Power CAT

Administering a low-code intelligent automation platform

Enterprise Deployment for RPA and more in Power Automate

Summary: This whitepaper outlines key considerations for planning, deploying, and managing an Automation Center of Excellence (CoE) for hyperautomation scenarios in Power Automate.

Writers: Anitha Natarajan, Jonathan Eckman, Lav Gupta, Brent Wodicka

Technical Contributors: Apostolis Papaioannou, Kent Weare, Pranav Rastogi, Vishwas Lele, Gautier Chastan, Kathy Osborne, Rakesh Krishnan, Amit Bhambri, Ashvini Sharma, Jonathan Kendall

Published: December 2021

Version: 1.0

Contents

Contents 2
Overview 3
Purpose of this whitepaper 4
Empower 6
Discover & Plan 7
Defining a Center of Excellence 7
Discover new automation projects 8
Compute provisioning 9
VM computing key factors 10
Storage 10
Network provisioning 10
Azure Virtual Machine key factors 12
Security provisioning 12
Accounts provisioning 14
Environment provisioning 16
License & capacity provisioning 18
Data governance 20
Design 22
Organization guidelines 22
Design for scale, throughput, and resiliency 22
Common components fundamentals: logging, credential management, and testing 22
Logging 22
Credential management 23
Testing 23
Error handling 23
Monitoring and alerting 23

Reusability and share 24
Build & Test 25
Develop automation 26
Testing 27
Deploy & Manage 28
Network deployment 28
Azure high availability 29
Machines and machine groups 30
Application Lifecycle Management 31
Infrastructure deployment 31
Application deployment 32
Monitoring 35
Insights 35
Visualize 37
Respond 38
Solution 39
Secure & Govern 39
Identity and access 40
Network infrastructure 46
Endpoint / Application 46
Data and Compliance 48
Proactive and reactive monitoring 50
Nurture 52
Nurture models 52
Nurture team roles 53
Real-world implementation example 54

Overview

Robotic process automation (RPA) and hyperautomation in general remains one of the fastest growing segments of the software market. It is increasingly seen by organizations as way to achieve operational efficiency by integrating disparate enterprise applications. Microsoft Power

Automate is striving to provide customers with a single intelligent automation service that covers all aspects of hyperautomation: API, UI (RPA), and AI.

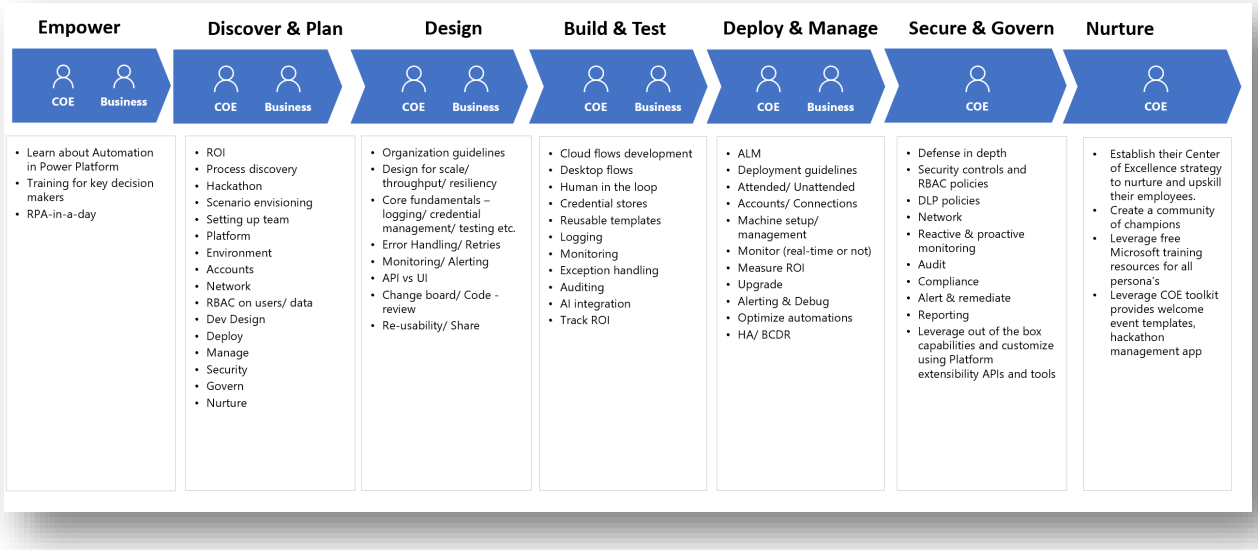
One of the challenges organizations face in accomplishing digital transformation is business process automation spanning legacy and modern technologies. RPA is a key technology to address many such challenges, but it requires a patchwork of automation services that need integration and management before the real work can get done. In enterprises, this involves integration with complex legacy systems and meeting stringent compliance and security requirements. Given the organizational complexity, RPA deployments can take time.

By following the best practices in this document, an Automation Center of Excellence (CoE) can ensure that the organization can realize automation and productivity objectives. By democratizing who can and build out automations, they can allow the organization to scale their automation practice.

Purpose of this whitepaper

The purpose of this whitepaper is to streamline the exploration, implementation, security, governance, and scaling of process automation across your organization

The whitepaper is based on HEAT (Holistic Enterprise Automation Techniques), which is a collection of learnings from deploying hyperautomation solutions at many enterprises.



Here is a quick summary of various sections:

Empower: The start of any successful automation project is to ensure that the key stakeholders understand the automation capabilities of the platform. In this stage, users new to Power Automate can learn about the automation capabilities in Power Automate.

Discover & Plan: This section begins with guidance on discovering existing processes within your organization that may be well suited for RPA. Next, this section helps you plan your RPA environment including provisioning compute, network, accounts, and security. Finally, this section discusses the licensing and capacity considerations for the common services that will make up your RPA solution.

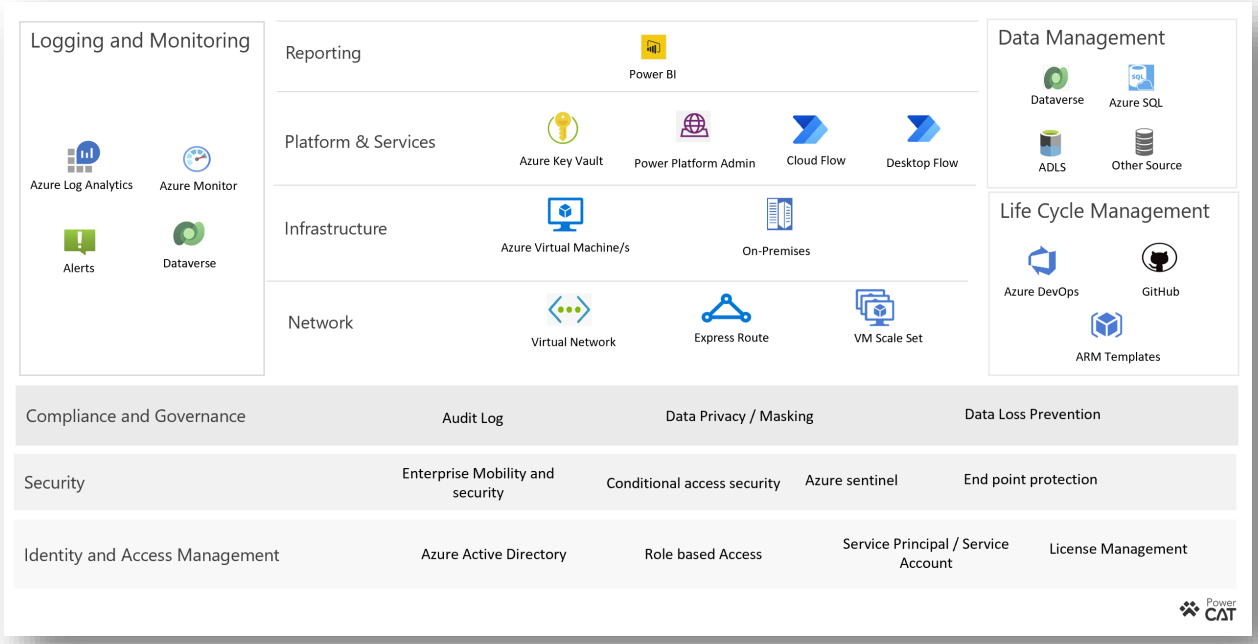
Design: This section begins with a discussion of establishing a consistent set of organization-level policies for your RPA solutions. Next, this section describes the considerations for building high-throughput and scalable RPA solutions. This section also includes a discussion on common components such logging, credential management, and testing. Finally, this section talks about reusing and sharing the RPA solutions you build.

Build & Test: This section dives deeper on planning, building each of the components that attribute to RPA solution, managing on-premises connectivity using direct connectivity, securing sensitive information, custom logging capabilities, and more considerations for building a robust RPA solution. Finally, this section talks about the various aspects of testing the solution.

Deploy & Manage: This section begins with planning the lifecycle of an RPA solution and deploying it safely to production. This section also talks about administering the deployment process using various tools and using the robust analytics framework to monitor the operational aspects of the solution. Finally, it discusses building reports out of the monitoring data to address any operational challenges proactively and reactively.

Secure & Govern: This section begins with a discussion on establishing a baseline security posture for your organization. It addresses extending the framework around different layers of security: identity management, endpoint, network, infrastructure application and data. This section also discusses building a vast and robust framework to enable proactive and reactive monitoring across your organization with respect to your RPA solutions.

The architecture represented below shows the various components and services that will be detailed in each phase.



Empower

The start of any successful automation project is to ensure that the key stakeholders understand the automation capabilities of the platform. In this stage, users new to Power Automate can learn about the automation capabilities in Power Automate.

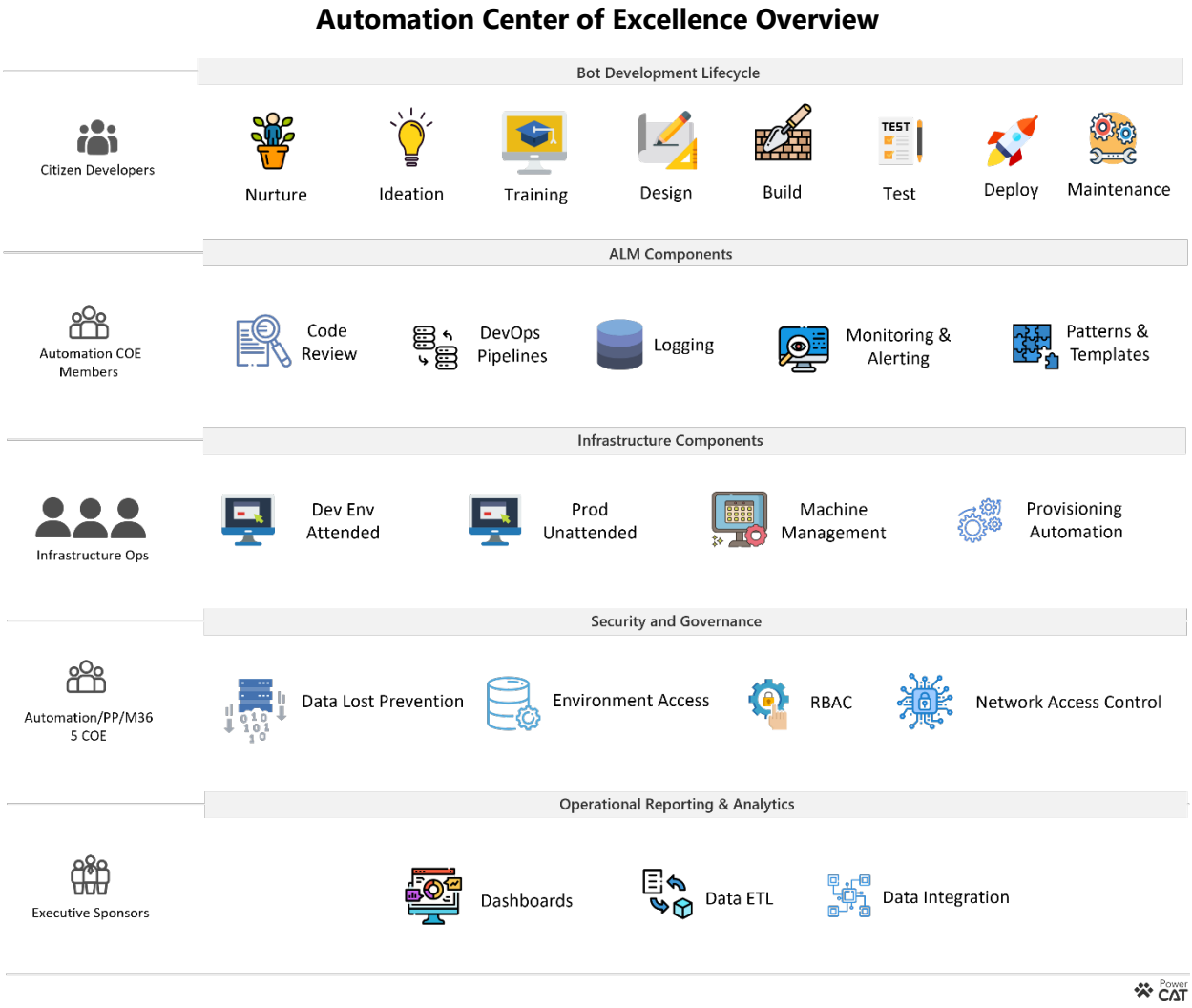
Training resources	**Links**
RPA in a Day Training	https://aka.ms/RPAinaDayPackage
Automate It video series	https://aka.ms/AutomateIt
Maker learning resources	Browse all - Learn | Microsoft Docs
Automation CoE Blueprint	https://aka.ms/autocoeblueprint
Automation CoE Strategy	https://aka.ms/autocoestrategy
HEAT	https://aka.ms/rpapnp https://aka.ms/rpapnpvideo
Manage Power Automate for Desktop on Windows	https://aka.ms/padonwindowspnp
Hyper-automation SAP playbook & video series	https://aka.ms/MicrosoftRPAPlaybookForSAPGUI https://aka.ms/AutomateItSAPSeries
Automation CoE Starter kit – Private preview	https://aka.ms/autocoestarterkitpreview

Discover & Plan

In this phase, the main objective of the Automation CoE is to perform a discovery process on your current setup in terms of network, infrastructure, line of business applications, and business processes and prioritize the next set of actions.

Defining a Center of Excellence

Define CoE roles and responsibilities



Stakeholder roles and responsibilities are shown below.

Roles	Responsibilities	Stakeholders
Business Champions	• Standardize on a platform for long-term success, show impact to business, and empower users	• COE Head • CIO

Roles	Responsibilities	Stakeholders
	• Ensure product meets compliance standards • Monitor process ROI • Business continuity	• Chief Information Security Officer (CISO) • Process stakeholders
Automation Platform Admin	• Establish Azure resource management • Establish security and compliance strategy • Manage users, capacity, licenses • Establish environment strategy • Set up data governance strategy • Long term sustainability • Attrition & succession planning	• Global Admin • Network Admin • Security Admin • Compliance Admin • Dynamics 365 Admin • Power Platform Admin
Automation Product Champions	• Challenge the manual tasks that reduce productivity • Application & architecture review • Library of reusable connectors, APIs, and components • Record unattended flows and develop using pro-dev tools • Metrics and monitoring audit log	• IT Operations Manager • Compliance Manager • LOB Application Admin • Legal and work council • Automation makers (Pro developers and/or citizen developers) • Process stakeholders
Compliance Champions	• Establish data privacy • Maintain data and governance policies • Access audit logs to detect anomalies	• Compliance Admin • Security Admin
Deployment Champions	• Create and manage projects and relevant artifacts • Create and manage work items • Manage build and release pipelines • Support application development community • Create and manage bugs in DevOps	• DevOps Build Admin • Project Admin

Discover new automation projects

The following list outlines the relevant activities to kickstart the discovery and establishing an Automation CoE.

- Discover current processes The process advisor tool quickly captures detailed steps for each process in your organization to help you to better understand places to streamline

workflow automation. The automation recommendation feature helps you identify automation opportunities

- Curate, prioritize and manage automation project ideas

Compute provisioning

Microsoft Azure compute provides the infrastructure you need for all your automation needs to host the RPA process. Azure hosting model offers auto-scaling and high availability benefits to the resources you host. Virtual machine is a key for running RPA processes.

The following checklist offers guidelines for provisioning and computing the workload of the automation process.

Compute Checklist	Activities	Related Resources

Virtual machines for workload	• Define performance based VM configuration template sizes that would sufficiently handle expected workloads of an RPA process
	• Decide on the right configuration for the VM:
	  • CPU
	  • Memory
	  • Disk I/O load
	• Understand the operating system requirements for RPA
	• Determine the storage needs for the automation process
	• Decide the location for the VM, especially important if you have RPA running on on-premises enterprise systems
	• Create and name the VM
		• Sizes for VMs
		• VM configuration
		• Power Automate Desktop pre-requisites
		• Azure VM Deployment Plan

Hybrid infrastructure	• Decide the hybrid infrastructure architecture • Standalone servers • Azure Stack HCI • Azure Stack Hub • Understand the hybrid networking and identity requirements • Decide the policy for high availability, disaster recovery, and monitoring of hybrid infrastructure	• Connect to On-Prem Server • Azure Stack Hub Compute • Azure Stack Hub VMs Recovery for On-Prem VMs

Compute Checklist	Activities	Related Resources
VM Scale Set	• Create the images for the scale set and necessary RPA software • Understand the scaling requirements • Specify policy to roll out upgrades	• Design considerations for VM Scale sets

VM computing key factors

Storage

Best practice is that all Azure virtual machines have at least two virtual hard disks (VHDs). The first disk stores the operating system and the second is used as temporary storage. You can add additional disks to store application data; the maximum number is determined by the VM size selection (typically two per CPU).

It's common to create one or more data disks, particularly since the OS disk tends to be quite small. Also, separating out the data to different VHDs allows you to manage the security, reliability, and performance of each disk independently.

The data for each VHD is held in Azure Storage as page blobs, which allows Azure to allocate space for only the storage you use. It's also how your storage cost is measured; you pay for the storage you are consuming.

Option	Description
Unmanaged disks	With unmanaged disks, you are responsible for the storage accounts that hold the VHDs that correspond to your VM disks.
Managed disks	Managed disks are the newer and recommended disk storage model. They elegantly reduce complexity by putting the burden of managing the storage accounts onto Azure. You specify the size of the disk, up to 4 TB, and Azure creates and manages both the disk and the storage. You don't have to worry about storage account limits, which makes managed disks easier to scale out.

Network provisioning

Attended or unattended automated flows should ideally be run on an individual virtual machine (VM). An obvious advantage of using Azure services (including VMs) is that these resources can be provisioned on demand and run only when they are required, instead of relying on local user machines, dedicating existing hardware, or purchasing hardware to perform automated tasks when your processes reach some level of scale.

This section contains key concepts and a checklist of actions to consider for a secure and scalable environment.

Network Checklist	Activities	Related Resources
Establish secure communication for Azure resources (virtual network)	• Create virtual network • Determine connection need for on-premises services vs another virtual network • Assign unique address space to each virtual network	Virtual network service integration Address ranges Virtual network limitations
Plan for virtual network segmentation	• Create subnets to segment virtual network into one or more sub-networks • Plan to deploy Azure resources in a specific subnet • Secure resources within subnet using Network Security Groups • Plan and reserve some address space for the future • Reduce management overhead by having a few large virtual networks rather than multiple small virtual networks	Subnet Planning
Configure IP address	• Determine and allocate a static IP address to the VM	IP Address
Configure Network Security group (NSG)	• Define the inbound and outbound rules for network traffic to subnet • Ensure the priority is set to be unique	Network Security Groups
Hybrid Network	• Design a hybrid network to give RPA infrastructure secure access to both your on-premises and Azure hosted services • Deploy an VPN gateway to your Azure Virtual Network • Securely connect your on-premises network to Azure using a site-to-site VPN connection, ExpressRoute, or Virtual WAN	Implement a secure hybrid network Create and manage a VPN gateway Connect an on-premises network to Azure Create a Site-to-Site connection using Azure Virtual WAN Set up ExpressRoute for Microsoft Power Platform
URL / IP Allow-listing	• Configure your firewalls to allow the outbound access required for your Power Automate services automation to function.	Adjust communication settings for the on-

Network Checklist	Activities	Related Resources
	• Configure your firewalls to grant your RPA infrastructure connectivity to the on-premises or Azure hosted applications they need to interact with.	premises data gateway Power Automate IP address configuration
Machine groups	• Register dedicated machines for RPA process • Set up machines groups to scale automations	Machine groups for scaling

Azure Virtual Machine key factors

VM Extensions

Azure VM extensions are small applications that enable you to configure and automate tasks on Azure VMs after initial deployment. Azure VM extensions can be run with the Azure CLI, PowerShell, Azure Resource Manager templates, and the Azure portal. You bundle extensions with a new VM deployment or run them against an existing system.

Automated Management

Azure Automation enables you to integrate services to automate frequent, time-consuming, and error-prone management tasks with ease. These services include process automation, configuration management, and update management. If you need a cluster of machines, consider using Azure Automation to operate your infrastructure.

Security provisioning

Security is fundamental to ensure that the right end users are using the data and the related automation technologies in a safe and compliant manner. One of the first things to establish is a foundation of security that includes authentication, authorization, and encryption.

The following checklist outlines the relevant activities to ensure the security components are configured correctly to give the right level of access.

Security Provisioning Checklist	Activities	Related Resources
Identity and access management with Azure Active Directory (AAD)	• Plan the cloud identity model • How will users authenticate (MFA, username password, etc.)? • Will an existing AD environment be extended into Azure, or will a cloud-first approach be used?	Identity Decision Guide Create AAD Users

Security Provisioning Checklist	Activities	Related Resources
	• Establish AAD onboarding, account creation, group assignment processes. • Create (or synchronize) the initial set of AAD users. • Create the initial set of AAD groups and make initial group assignments.	Create AAD Groups

AAD tenant roles for standard directory access	• Map and assign relevant AAD tenant roles
	  • Tenant Admin roles (Global Admin, User Admin, Billing Admin)
	  • MPP roles (Dynamics 365 Admin, Power Platform Admin)
	  • Relevant Microsoft 365 roles (Teams Service Admin, SharePoint Admin, etc.)
	• Establish emergency ("break glass") account access
		Azure AD built-in roles
		
		Use Service admin
		
		Emergency access account

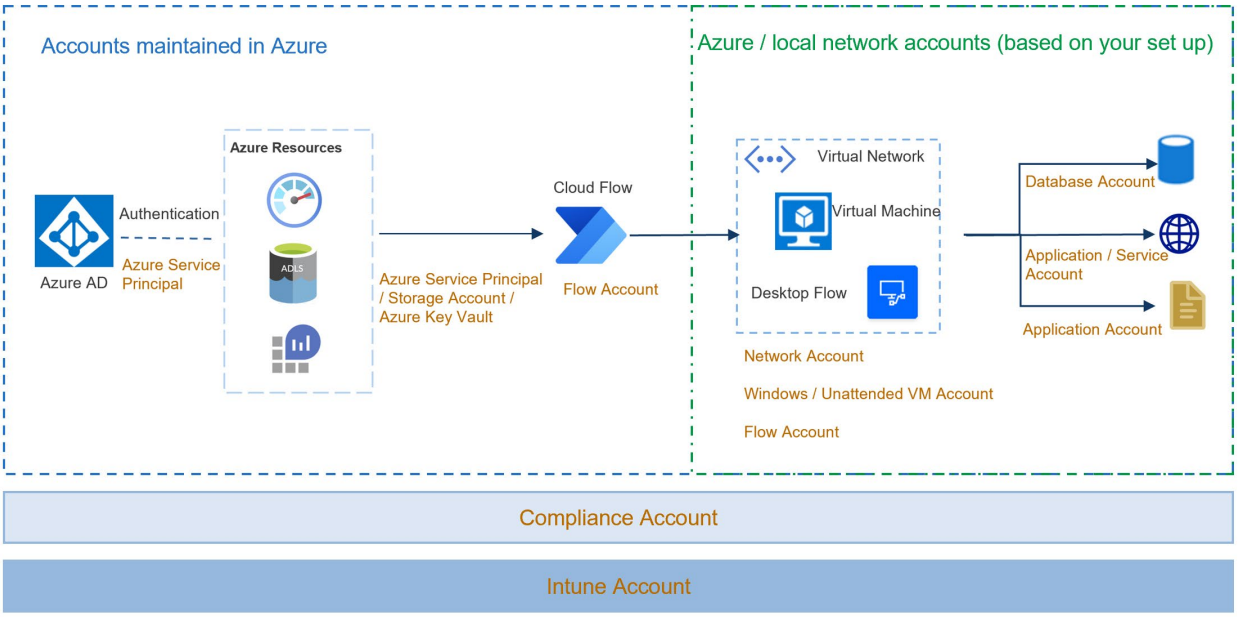
Azure Role Based Access Control (RBAC)	• Identify and document the minimum set of Azure roles required to build and verify the environment.
	• Define the required level of access to Azure resources and resource groups (VM, virtual network, storage, etc.) using RBAC. Plan access for the following data points at each level of access:
	  • Security principal (user / group / service principal)
	  • Role definition
	  • Scope
	• Create any custom role definitions required for the environment (if appropriate after environment initial operating capability)
		Azure Role Based Access Control
		
		Azure Built-in Roles
		
		Create and assign custom role

AAD Service Principal(s)	• Determine the appropriate Azure Active Directory tenant. • Ensure cloud identity model has been established and is understood by appropriate stakeholders. • Determine the initial set of "service accounts" required for operations. Azure DevOps service connections, Automation accounts, etc. • Create an application registration in Azure AD.	Azure AD hybrid topologies Create an Azure AD application and service principal

Security Provisioning Checklist	Activities	Related Resources
	• Determine the intended service for the principal (Power Automate connection, Azure DevOps service connection, etc.) and assign appropriate access to the instance of the AAD App. • This completes the creation of the service principal – which is an "instance" of the AAD app, specifying Azure service and access level.	
Operational Security: Secure credential storage and rotation	• Provision Azure Key Vault in the appropriate Azure Subscription • Store supporting service credentials (keys, passwords, etc.) • Use Key Vault secrets in your flows' data connections • Implement a consistent secret rotation process for all appropriate Azure services (see Secure & Govern section)	Create a key vault using Azure portal Secure secrets in Power Automate with Azure Key Vault Automate Secret Rotation

Accounts provisioning

To deploy RPA in production, there are many components and services that are involved. The figure below depicts a typical set up of automation process and different services and components involved. For each of these services, a connection needs to be established and you must ensure the appropriate account is used.



Key things to consider while planning the accounts:

- Type of account
- Access level
- Business continuity

Below are high-level guidelines in structuring different accounts:

- **Azure AD –** Enterprise identity service that provides multifactor authentication and access to the tenant for all users
- **Azure Service Principal** – Security identity that user-created apps, services, and automation tools use to access specific Azure resources
- **Azure Key Vault** – Stores the application secrets securely. Use Managed Identity to authenticate to the key vault and retrieve credentials
- **Flow Account** – User account used to author and run the automation process, assigned with appropriate license
- **Unattended VM Account –** Windows account on a VM used to connect to the VM and manage resources and processes within the VM
- **Service Account** – Provides security context and ability to access local and network resources, best suited to connect with SQL, SAP, etc.
- **Application Account** – Holds access to the application to perform operations
- **Database Account** – Based on your set up, use application or database account to perform CRUD operations
- **Compliance account** – Enables and configures auditing on all the above accounts
- **Intune Account** - Maintains device compliance, configures profiles and policies, and performs remote tasks

- **Emergency Account** – Used for emergency purposes to gain access to a system or service that is not accessible under normal controls

The following checklist provides guidance on understanding the account needs and activities to establish the accounts:

Accounts Type Checklist	Activities	Related Resources
Types of Account	• Finalize your application infrastructure • Define the account needs • Create the necessary accounts in your tenant • Ensure appropriate license is available for the accounts	Create Service Principal Create Azure Key Vault Assign license to users
Account auditing	• Establish Admin account auditing needs • Enable auditing on all privileged accounts	Auditing for admins

Environment provisioning

Establishing controlled environments is important for any kind of development. Because RPA solutions depend on resources, networks, application UI elements, images, or coordinates that are likely to change, it is important to establish a small number of well-defined environments with a controlled promotion process.

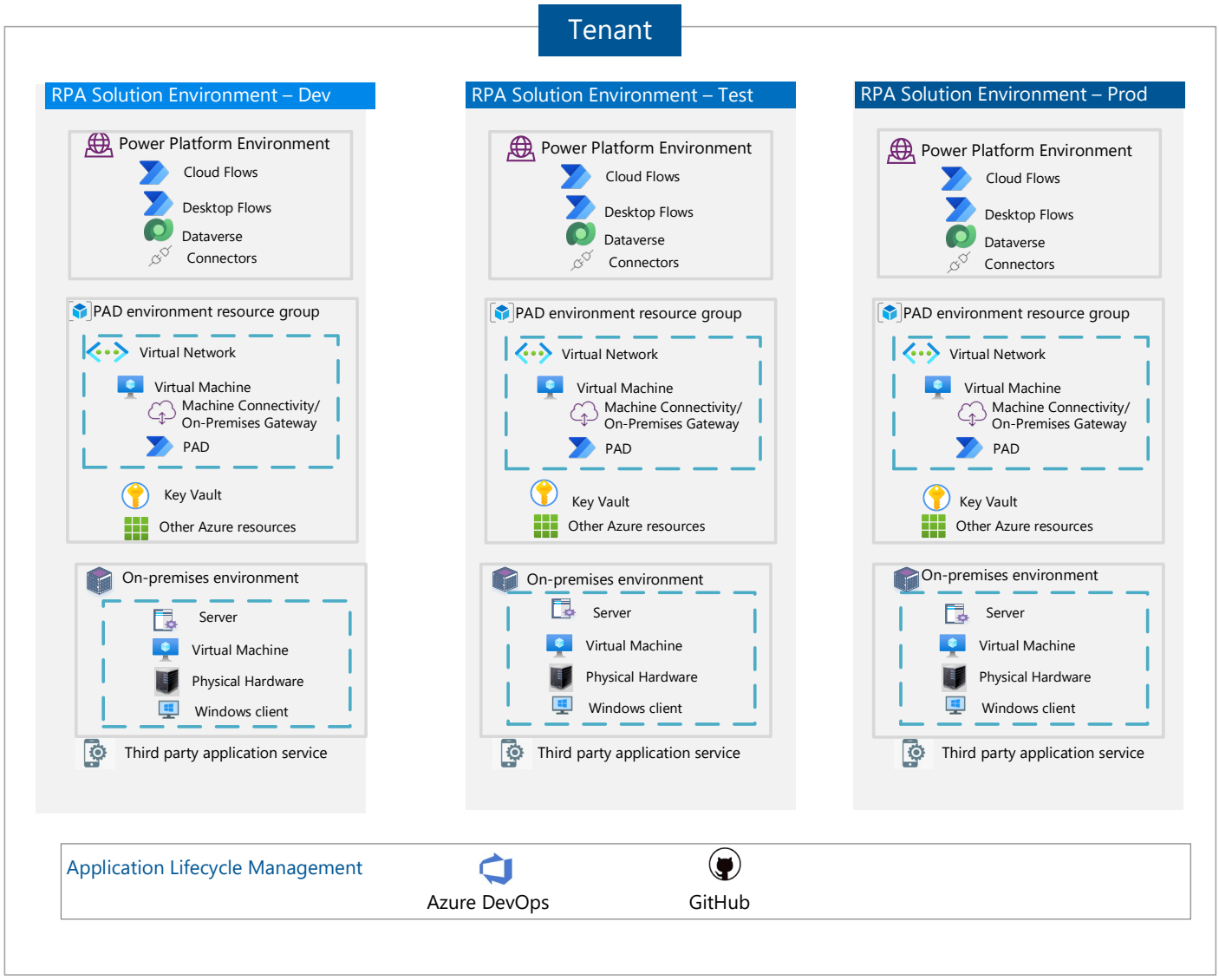
For most RPA setups, consider provisioning three environments for each logical functions within your organization: Dev, Test and Production. An RPA solution environment can also be set up to address compliance requirements, such as geographical location.

An RPA solution environment comprises Power Automate environments, Azure resources (resource groups, VMs, storage, Azure Key Vault etc.).

Development - An environment in which an RPA developer can build and test their desktop flows. Testing is limited to unit-test style testing.

Test - An environment where Q&A engineers can run a series of end-to-end tests to ensure that a desktop flow can be deployed to production. It is important to ensure that the test environment mimics the production environment in terms of dependent services such as Azure Key Vault and Azure virtual network.

Production - An environment where the automation is put into actual use.



The following checklist provides the list of actions to establish a comprehensive and functional environment.

Environment Checklist	**Activities**	**Related Resources**
Environment strategy	• Determine the workload • Determine the region / geographic location for the environment • Define # of environment groupings based on target audience/functions	Assess workload Azure Regions
Azure Resource Group	• Create resource groups for each target function • Define resources per resource groups	Regions Manage Resource Groups

Environment Checklist	Activities	Related Resources
	• Define Azure Key Vault and identity management for each resource group	
Power Platform Environment	• Create automation environment for each target function	Types of environments Create and manage environments
Environment Database	• Create database in the environment • Enable auditing on RPA tables	Create Database
Manage environment	• Assign admins and environment privileges • Manage environment from admin center • Avoid the use of default environments, because everyone in the organization has access to the default environment • Limit environment creation to prevent unaccounted capacity consumption • Establish process to allow users to request a new dedicated environment • Group resource on-premises for each target function	Environment permission Environment administration Power Platform Admin Center Default Environment
Azure Deployment Environment	• Create target deployment environment	Create DevOps Environment

License & capacity provisioning

There are a few distinct aspects to be considered as you plan the license requirements for RPA solutions, depending on the resources, infrastructure, applications, and data needs. To plan and estimate licensing based on the extensive set of services offered, refer to Azure pricing.

Following are a few key considerations as you plan license provisioning.

VMs – A subscription will be charged for every VM, based on two aspects:

- **Compute costs** - priced on a per-hour basis but billed on a per-minute basis. For example, you are charged for only 55 minutes of usage if the VM is deployed for 55 minutes. You are not charged for compute capacity if you stop and deallocate the VM since this action releases the hardware.
- **Storage costs** – priced based on the storage the VM uses. The status of the VM has no relation to the storage charges that will be incurred; even if the VM is

stopped/deallocated and you aren't billed for the running VM, you will be charged for the storage used by the disks.

Security – Safeguard and maintain control of keys and other secrets used in the RPA solution using a service such as Azure Key Vault. To see a comparison between the Standard and Premium tiers, see the Azure Key Vault pricing.

Automation platform – Workload and capacity provisioning using various application services: Power Automate cloud flow, desktop flow, AI Builder. For detailed license guide, please refer to License Guide.

Management and Governance – App management, security & compliance, and data and information protection.

Application lifecycle management – Licensing needs to plan continuous integration / continuous delivery across different environments.

Below is a checklist of guidelines to follow when determining your licensing needs:

Checklist Item	Activities	Related Resources
Compute and Network Services	• Determine the services and workload needs for automation • Predict the future workloads to accommodate • Evaluate the virtual network set up and on-premises connection needs	Azure Licensing FAQ Azure VM Pricing Virtual Network pricing
Security	• Synchronize on-premises directories • Utilize services to safeguard and control keys and other secrets in RPA solution	Azure Key Vault pricing Azure AD Pricing
Automation workflow	• Decide between the per user vs per app license model based on expected workload • Determine the attended vs unattended license need • Determine the capacity need • Determine and monitor the API usage and limitation	Power Automate Licensing Licensing FAQ Capacity Management Subscription add-on API Limits Power BI Licensing
Management and Governance	• Evaluate the security and compliance needs for the organization • Evaluate the information protection and governance needs for RPA solution	Microsoft 365 comparison guide

Checklist Item	Activities	Related Resources
Application deployment service	• Determine the application deployment model (Refer to Deploy & Manage section below)	Azure DevOps licensing GitHub pricing

Data governance

One of the critical aspects to be designed in all Automation scenario is a data governance model that is relevant to business usage scenarios. This governance model typically consists of data retention, Role Based Access Control, PII data masking and data encryption.

The following checklist outlines the critical items, the relevant activities to ensure data governance is modeled correctly:

Checklist Item	Activities	Related Resources
Azure data privacy	• Decide how to segregate and isolate data in Azure • Decide data protection policy using redundancy • Determine where your data is located • Decide policy for data destruction, if applicable • Monitor for unauthorized transfer of sensitive data • Allow/ disallow connectors based on whether Business Data is allowed or not • Allow/ disallow select IP addresses to use to connect to data sources	Azure Storage Redundancy Data residency Manage Data Exfiltration Data protection policy Endpoint filtering in Power Platform
Role Based Access Control	• Define permission scope for applications and services – Virtual Machines, Azure Key Vault, etc. • Use RBAC to assign specific permissions to users, groups, and applications based on their role in your organization • Leverage service-specific security controls	Azure role-based access control
Personal Identifiable Information (PII) masking	• Determine the business process and any PII to be protected • Use Dataverse security features to enforce field-level restriction to store sensitive data • Use features like secure input/output and sensitive text to prevent exposure of PII data	Field level security Secure Inputs and Outputs

Checklist Item	Activities	Related Resources
	• Use service principal or environment variable to protect the authentication key for the API connection	
Data encryption	• Protect data in Power Platform environments using encryption at rest • Secure application data by encrypting it with Azure Key Vault-managed secret keys • Define the various components taking part in data protection implementation • Choose a key management solution • Ensure data protection in-transit • Plan for VM disk encryption	Dataverse key management Azure Key Vault - Connector Azure Encryption Protect data at rest and in transit Azure VM Disk encryption
Data retention and management	• Plan for environment backups • Define data loss prevention policy at tenant vs environment level • Define allowable / restricted connectors • Understand the retention limits for RPA	Environment backup RPA retention limits
Cross Region Deployment	• Use Azure Policy to control data residency • Understand Azure geographies and regions mapping to avoid data residency breaches during replication • Ensure your RPA solution aligns with the redundancy and disaster recovery strategy of your organization's cloud and Power Platform strategy	Azure Policy Azure Geography and Regions International Availability of Power Platform Protect and recover in Azure Power Platform datacenter regions Azure datacenter regions

Design

In this phase, the main objective of the CoE is to create design guidelines/ templates for developers to use to follow best practices based on the business requirements and technical planning done in the previous phase.

Organization guidelines

Each business organization has its own objectives, priorities, and preferences. To design a solution that achieves the organization's objectives:

- Identify the business-critical steps in the process and prioritize them.
- Design to scale the solution as per the business commitment.
- Design to protect and recover the solution.
- Understand the impact of potential infrastructure outages, to evaluate the cost and ROI.
- Make sure to always have inventory and visibility of business-critical processes.

Design for scale, throughput, and resiliency

An RPA solution should yield high throughput to impact organizational processes. The solution should be able to handle increased load by scaling the instances of the desktop flows running. Resilient systems are designed to handle failures and they self-correct with minimal impact on the availability and throughput.

- Design infrastructure for the planned capacity.
- Ensure system readiness for parallel execution.
- Design stateless components so that the execution does not have local dependencies.
- Avoid service failures with Health Check.
- Design for resiliency with redundancy; use Machine Groups to run your solution at scale.
- Create clusters of gateways/machine groups for high availability and load balancing.
- Consider storage technology that scales proportionally.

Common components fundamentals: logging, credential management, and testing

Design a framework of common components that codifies the CoE best practices.

Logging

Logging allows you to collect and correlate log data from applications, services, and infrastructure in the RPA solution. It provides insight into the execution state, failures, and informational messages of the system. Logging is crucial for tracing the flow and progression of the application and helps monitor system health and availability.

- Logs pertaining to a transaction or event chain should have a uniquely identifiable identifier (Correlation ID).
- Design a system where the logs are centrally stored.
- Make sure that the recorded log meets regulatory guidelines and doesn't contain any PII data.

- Logging should be asynchronous and should have no performance overhead to the application.
- Log data should have a structured format for consistency and for ease of querying.

Credential management

A business application may use several credentials to connect with different systems. These credentials should be stored securely and easily retrievable during the execution. The RPA solution should be designed to use on-premises, cloud, or hybrid identity to seamlessly use desktop flows. For any non-default connections and accounts, secure key management services can be used to store secrets and credentials.

Testing

A cloud business application is composed of multiple services. It's hard to test them. Therefore, the system should be designed in such a way that each component can be separately tested in isolation. Each component should be designed using service-oriented interface to be able to be tested by mocking or stubbing the dependent services.

Error handling

In a distributed system, failures can occur due to several reasons, such as underlying infrastructure, third party service outages, and issues with other dependent systems. Design an error handling mechanism to ensure your application can recover from errors to avoid lost data and missed events.

- Applications running on cloud are prone to transient faults. Handle these errors with retry policies. See: Handling Transient Faults in Azure
- Provide appropriate time-outs while making service calls.
- Handle errors from applications as well as from the platform in use.
- Implement exception handling with the Try-Catch-Finally pattern. See: Try Catch Finally Pattern for Power Automate
- Introduce an error handling pattern, such as circuit breaker, to avoid cascading failures due to unanticipated events.
- Use application health probes to check availability of the critical part of the systems such as load balancers and traffic managers.

Monitoring and alerting

Design a monitoring framework that shows you the real-time operational analytics of your solutions, to help operations teams constantly monitor the efficiency and effectiveness of the digital workforce and easily measure operational performance.

Your monitoring framework should:

- Help measure the efficiency, effectiveness, and operational performance of the solution.
- Notify respective teams through alerts in case of any unanticipated events.
- Help teams improve the resiliency of the system by automating a self-heal process.

- Help governing bodies with reports and dashboards.

Reusability and share

Reusable components allow developers to quickly build solutions without having to maintain the same code in multiple places. Reusability and sharing components across teams will help the RPA adoption grow at a rapid pace. One of the advantages of CoE-driven design process is the visibility of components that are repeatedly used. It will help discover and identify candidate components for reusability.

Please note that as the number of environments increases, so does the complexity of keeping the versions of reusable components in sync, which is why this becomes an important consideration when defining your overall automation environment strategy.

To create reusable components:

- Clearly define boundaries such as pre-requisites, input, and output.
- Design with portability in mind.
- Build components where business logic is decoupled.

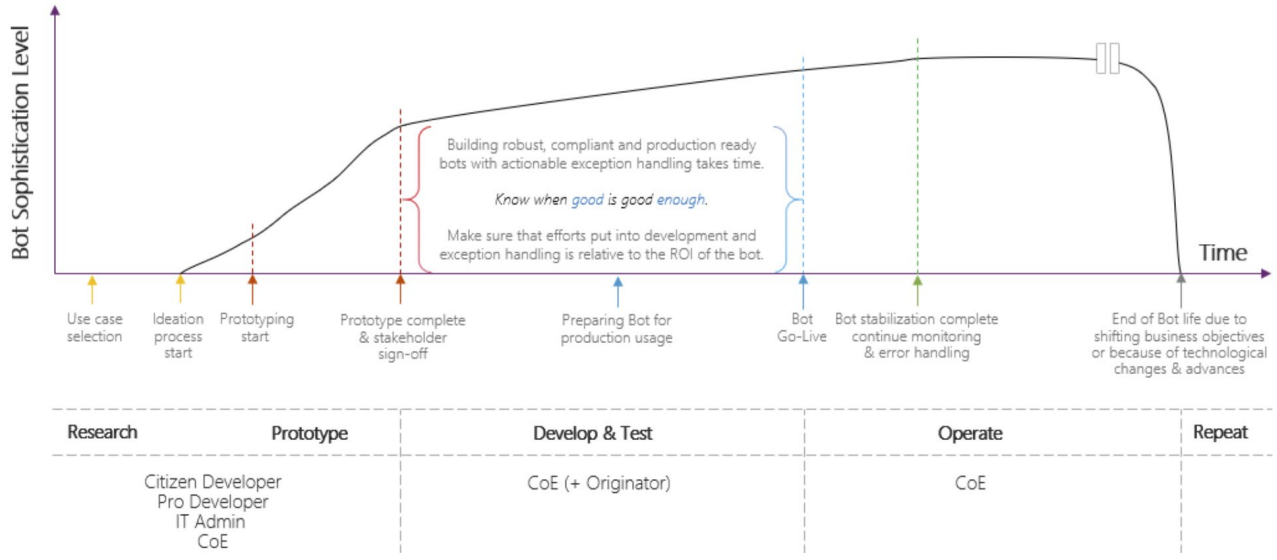
The following is a checklist to design a reliable and scalable RPA solution:

Checklist item	**Activities**	**Related Resources**
Design scalable environment	• Determine the right VM size • Design load balancing architecture using gateway clusters • Design stateless component so that the execution does not have local dependencies • Design for resiliency with redundancy; use Machine Groups to run your solution at scale	Scalable Azure architecture On-premises gateway cluster Machine Groups
Application Lifecycle Management	• Employ solutions to transport flow components between environments • Use environmental variable	Solution overview Using Environment variable
Error Handling	• Employ guidelines for handling exceptions in cloud flows • Employ guidelines for handling action-level and block-level exceptions in desktop flows	Cloud flow exception handling Desktop flow exception handling
Templates and reusable components	• Create templates to handle errors – eg: Try Catch Finally, Retry Process, Configure Run After options	Build template gallery Creating child flows

Checklist item	Activities	Related Resources
	• Identify reusable logic across RPA solutions and build common components	
Enable Auditing	• Enable auditing for taking proactive steps to avoid, detect, and remediate issues • Plan for auditing custom entities to proactively monitor and act on user activities	Power Automate Audit logs Dataverse auditing

Build & Test

To successfully build and test a compliant, scalable solution, establish organization-wide guidelines and best practices to reduce human errors and build a reliable solution. It's important to understand that building sophisticated, robust, and impactful RPA solutions that span multiple legacy systems takes time. And, as shown in the following image, most of the time is spent on production readiness, including advanced retry and exception handling.



Here are a few key guidelines to consider while developing an RPA solution:

- Automation of Power Automate for desktop installation on machines
- Code organization
  - Child flows – Allows reuse in multiple places, easy maintainability
  - Credential management – Use Azure Key Vault as the central cloud repository for storing credentials to an app or service
  - Sensitive data handling - Secure your passwords and secrets
- Logging and exception handling

    - Custom logging – Allows you to monitor the RPA execution progress and handle appropriate exceptions
    - Exception handling
        - Notify on a failure
        - Try / Catch block to continue the process on an exception
        - Action level or Block level exception handling
- Solution Application Lifecycle – allows to transport apps and components from one environment to another or to apply a set of customizations to existing apps
- Testing and debugging
- Understand and gauge the current as-is process and establish a healthy baseline
- Understand the business needs in terms of new and unique users, duration taken for a process to complete, and navigation patterns.

Develop automation

There are many aspects of an RPA solution that can be automated to effectively standardize infrastructure deployment and process automation and continuously audit the key aspects of your business process flow.

Below you will find guidelines on developing a scalable RPA solution.

Checklist Item	Activities	Related Resources
Manage desktop flow deployment (automation)	• Automate the installation of Desktop flow on the machines	Install Power Automate for Desktop
Code organization	• Evaluate the business process and build a plan to accommodate asynchronous processing • Build a reusable solution by creating child flows • Establish a plan to handle sensitive data in cloud / desktop flow • Establish a plan to manage credentials and secrets using Azure Key Vault	Child flows Sensitive data handling Secure secrets in Power Automate with Azure Key Vault Use sensitive text
Logging and Exception handling	• Establish guidelines on handling exceptions in cloud flows using techniques – Configure run after, Try/catch, Error • Establish guidelines on handling exceptions for desktops flow – Action-level and block-level exception	PAD exception handling Template gallery Auditing in Dataverse

Checklist Item	Activities	Related Resources
	• Create organizational template for custom error handling • Enable auditing to capture the custom logging needs	
Connectors	• Evaluate which services / apps to connect to • Create custom connectors as needed • Enable data protection policies in compliance with your organization policies • Create guidelines to reuse the connectors across projects	Connectors Custom Connectors Data protection in connectors
RPA Apps and AI Builder	• Evaluate the process to identify the best use of AI Builder model processing • Identify and plan for integration with legacy apps, or third part of apps via API or UI capture model	AI Builder in RPA SAP Integration
On-Premises Gateway / Machine Groups	• Evaluate your choice between On-Premises Gateway / Machines based on your infrastructure	Silent install of On-Premises Direct machine connectivity
Application Lifecycle Management	• Set up a deployment plan to – create, update, upgrade, and patch a solution • Establish guidelines for managed and unmanaged solution package • Identify deployment tools to run pipeline deployments	Solutions Solution layers

Testing

For an RPA automated process to successfully transition from build to production, optimal and concise testing is extremely important.

Two main focal points for a successful test plan:

- RPA Testing – Validating the automated business process
- Business Component Testing – Validating other components, workload management, process branching, exception handling and performance measurement

Below are key guidelines to ensure an efficient test plan:

Checklist Item	Activities
Validate against as-is process	• Ensure the RPA tool is interacting with the applications and services as expected

Checklist Item	Activities
	• Ensure the key read/write actions to the applications and services • Ensure the navigation is defined as expected
Validate all branches of process	• Identify all possible branches of process – infrequent, outlying cases, presumed path, etc., • Do not overlook a branch of action
Validate error handling	• Test for system unavailable exceptions accounting for service and application downtime • Test for business-rule-driven exceptions – ex: data format or missing data • Account for hand-off to human on business referral exception • Account for notifications on business rule driven exceptions • Account for uncommon UI interruptions – eg: pop-ups
Validate data handling	• Include data validation – missing data, incorrect format • Ensure the sensitive data is secured and not exposed at any level
Validate user access	• Understand the business process access control at data, process, and approval level • Set up various test accounts to ensure data is secured in the execution
Validate Load and performance expectations	• Set a baseline on the time to run a process, number of simultaneous processes expected (load expectations), resource consumption, etc. • Validate the load and performance expectation limits and behavior • Monitor the resource consumption • Document the improvement in ROI, processing time, quality, etc

Deploy & Manage

This section contains guidelines for successful deployment and management.

Network deployment

As you now prepare for deployment of the automation process, it becomes key to establish a dependable and scalable network set up. A few possible and expected failures in an RPA implementation could be:

- Physical device failure – Azure will move the VM to a healthy host server automatically, but this self-healing migration could take several minutes, leading to failure in the RPA flow.
- VM maintenance – Upgrade and maintenance of VM introduces VM unavailability
- Gateway failure – VM failure also causes gateway connection to be interrupted

Below is a checklist that outlines the relevant activities widely to establish a network deployment plan:

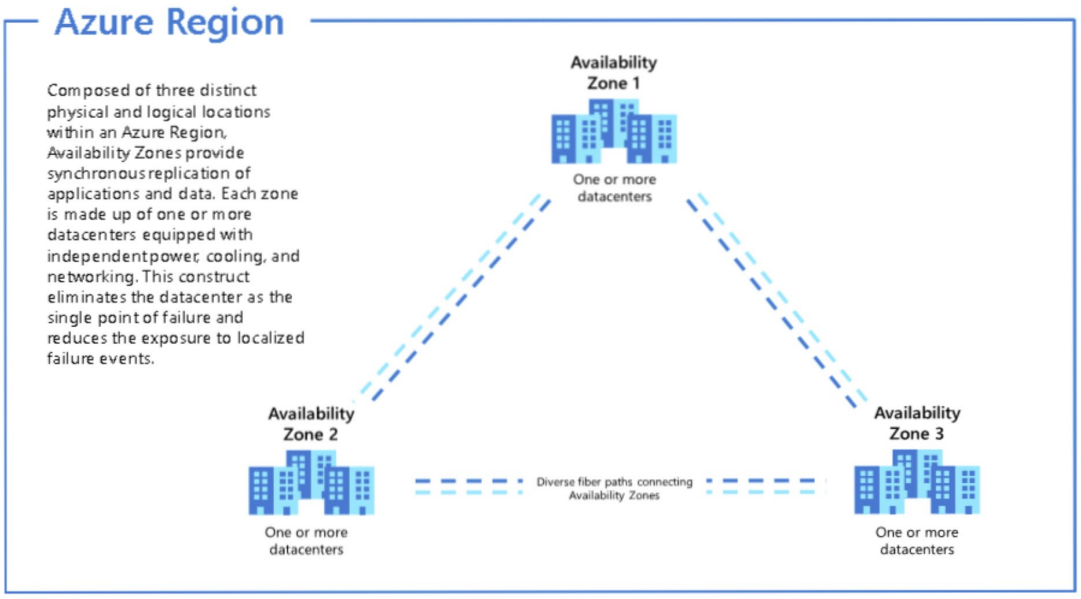
Checklist Item	Activities	Related Resources
High Availability	• Define target availability for resources and services • Ensure required capacity and services are available in targeted regions • Employ Azure availability zones • Plan for disaster recovery	Availability Azure Regions and Zones Azure disaster recovery
VM deployment plan	• Create VM in an availability zone • Utilize VM extension to automate post deployment configurations • Employ Azure automation to build and execute a comprehensive update management	Availability Zones VM in an availability zone VM Extension
Machine Groups	• Identify the machines based on the environment and process to be run • Define permission and access level for the machine group • Join 2 or more machines to form a group • Utilize the Machine Registration app to silently register machines and join to machine groups	Manage machine groups Silent registration of machines/machine groups

Azure high availability

To ensure your services aren't interrupted and avoid a single point of failure, it's recommended to deploy multiple instances on virtual machines into a regional Availability Zones.

An Availability Zone is a high availability offering that protects your virtual machines and data from datacenter failures. Availability Zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions. The physical separation of Availability Zones within a region protects applications and data from datacenter failures. Zone-redundant services replicate your applications and data

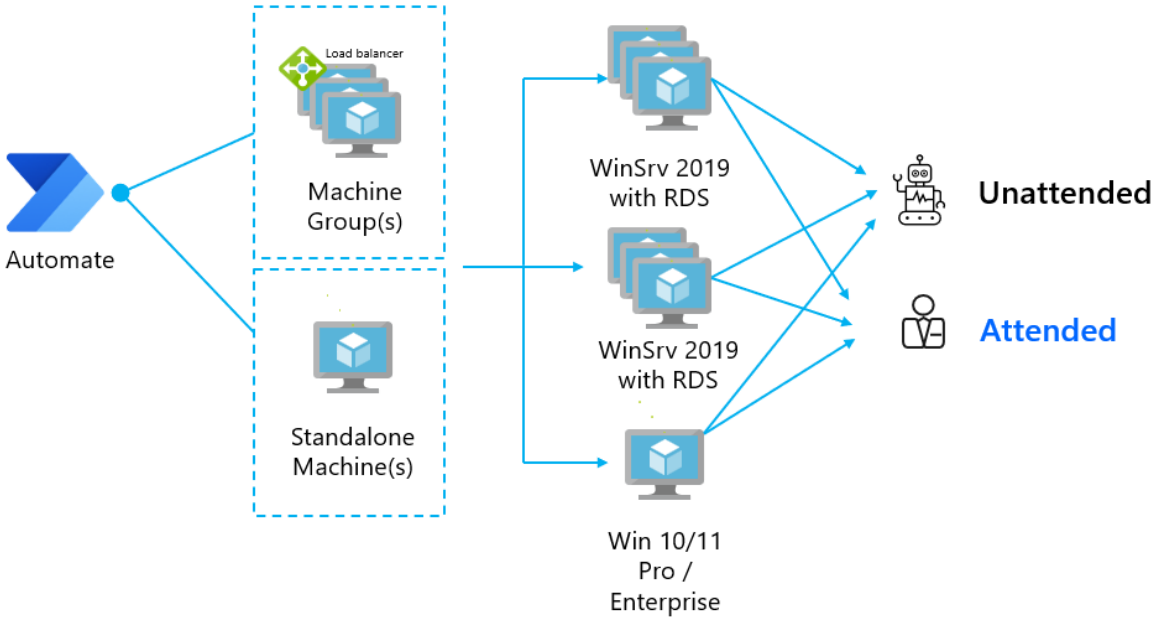
across Availability Zones to protect from single-points-of-failure. With Availability Zones, Azure offers a 99.99% VM uptime SLA.



Machines and machine groups

As you now have a designated machine to run the RPA automation, the machine should be registered to Power Automate to instantly start your desktop automation. When installing the desktop flow on a machine, you can also register the machine to the cloud flow to establish the connection automatically.

RPA can run either on standalone machines or load balanced via machine groups as shown in the figure below. By grouping the machines, you can benefit from automatic load balancing.



Application Lifecycle Management

Before you set up ALM for your RPA solutions, you should codify and automate the infrastructure needed for them.

Infrastructure deployment

Use infrastructure as code to create Azure infrastructure for your solution declaratively. Use platform tools like Azure Resource Manager (ARM) for this. ARM provides built-in features like orchestration, validation, and extensibility and lets you build your infrastructure repeatedly and consistently. You can reuse the templates across different environments and manage the environment-specific settings through parameters and variables. At very high level, these are the steps involved in automating your RPA solution infrastructure using ARM:

Checklist Item	Activities	Related Resources
Infrastructure Deployment plan	• Define resources (VMs, virtual networks, Azure Key Vaults) that you'd like to stand up or manage within Azure. • Deploy, manage, and monitor all the resources for your solution as a unit • Use version-controlled Templates. • Use Azure Desired State Configuration (DSC) to install and configure RPA related software.	ARM Templates Azure Deployment using ARM

Checklist Item	Activities	Related Resources
Configure Infrastructure for RPA solution	• Plan the Networking setup and security rules • Use Azure Desired State Configuration (DSC) to install and configure Power Platform software • Perform infrastructure verification and fitness for RPA	PowerShell DSC

Application deployment

Packaging and deploying solutions across various environments can be challenging. Distinct environments should be established development, testing, and for production, understand the best practices and guidelines to manage the solution lifecycle.

The following concepts are essential for understanding ALM using the Microsoft Power Platform.

- Solutions are the mechanism for distributing components across environments through export and import packages. A component represents something that you can potentially customize. Anything that can be included in a solution is a component, such as Power Automate flows. Solutions are of two kinds:
    - *Unmanaged* solutions are used in development environment and should be considered your source for all components.
    - *Managed* solutions are deployed as a "package" and are used in test/prod and does not allow you to edit any component
- Although solutions are stored in *Dataverse* stores. You should automate the process of exporting to a source control for collaboration and version control and backup.

Use ALM and version control tools to manage the development, testing and deployment of your application. You can use built-in constructs such as GitHub Power Platform Actions and Azure DevOps tasks for managing the application lifecycle. The actions can be used individually to perform a simple task, such as importing a solution into a downstream environment, or used together in a pipeline to orchestrate a scenario build, test, deploy, and verify your RPA solutions.

The following diagram shows the typical steps involved in deploying RPA solutions. Manual tasks are prone to errors, and often these steps are missed or incorrectly executed, leading to defects and inconsistency across environments. There should be an integration environment (dev) where the solutions can be verified before promoting to QA and Prod. The build pipelines can incorporate both Infrastructure and Application deployment. You can incorporate quality assurance techniques like code analysis as part of the build pipeline.

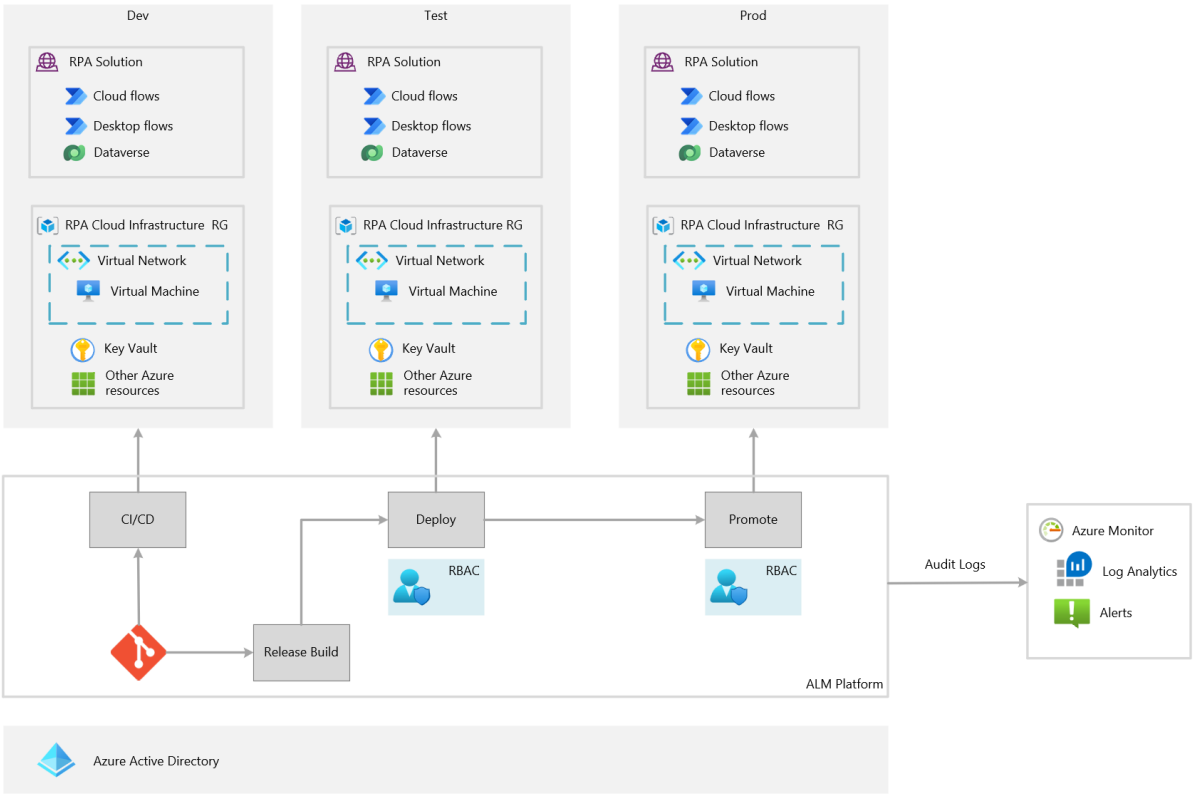


Figure 2: Sample build / release process

Follow the below guidelines to plan the deployment and continuous management of solutions:

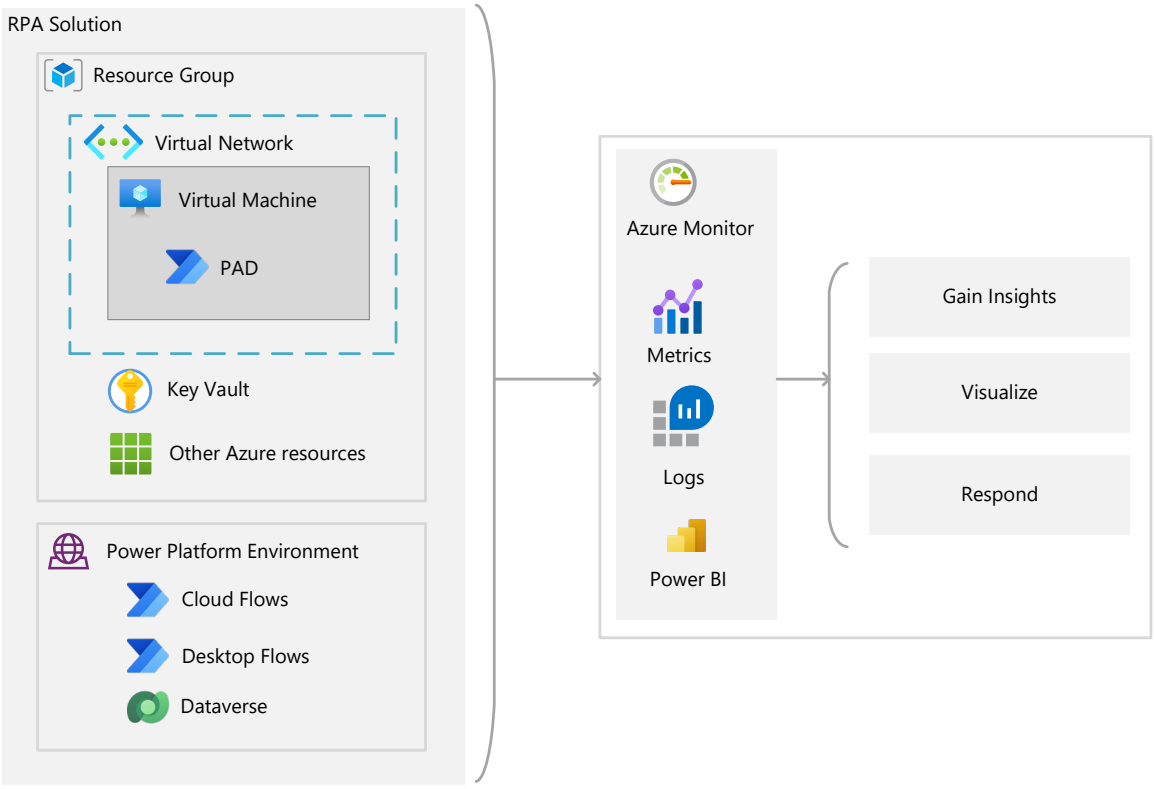
Checklist Item	**Activities**	**Related Resources**
Setup a Deployment plan	• Establish environment strategy for ALM • Create templates and scripts to manage environments	ALM Environment strategy Connect to Environments
Source control your deployment scripts	• Ensure solution templates and scripts are source controlled • Create a service connection with sufficient access to appropriate subscription(s) to execute deployment	ALM for Power Automate Solutions

Checklist Item	Activities	Related Resources
		Solution concepts
Release strategy & pipelines	• Design branching and release strategy with Git • Create deployment pipeline(s) • Add Azure CLI / PowerShell, etc. pipeline tasks to track • Test each pipeline, adjust tasks as needed • Setup pipeline triggers appropriate for each pipeline • Use Service Principal as a login to perform ALM tasks.	Continuous integration and deployment
Quality Assurance & Verification	• Use static analyzers for quality checks • Design a strategy for integration testing. • Test components in isolation even if the end-to-end test is not feasible	
Solution Lifecyle and Backup	• Distinguish between solution upgrade, update, and patch • Plan for the updating pre-requisites software • Determine backup and restore strategy for environments and application data in case of failed deployment	Solution Lifecyle Azure Automation Update Management Backup and restore environments
Security	• Use RBAC to control access to environments • Use available tools to audit and monitor changes to the source and configuration	Azure DevOps Auditing Azure DevOps Permissions
Use solution specific best practices	• Update connection references automatically • Visualize and review differences for text-based components	Connection References

Monitoring

With your RPA solution now deployed and online, its essential to monitor its components to ensure it is performing as expected. Use the log and metric and metric data collected from Power Platform and Azure to prepare a comprehensive solution for your monitoring use cases that includes:

- Develop dashboards to visualize the availability and performance of your solution
- Get proactively alerted when critical conditions are detected
- Trigger automation that will streamline the process of taking corrective action



This section describes the techniques your CoE can use to establish best practices around monitoring your enterprises scale RPA solution.

Insights

In Azure, Azure Monitor provides actionable insights to monitor workloads and resource utilization, get a unified view of operational health, and respond to system-level performance changes. Azure Monitor collects monitoring and operational data in the form of logs and metrics, providing you with a unified view of Azure resources, applications, and services that run on Azure and on-premises. Use Azure Monitor to detect anomalous behavior in your environments, set alerts, visualize logs and metrics, take automated actions, troubleshoot issues, and discover insights to keep your workloads and application functioning smoothly.

In the Analytics area of the Power Platform admin center, environment admins can view the overall status of Cloud and Desktop flows that have run within any environment over the past 28 days. Desktop flow run time data is stored in Dataverse within the following tables.

- Process - table that stores all workflow entities that contain information about all Desktop Flows built in Power Automate within a specific environment.
- Flow Sessions - table that stores Desktop Flow flowsession entities that contain information generated when a flow is run.
- Additional Context (additionalcontext) – is not a table or entity but a file attached to each flowsession entity. This file stores additional information related to the flow session, such as details about each action's execution. Using this information, you can get fine grained insights how the instance ran, which action failed, what was the failure reason, and what were the inputs and outputs for each action.

To centrally analyze, visualize, and be alerted to any availability and performance issues that occur across any aspect of your RPA solution, you must push your Cloud Flow and Desktop Flow run log information to Azure Synapse. Since the Desktop Flow information is stored in Dataverse tables, you could use a variety of techniques to retrieve the data, including:

- Dataverse Web API - use the Dataverse web API to access flow monitoring data from, for example, an Azure Data Factory pipeline or custom code for transferal to Log Analytics or another Azure data store.
- Dataverse row change flow trigger – use this Power Automate flow trigger to run a flow when a Dataverse table that contains Desktop Flow monitoring data changes.
- Azure Synapse Link for Dataverse – use this option to continuously export flow monitoring data from Dataverse to Azure Data Lake Storage Gen2 or Azure Synapse Analytics. As of this writing, this method does not transfer data from the Flow Sessions table.
- Log Analytics Data Collector API – use this option to surface information as logs in Log Analytics.

Use the following checklist to access monitoring data across Azure and Power Platform to build an enterprise scale RPA CoE monitoring solution.

Checklist Item	Activities	Related Resources
Prepare Azure Monitor for data collection	• Deploy a log analytics workspace.	Create a Log Analytics workspace in the Azure portal
Monitor your PAD machine group	• Add VM Insights solution to your Log Analytics workspace	Configure Log Analytics workspace for VM insights

Checklist Item	Activities	Related Resources
	• Install VM Insights agent on all virtual machines or on-premises servers you wish to monitor • View the health and metrics for your network in Azure Monitor VM Insights	Enable Azure Monitor for virtual machines What is VM insights?
Monitor your PAD machine group network	• Configure connectivity tests using Connection Monitor • Log network traffic using NSG log flows • View the health and metrics for your network in Azure Monitor Network Insights	Connection Monitor in Azure Log network traffic flow to and from a VM Azure Monitor Network Insights
Monitor other Azure resources in your RPA solution	• Configure Azure Monitor Key Vault insights • Configure remaining services to capture logs	Monitor Azure Key Vault with Key Vault insights Azure Monitor Logs
Monitor Cloud / Desktop Flow runs	• Create a Power Automate Cloud Flow to push flow run logs to Log Analytics when a row in the Flow Sessions table is added • Determine the data table and sync with Azure data lake store using Synapse link	Azure Log Analytics Data Collector Connector Azure Synapse link Azure synapse link pre-requisites Azure Synapse workspace
Custom logging and monitoring	• Plan for custom logging using pre-defined templates to capture additional events on key actions • Evaluate custom dashboard needs	

Visualize

In Azure, you have the ability to visualize log and metric data in a number of ways, including interactive workbooks and Azure dashboards. In the Power Platform, admins at varied levels have access to rich flow health and usage visualizations within the Flow Analytics and the Power Platform admin center mentioned in the previous section.

You can also Power BI directly to both Azure and Power Platform monitoring datasets to create dashboards. Great examples of this approach can be found within the Automation CoE Starter Kit, the RPA run log analytics Power BI dashboard, and the Azure Continuous Cloud Optimization Power BI dashboards.

Since you have pushed your Desktop Flow log data into Azure data factory in the previous section, you can leverage any of the Azure Monitor techniques to visualize any of your RPA solution components on a single pane of glass. Use the following checklist to prepare a Power BI dashboard your team can use to view the health of your RPA solution.

Checklist Item	Activities	Related Resources
Flow Analytics	• Get insights into runs, usage, errors, connectors on Cloud and desktop flows	Flow Analytics
Visualize your RPA solutions health and availability	• Import the log and metric data collected in Log Analytics • Connect Azure data factory to Synapse analytics to visualize the health and • Develop Power BI reports and dashboards to visualize the health and availability of your RPA solution	Analyze data by using Power BI Create reports and dashboards in Power BI

Respond

Alerts in Azure can be configured to fire whenever some critical condition has been met that you or your team should be notified of. When a signal, such as a metric or log, is emitted from a target resource and meets certain criteria, such as log count > 3 or CPU utilization > 75%, then an action you specify will fire, such as sending an email to the CoE team inbox.

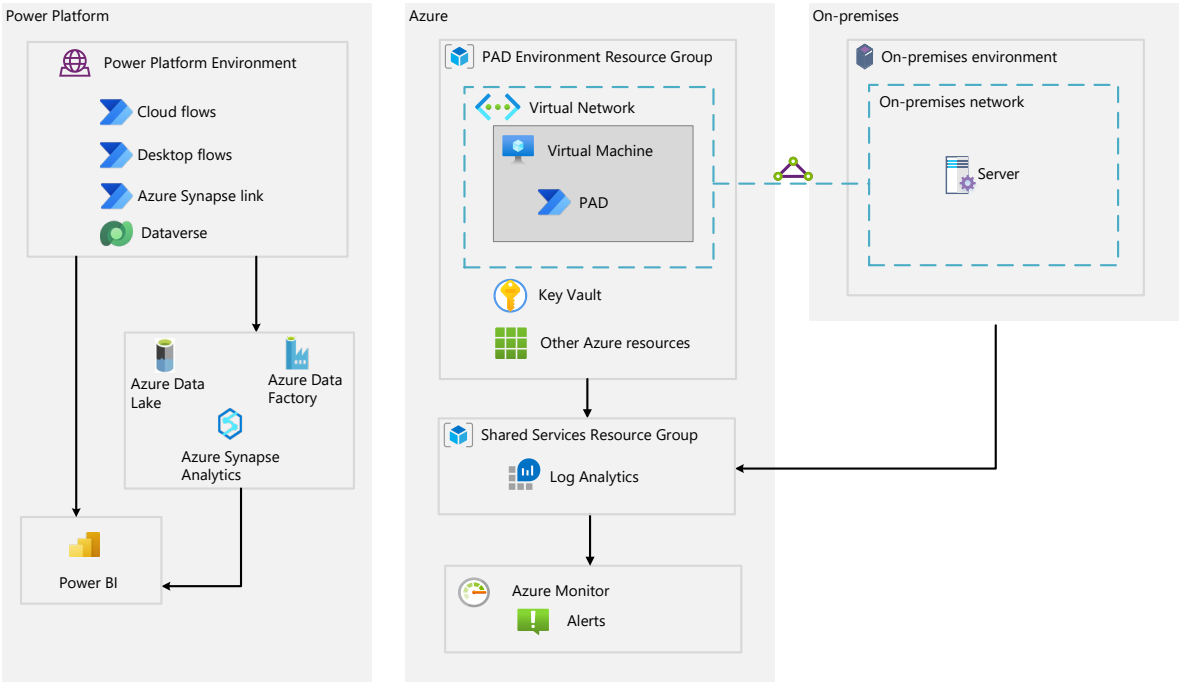
Normally Azure Alerts only consider Azure resources, but since you are pushing Desktop Flow session logs into Log Analytics, you are able to use log alerts to centrally respond to a critical condition related to any component of your RPA solution, regardless of whether it resides in Azure or Power Platform. The checklist below describes the actions you must take to configure this centralized alerting capability. Consider extending this checklist to include more complex response workflows to meet your organizations specific needs, such as sending a message to a Teams channel or creating a work item on your CoE backlog to respond to the alert.

Checklist Item	Activities	Related Resources
Configure alerts on critical conditions your team should be aware of	• Work with your stakeholders to establish what conditions should alert the team • Use log alerts to be notified of issues detected in the Desktop Flows logs	Cloud monitoring and alerting Create an alert rule

Checklist Item	Activities	Related Resources
	• Create Azure Alerts for each condition	Log alerts in Azure Monitor

Solution

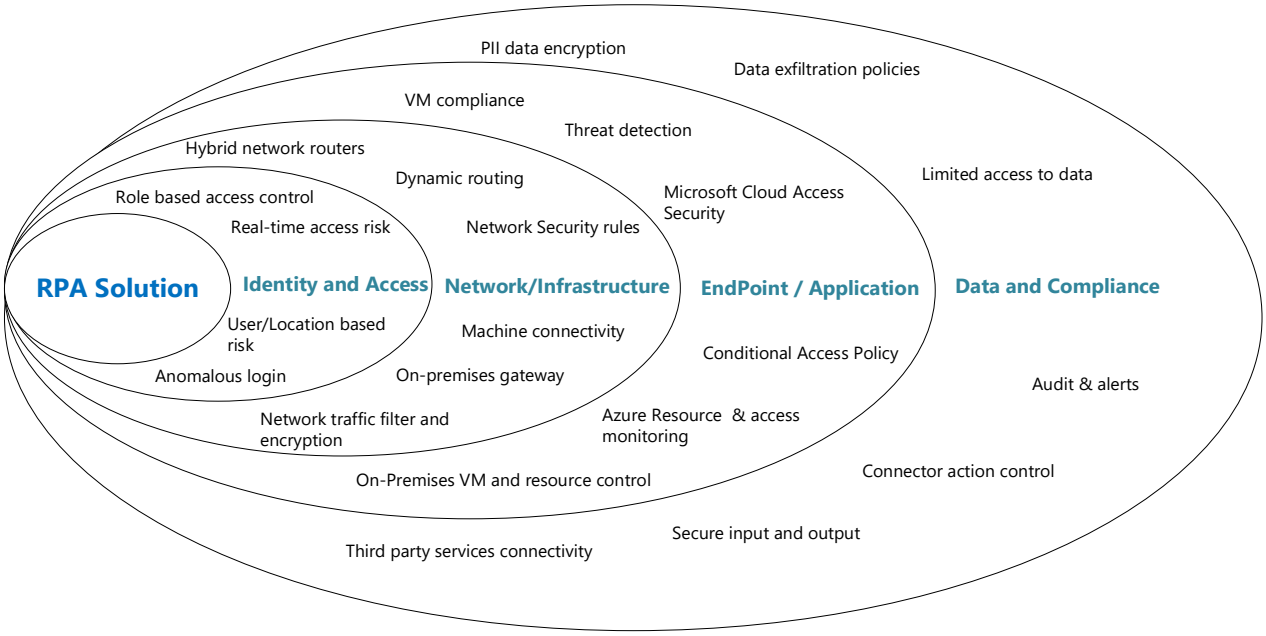
Having completed all the activities in the monitoring checklists above, you now have a monitoring solution that equips your RPA CoE with the ability to gain insights into and visualize an RPA solution from a single pane of glass. The diagram provides a complete picture of that solution.



Secure & Govern

With the corporate data footprints that have expanded outside of your network, you should focus on designing a model that effectively adapts to the ever-changing data landscape. Microsoft's Zero Trust security model provides a comprehensive control plan, helping you to protect identities, data, applications, and devices across on-premises, cloud, and mobile.

Zero Trust security model by Microsoft offers defense in depth layering security to reduce security breach threats. The architecture depicted below elaborates on a model that offers 6 layers of security – Identity, Endpoints, Data, Applications Infrastructure and Network



Identity and access

All Azure resources are governed and controlled through Azure Active Directory, which is the central identity plane. Microsoft Azure Active Directory (AAD) has been recognized as a leader in Gartner Magic Quadrant for Access Management, worldwide.

Key considerations as you model your identity management:

- Azure Active Directory – Identity and access management as a service that authenticates users, devices and processes to your resources, applications and other services used in RPA solution
- External Identities – Authenticates people outside your organization to access your apps and resources with either as:
    - Azure business-to-business (B2B) account - guest to your tenant
    - Azure business-to-customer (B2C) account – identify using an already established external identity
- Conditional Access – Analyzes signals such as user, device, and location to automate decisions and enforce access policies for resources.
- Role Based Access Control (RBAC) – Ensures only authorized and approved users have appropriate access to all services and azure resources comprising RPA. An RBAC role assignment can be defined using these three elements.
    - Security principal: A user or a group or a service principal or a managed identity that is requesting access.
    - Role definition: A role definition is a collection of permissions. It specifies the management operations that the role allows to be performed on Azure resources.
    - Scope: Scope is the set of resources that the access applies to.

- AAD Service Principals - The concept of a "service account" in AAD is often achieved through an App Registration and "service principal." You can think of the service principal object as an instance of the app registration, with specific access policies and permissions for an assigned resource

Design an RBAC strategy that grants users and service accounts access to the scopes they require to do their job. By using Azure Active Directory, you can manage this access centrally across all platforms.

The following tables show a real-world RBAC strategy example. To start, you should always formulate a predictable group naming convention. In this example, that convention is [ORG_NAME]-[BUSINESS_UNIT]-[APP_NAME]-[ENVIRONMENT]-[RESPONSIBILITY].

Environment based access:

In the last 6 rows, you can see that the example includes a common use case of having different groups for the different security or environment boundaries you define for your application. To keep it simple, the example show only production (example: CONTOSO-COE-APP1-PRD-ADMINS) and non-production (CONTOSO-COE-APP1-NONPRD-ADMINS). In your environment, you may need to expand non-prod further to cover other boundaries, such as a staging environment (example: CONTOSO-COE-APP1-STAGING-ADMINS).

AAD Security Group Name	Group members	Power Platform/ Dataverse		Azure Roles		Microsoft 365 Roles
		Scopes	Role	Scopes	Role	Role
CONTOSO-GLOBALADMIN	2 - 4 users entrusted with complete control of everything under the AAD tenant.	AAD	Owner	AAD	Owner	Owner
CONTOSO-COE-HEAD	COE Head	Environment	Basic User - extended to custom entities	Subscription	Reader	Compliance Reader
						Security Reader
						Reports Reader
CONTOSO-COE-SECURITY	CISO Security Team	Environment	Basic User - extended to custom entities	Security Center	Security Admin	Compliance Admin
				Subscription	Contributor	Intune Service Admin
						Endpoint Security Manager
CONTOSO-COE-GLOBALREADER	CIO	Environment	Basic User - extended to custom entities	Subscription	Global Reader	eDiscovery Manager
						Security Reader
						Compliance Reader
						User Admin

CONTOSO-COE-ONBOARDING	User onboarding team					Reports Reader
						Billing Admin
CONTOSO-COE-APP1-PRD-STAKEHOLDERS	Business Process Owners of App 1	Environment	Reader			View-Only Audit Log
		Workspace	Power BI Contributor			Reports Reader
CONTOSO-COE-SYSOPS-PRD-ADMINS	System Operations Service Admin accounts of App 1			App 1 Resource Group(s)	Contributor	
CONTOSO-COE-PPOPS-PRD-ADMINS	Power Platform Service Admin accounts of App 1	Environment	Environment Admin			
CONTOSO-COE-APP1-PRD-DEVELOPERS	COE Developers Business Process Citizen Developer(s) for App 1	Environment	Environment Maker	App 1 Resource Group(s)	Reader	
CONTOSO-COE-POWERBI-PRD-ADMINS	Power BI admin	Workspace	Power BI Member			
CONTOSO-COE-APP1-PRD-USERS	Application users	Environment	App User			
CONTOSO-COE-APP1-NONPRD-STAKEHOLDERS	Business Process Owners of App 1	Environment	Basic User - extended to custom entities			View-Only Audit Log

						Reports Reader
CONTOSO-COE-SYSOPS-NONPRD-ADMINS	System operations admin			App 1 Resource Group(s)	Contributor	
CONTOSO-COE-PPOPS-NONPRD-ADMINS	Power Platform admin	Environment	Environment Admin			
CONTOSO-COE-APP1-NONPRD-DEVELOPERS	COE Developers Business Process Citizen Developer(s) for App 1	Environment	Environment Maker	App 1 Resource Group(s)	Contributor	
CONTOSO-COE-APP1-NONPRD-USERS	Test team Early access application users	Environment	App User			

Below is a checklist that outlines widely the relevant activities to establish Identity management deployment plan:

Checklist Item	Activities	Related Resources
Identity Management	• Synchronize your on-premises directory with your cloud directory using Azure AD • Use Single sign-on to enable users to access all SaaS applications • Enable on-premises integration with Azure AD password protection	Hybrid identity with AAD Single Sign-on Identity Protection Password management
Define administrator and other CoE roles (refer above for a use case)	• Assign at-least two global administrator accounts for use in case of emergency • Define other key administrator roles and limit access to only the areas needed • Not all administrators need be global administrators • Enable Privileged Identity Management to start tracking administrative role usage.	Security emergency access Built-in roles Privileged Identity Management
Conditional access	• Block legacy authentication to Azure AD with conditional and deploy multi-factor authentication (MFA) • Create policy to require MFA for administrators excluding service accounts and service principals • Review and deploy common policies and security defaults as it suits your organization needs • Define custom policies as per your organizations requirements	Block legacy authentication Multi-Factor Authentication Security defaults Common policies
Role Based Access Control	• Plan role definitions and scopes for each environment / subscription • Use built-in roles and create custom only when needed • Define the necessary service principal to assign the roles • Define Security groups to manage the automation flow in Power Platform • Define roles to be assigned to the security group and teams	RBAC overview Azure built-in roles Create custom roles Role based assignment steps Power Platform Security roles

Network infrastructure

All data in your organization is ultimately accessed over network infrastructure. Networking controls can provide critical controls to enhance visibility and help prevent attackers from moving laterally across the network.

Checklist Item	Activities	Related Resources
Network segmentation	• Create dedicated virtual networks for different applications/services • Create a central virtual network to set up security posture and connect virtual networks in a hub-spoke architecture • Deploy Azure Firewall to inspect and govern traffic between the virtual networks	Virtual networks Hub spoke architecture
Firewall protection	• Define Network security groups to allow/deny traffic from several azure resources • Protect Http/s endpoints • Enable threat intelligence-based filtering	Azure web application firewall Threat intelligence-based filter
Encrypt data at rest and in transit	• Enforce HTTPS only communication for internet facing applications • Turn on encryption for any point-to-site traffic • Encrypt connection to virtual machines using RDP	Redirect traffic to HTTPS Azure VPN Gateway Site-to-site VPN
Network security rules	• Control traffic to and from all RPA resources - azure resources, database, power automate, cloud app • Configure your firewalls to grant your RPA infrastructure connectivity to the on-premises or Azure hosted applications they need to interact with.	Network Security groups Application security groups Power Automate IP address configuration
On-Premises Gateway	• Enforce clustering and manage gateway loads	Gateway cluster Machine groups

Endpoint / Application

With the identity model defined, the next level of protection is to manage the devices in the cloud, on-premises, and hybrid environment. Microsoft Endpoint Manager is a solution platform

that unifies the services and tools to manage and monitor mobile device, RPA access from a mobile device, RPA desktop computers, RPA virtual machines, and servers. Following are some of the key features which can be used for end-point management.

- Microsoft Intune – Mobile device management (MDM) and Mobile application management (MAM) to check compliance and deploy apps, settings to your devices using the cloud
- Configuration management – Plan to automate app deployment, software updates and operation systems on the on-premises devices
- Microsoft Defender – Key component in protecting your endpoint against treat and apply appropriate remediation

Checklist Item	**Activities**	**Related Resources**
Manage endpoints	• Restrict access to your cloud-apps from Intune-managed / domain-managed devices • Define device / location based conditional access policies • Configure automated notifications emails to help manage risk detection	Location based restriction Require Compliant devices Configure notifications
Manage machines and configuration	• Determine your objectives for device management – type of apps, allow personal devices, etc. • Register devices with your central identity provider. Ex: Azure active directory • Review existing policies and infrastructure • Define, monitor, and plan remediation actions for compliant rules to access organization resources • Determine authentication model for the devices to connect to your organization • For on-premises endpoints, user Configuration manager to protect the endpoints • Create and deploy package to manage Power automate Desktop on VMs	Device management Integrate with Security services Using device profiles Manage auto-deployment Device status Configuration manager
Protect and track device security	• Automate plan to deploy updates across your organization's devices • Set up dedicated cloud instance of defender for Endpoint • Determine security administrators and operators to use the Microsoft 365 defender portal	Defender for endpoint Defender portal

Checklist Item	Activities	Related Resources
	• Configure proxy app and internet settings to enable communication between your devices and defender for endpoint	
Data Loss Prevention Policies	• Identify sensitive information stored and provide remediation – remove external sharing permission, encrypt file, etc. • Restrict connectivity to external apps, based on the exposed threat level • Enforce DLP policies to assign labels to information being transported	Protective actions of DLP Policies
Conditional Access policies	• Define policies to allow or block access to cloud apps based on user/group, device, location • Define policies in Azure for Power Automate to grant or block based on user/group, device, location • Set mobile application protection policies for Power Automate apps on Android and iOS	Cloud app access policies Power Automate Conditional access policies
App discovery and restrictions	• Establish an app discovery process to identify new app connections • Establish integrations with the app to detect threats and anomalies detected	Microsoft Cloud App security management Anomaly detection policy User activity policy Cloud App Security

Data and Compliance

The core of this section is to protect data while at rest, in use and when it leaves the endpoints, app, infrastructure and network. Three core elements of data protection are:

- **Know your data** - Discover data across your entire organization and classify all data by sensitivity level
- **Protect your data and prevent data loss** - Sensitive data needs to be protected by data protection policies that label and encrypt data or block over-sharing. This ensures only authorized users access the data, even when data travels outside of your corporate environment.

- **Monitor and remediate** - You should continuously monitor sensitive data to detect policy violations and risky user behavior. This allows you to take appropriate action, such as revoking access, blocking users, and refining your protection policies.

Checklist Item	Activities	Related Resources
Organization Data Policy	• Define information protection policy for your organization • Auto classify data with sensitive labels	Manage Information policy Getting started with sensitive data
Auditing and Compliance	• Set up auditing to manage the Power Automate for desktop app • Enable auditing to monitor activities events and life cycle events of Power Automate	Audit and Compliance for PAD app Power Automate data privacy Compliance
Anomaly detection	• Define network access policies for Power Platform environments blocking connectors, application gateways as necessary • Establish policies to block download, copy or print sensitive information from unmanaged devices • Set up in-app permission to restrict access to the data	Microsoft 365 Data loss prevention policies

Manage sensitive data in cloud flow	• Determine the data need be protected like –
	  • Secrets or Password to connect to an application
	  • Input variables to desktop flow
	• Identify the right authentication model to the applications/systems involved
	• Secure the passwords / access token in Azure Key Vault and apply necessary access policy
	• Turn on Secure Input / Output to protect exposure of this information in run history
		Power Automate authentication security
		
		Securing Inputs and Outputs

Secure credential storage and rotation	• Secure the supporting service credentials, passwords, keys in Azure Key Vault and apply necessary access policy • Implement a consistent secret rotation process for all Azure services	Secure secrets in Power Automate with Azure Key Vault

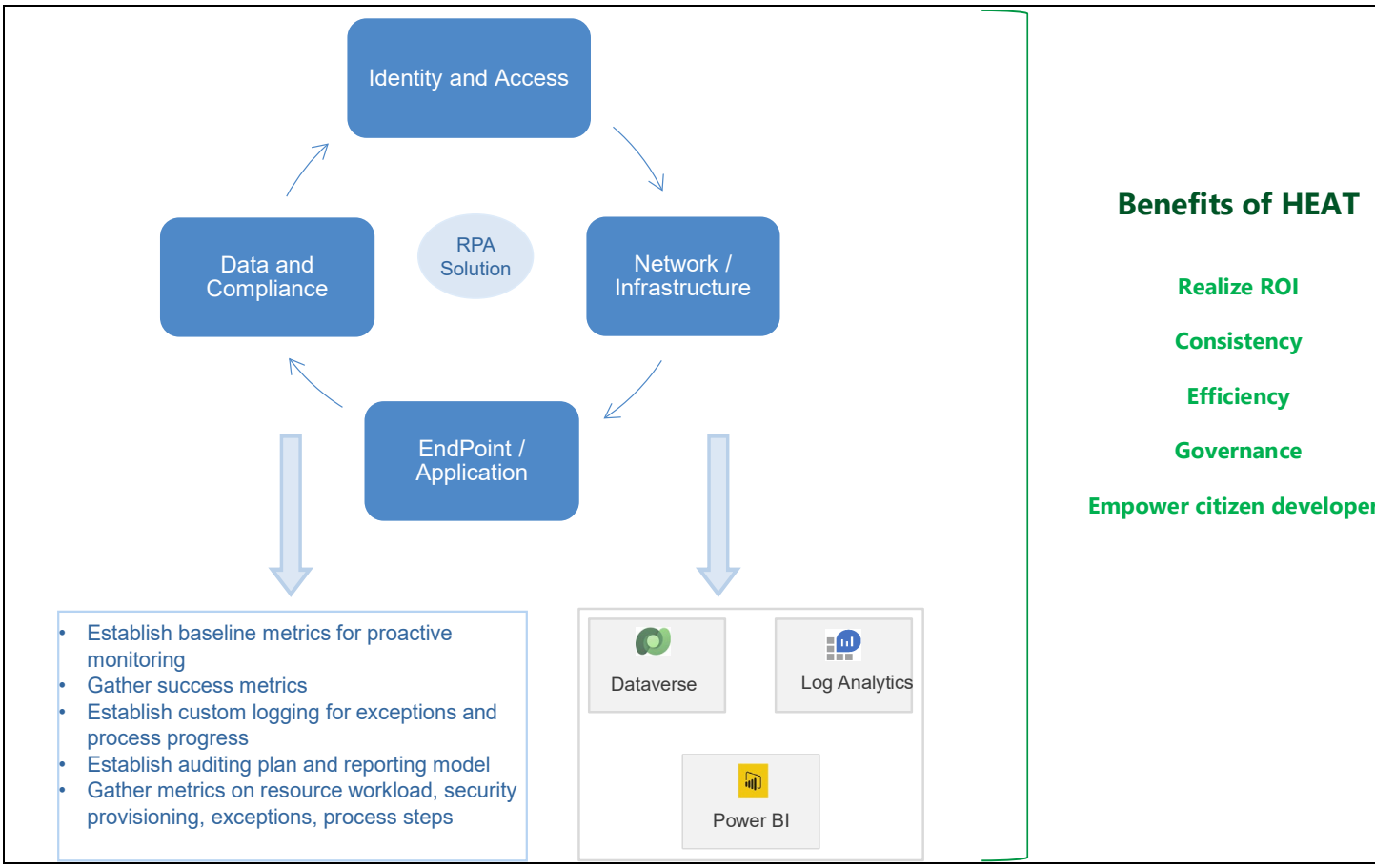
Checklist Item	Activities	Related Resources
		Automate Secret Rotation
Manage sensitive data in desktop flow	• Determine sensitive information used in the automation as input • Encrypt the data by using input as sensitive text type	Use sensitive text Secure data in Desktop flow
Manage data source access	• Determine sensitive information stored in the Dataverse tables regards to the automated flow run history • Establish a security plan to define security roles at business unit, table, field and/or record level • Determine all the data sources used as part of the RPA solution and establish a security plan to set access for non-administrator users	Dataverse Security concepts Dataverse Record Level security Azure SQL non-administrator user access

Proactive and reactive monitoring

To improve the overall operational productivity of the RPA solution, proactive monitoring is a key and allows to act and remediate before the identified problem arises.

As discussed in the previous sections, to proactively monitor an RPA solution, key metrics to be captured

1. Establish baseline - Understand and gauge the current as-is process and establish a healthy baseline
2. Gather success metrics - Understand the business needs – in terms of new / unique users and / or visitors, duration taken for a process to complete, navigation patterns
3. Define Security deviations - Evaluate and define security deviation based on the baseline



With the use of varied set of services and tools to secure and govern the RPA solution components, it is best practice to unify the security data and analytics into an elected Security Information and Event Management (SIEM) tool to get insights across the enterprise.

You also have the flexibility to use any SIEM provider that you have employed, publish data to Azure Data factory, configure Power BI to create and monitor dashboards.

Below checklist highlights the activities and permissions required to ingest data into SIEM tool:

Checklist	**Activities**	**Permission required**
Prepare Azure subscription	• Determine the Azure subscription to be monitored • Ensure necessary policies are enabled at all levels as detailed in the previous section	Subscription owner
Connect Azure subscription to security services/tools	• Create Managed identity and assign privileged (Owner) access to the subscription	Managed identity to have owner access to subscription Security admin

Checklist	Activities	Permission required
	• Establish connection between the services / tool and subscription using the managed identity • Subscription to Security Center • Azure resource manager service connection • Log analytics • Conditional Access Policies • Microsoft 365 Data Loss Prevention policies • Power Platform Data Loss Prevention Policies	Compliance admin Power Platform Environment admin
Ingest Dataverse data for proactive monitoring	• Determine the custom data tables to be exported • Export data using Azure Synapse link for Dataverse • Connect Dataverse data tables to Services / tools • Create data factory under the same subscription • Transform your Dataverse data	Subscription owner Dataverse environment - System Administrator
Security Information Event Management (SIEM)	• Identify a SIEM provider to deliver intelligent security analytics and threat intelligence. Ex: Azure Sentinel • Connect the RPA data sources to SIEM to corelate alerts • Integrate threat intelligence solution • Create custom rules for alerts	Azure sentinel contributor

Nurture

Low code development is the core of RPA automation and promotes involvement of citizen. Citizen developers have a key role either in the success of automation platform by either developing/migrating the business process or to work with the technical team as business process experts.

Nurture is an essential part of the Center of Excellence (CoE) that focuses on to continue the growth and onboarding of makers and moving your organization to embrace a digital culture.

Nurture models

Below are few ideas to consider:

Evangelism	Community Development	Training and Support

Run internal RPA in a day workshops	Create an internal community on Teams for your champions	Provide internal learning resources and tracks for beginner, intermediate and advanced makers
Organize bot wars with real business scenarios	User a Teams or SharePoint site to store resources like your own best practices or brand guides	Share your organizations best practices and templates
Share success stories	Update your makers about new features in the RPA platform with a monthly email newsletter	Organize hands on labs session to master the technical knowledge Admin in a day Azure DevOps Build tool
Hold Show & Tell sessions to learn what other makers are creating	Offer individual recognition and career paths	Microsoft Power Automate teams blog frequently on both latest updates as well as ongoing examples of using the features of the products. RPA team also released a series of articles and videos on the Holistic Enterprise Automation Technique for RPA.

Nurture team roles

First steps to organize and successfully run the nurture program in your organization, will be to have the right team working together. From our definition of separate roles in CoE, below would be good team to work on the nurture phase:

Automation Product Champions – A team of architects, pro-developers and citizen developers will contribute to develop mentoring and coaching programs and lead the effort in successfully delivering the program.

Automation Makers – A team of pro-developers and citizen developers will share best practices, templates, success stories and provide technical support to the new onboarding makers

Automation CoE team – A team of pro-developers, leadership will contribute to the business goals for the automation program by collaborating with business leaders.

Real-world implementation example

The following diagram depicts an example of an architecture that combines Power Automate RPA and Azure resources. Here are the critical components of the diagram:

- **Workload and Endpoint -** (**depicted as "1" in the diagram**) – This subscription includes several Azure services joined to the Azure virtual network, such as the Virtual Machines that host unattended RPA instances. All resources inside the virtual network can communicate securely without being exposed to the public Internet. Resources like the VM inside the VIRTUAL NETWORK can be segmented using subnets for additional security.
- **Infrastructure Services** - The workload subscription connects to infrastructure services (**depicted as "2" in the diagram**) such as identity and access management (Azure AD) and security services (Azure Key Vault).
- **Power Platform Services** - The workload can connect to Power Platform-based services such as Dataverse **(depicted as "3" in the diagram**) via the Power Platform VIRTUAL NETWORK Service (preview) or directly over the Internet or use the Azure Express Route service as shown in the diagram below
- **On-premises network** - The workload subscription can connect to the on-premises corporate network (**depicted as "4" in the diagram**) via Azure Express Route.