerability from the internet?
* How difficult is it to exploit this vulnerability?
* Is there known, published exploit code for this vulnerability?
* What would be the impact to the business if this vulnerability were exploited?
* Are there any other security controls in place that reduce the likelihood and/or impact of this vulnerability being exploited?
* How old is the vulnerability/how long has it been on the network?

###### Web application vulnerability scanners

They are specialized tools that can find vulnerabilities in websites and other web-based applications. Web application scanners look for [common types of web flaws](https://owasp.org/www-project-top-ten/) such as [cross-site scripting (XSS)](https://www.csoonline.com/article/3269028/what-is-cross-site-scripting-xss-low-hanging-fruit-for-both-attackers-and-defenders.html), [SQL injection](https://www.csoonline.com/article/3257429/what-is-sql-injection-how-sqli-attacks-work-and-how-to-prevent-them.html), command injection, and path traversal.

###### Continuous vulnerability management

Periodic vulnerability scanning can uncover blind spots. You must do it continuously to have an effective vulnerability management. However, increased scanning frequency must be accompanied by increased patching frequency to be effective. Center for Internet Security (CIS) encourages organizations to deploy automated software update tools and policies in order to ensure their systems and applications receive the latest security patches as quickly as possible.

##### Threat detection

Threat detection is the practice of analyzing the entirety of a security ecosystem to identify any malicious activity that could compromise the network. If a threat is detected, then mitigation efforts must be enacted to properly neutralize the threat before it can cause any harm.

*Security is a continuous process, and nothing is guaranteed.*

The concept of threat detection is multifaceted. Even the best security programs must plan for worst-case scenarios, e.g., when someone or something has slipped past their defensive and preventative technologies and becomes a threat.

When it comes to detecting and mitigating threats, speed is crucial. You must be able to detect threats quickly and efficiently to stem any further harm/loss of your assets. A business’s defensive programs can ideally stop most of threats, because often they've been seen before, known threats. There are also unknown threats that an organization aims to detect. This means the attacker is using brand-new methods or technologies never seen before.

Known and unknown threats must be actively pursued in your environment.

##### Leveraging threat intelligence

It is a way of looking at signature data from previously known attacks and comparing it to enterprise data to identify threats. It only works for known threats.

###### Analyzing user and attacker behavior analytics

With user behavior analytics, you can gain a baseline understanding of what normal behavior for an employee would be: what kind of data they access, what times they log on, and where they are physically located. That way, a sudden outlier in behavior stands out as unusual behavior and something a security analyst may need to investigate.

###### Setting intruder traps

It is the concept of luring the attacker. An intruder trap could include a honeypot target that may seem to host network services, or “honey credentials” that appear to have user privileges needed to gain access to sensitive systems or data. When an attacker takes the bait, it triggers an alert so the security team knows there is suspicious activity in the network that should be investigated.

###### Conducting threat hunts

A threat hunt enables security analysts to actively go out into their own network, endpoints, and security technology to look for threats or attackers that may be lurking as-yet undetected. This is an advanced technique generally performed by experienced security and threat analysts.

###### A two-pronged approach

Threat detection requires both human and technical elements. The human element includes security analysts, patterns in data, behaviors, and reports, and those who can determine if anomalous data indicates a potential threat or a false alarm.

Threat detection technology also plays a key part in the detection process. No single tool will do the job. A threat detection program should leverage: Security event threat detection technology to aggregate data from events across the network, including authentication, network access, and logs from critical systems; understand network traffic patterns and monitor traffic within and between trusted networks, and to the internet.

##### Alerting

Security alerts help organizations quickly detect cyber-attacks. The biggest challenge for a Security Operation Center (SOC) is simply understanding normal behavior, false positives and actual threats.

###### Privileged User & Account Monitoring

* Privileged user accounts are one of the most common security weaknesses for organizations.
* Hackers regularly attempt to obtain privileged user accounts and ways to escalate privileges.
* Using a privileged account, they can potentially bypass firewalls or Intrusion Detection Systems (IDS).
* Dashboards should be created to track privileged user activity.

###### Abnormal External Communication

* It can take place in your network through an abnormal port or protocol.
* Your firewall can help with traffic filtering but may not catch everything.
* External communications are filtered, monitored, and blocked. Any traffic that gives cause for concern should be validated against your security policy and reviewed against malicious patterns.

###### Acceptable Use Policy Violations

* Acceptable Use Policies (AUP) can be something that your employees signed when they first onboarded but rarely follow now. They are important to protect your organization from malicious activity on your network but are often not enforced or monitored.
* Security alerts should be set up and dashboards to review AUP violations.

###### Data Exfiltration/Unusual Port Activity

Threat actors can infiltrate frequently used ports to avoid firewalls and Intrusion Detection System (IDS). Commonly used ports for data exfiltration include common Internet services, expecting they’ll have any rule for that port on firewalls: TCP: 80 (HTTP), TCP: 443 (HTTPS), TCP/UDP:53 (DNS).

###### Hackers typically use the following techniques to conduct an attack

* Backdoors, Web applications, File transfer protocol (FTP). Use SFTP or FTPS instead, or even better, use a secure cloud provider
* Alerts can be set up using the network intrusion and prevention system logs to identify any of the suspicious port activity mentioned above. You may even uncover that the traffic represents malware infiltration. You can also set up specific security alerts when data is shared externally more than normal.
* Baseline your normal traffic and trigger alerts when traffic outside the baseline occurs.

##### File Integrity Monitoring

* Another area your analysts may want to look at closely is File Integrity Monitoring (FIM). The auditing policy should include unexpected changes in a file’s status and alert on it, through NG Endpoint Protection (EPP), SIEM or both.
* File access auditing can tell you which files have been viewed, what programs are using those files, and if any files were deleted or created. Any suspicious files should run through the antivirus tool or next-gen endpoint protection solution.

#### Browser based administration

For browser-based administration, elevated privilege access is required. Unfortunately, mismanaged, unattended and forgotten privileged accesses are the most attractive targets for attackers. So, it is incumbent that securing privileged access be the top security priority because of the significant potential business impact (and high likelihood) of attackers compromising this level of access. The list below shows best practices for browser-based administration to be secure:

##### Protect admin accounts

* Require MFA authentication for admin accounts
* Use security keys for two-Step Verification
* Don’t use an admin account for daily activities
* Don’t stay signed into an admin account

##### Manage admin accounts

* Set up multiple admin accounts
* Create per-user admin role accounts
* Delegate daily admin tasks to user accounts

##### Monitor activity on admin accounts

* Set up admin email alerts
* Review the Admin audit log

##### Identify and categorize accounts in highly privileged roles

After turning on Azure AD Privileged Identity Management, view the users who are in the following Azure AD roles:

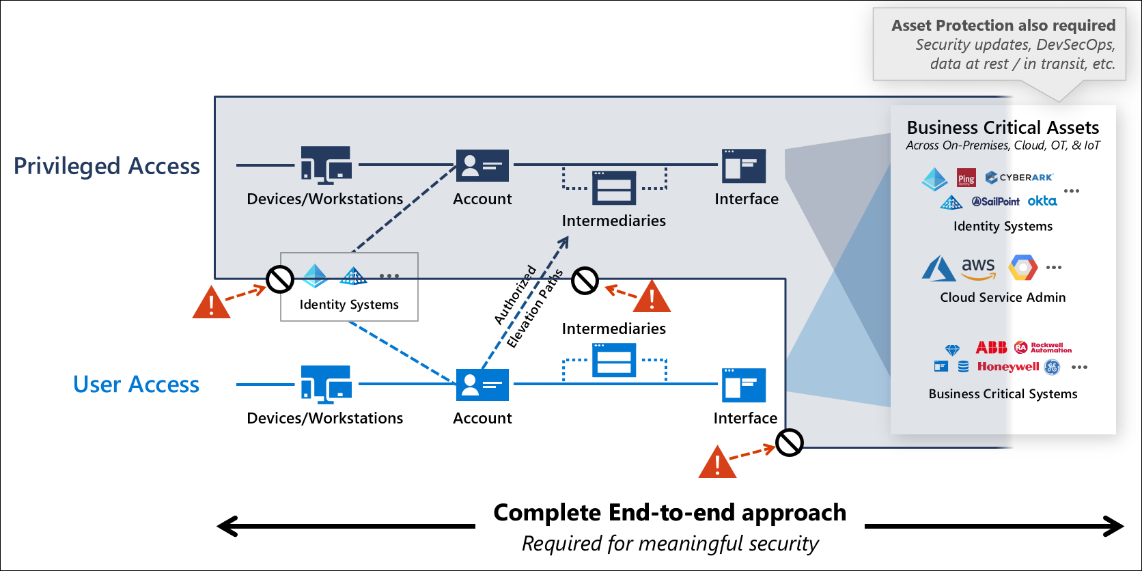
* Global administrator
* Privileged role administrator
* Exchange administrator
* SharePoint administrator

Remove any accounts that are no longer needed in those roles. Then, categorize the remaining accounts that are assigned to admin roles:

* Assigned to administrative users, but also used for non-administrative purposes (for example, personal email)
* Assigned to administrative users and used for administrative purposes only
* Shared across multiple users
* For break-glass emergency access scenarios
* For automated scripts
* For external users

##### Prepare for admin account recovery

* Add recovery options to admin accounts
* Gather information for password reset
* Enroll a spare security key
* Save backup codes ahead of time
* Set up an additional admin (at least 2)



*Recommended privileged access strategy*

Securing privileged access effectively requires a holistic approach combining multiple technologies to protect and monitor those authorized escalation paths using Zero Trust principles including explicit validation, least privilege, and assume breach. It requires rigorous operational execution to build and sustain assurances over time.

##### Just-in-time access methodology

Elevate users in real-time to provide elevated and granular privileged access to an application or system to perform a necessary task. Cybersecurity industry analysts recommend JIT access as a way of provisioning secure privileged access by minimizing standing access.

#### Optum CareData Platform Login Concerns - Snowflake, FHIR, Tableau, and Azure Data Factory

While a lot of good work is being done at Optum CareData platform, there are still a couple of areas that are not adequately addressed. They are user management and access controls. It is not because of lack of knowledge, experience or being aware of vulnerabilities they pose and inefficient resource usage they cause. Instead, the decisions are made at some other part of the organization that Optum CareData platform must abide by.

There is a lack of an enterprise grade solution that provides federated authentication, single sign-on and MFA capabilities for internal (integrated) and external (non-integrated) users. In lieu of that, the DWaaS team has been trying various solutions and presenting them to the EIS group for their approval and other stakeholders. Unfortunately, all those solutions keep getting rejected. For example, when DWaaS team tried to authenticate external users, PingFederate authentication kicked in and the login failed; they tried a user who had Citrix, it worked as expected via Citrix, but the EIS rejected that approach due to scalability concerns; the VPN addresses provided by the EIS group did not work either. To compound the problem, it is almost impossible to get all those stakeholders in a meeting, so that they can all come to a consensus. But that is proving to be a challenging task.

The latest patchwork of a solution is –

* For integrated users, users need to create a Secure request to get access to the Snowflake. The user accounts are created automatically when an access request is made in Secure;
* Non-integrated users are invited by email. Non-integrated users are also using email id but, they must use their password along with the email id. They are setup as local users in the Azure Active Directory (AAD).

***Note:*** *This was meant to be a temporary solution for only a select few users, but lately there are quite a few of them (over a hundred at the last count). It is burdening the organization with non-integrated user management and creating potential vulnerabilities. It is a patchwork, at best.*

Service accounts are treated as non-integrated users. For example, when a request was made in Secure to create a service account and the response received was, an email address is required, but service accounts do not have email accounts.

This type of a patchwork of a solution exposes OCDP to risk vulnerabilities, increases user management burden, presents opportunities for mischief, exposure to regulation violations; potentially damage Optum reputation (brand) and even create financial liabilities. It is a known vulnerability and should be addressed ASAP. Because OCDP security is only as strong as its weakest link.