**Final Report: Implementation of IAM and PAM Strategy to Secure Access in Active Directory Environment**

**1. Introduction**

**1.1 Overview of the Project**

In the modern cyber threat landscape, the management and security of identities and privileged accounts are critical. Active Directory (AD), as the core directory service in many enterprises, often becomes a target for malicious actors due to its central role in managing access to resources across the network. Attacks like **pass-the-hash**, **pass-the-ticket**, and exploitation of vulnerabilities in AD services can severely compromise security.

The aim of this project is to design and implement an **Identity and Access Management (IAM)** and **Privileged Access Management (PAM)** strategy that ensures the security of users and privileged accounts in an Active Directory environment (spanning versions 2016, 2019, and 2022). By securing access controls, automating identity lifecycle management, and monitoring access patterns, this strategy seeks to mitigate common attack vectors and improve overall enterprise security.

**1.2 Project Objectives**

1. **Analyze Risks Related to Identities and Privileged Access in Active Directory**:
   * Identify prevalent attacks that target Active Directory, such as **Kerberoasting**, **DCSync**, **Golden Ticket**, and more.
   * Explore vulnerabilities that arise from misconfigurations and insecure setups in Active Directory environments.
2. **Design and Implement IAM Solutions**:
   * Integrate **Multi-Factor Authentication (MFA)** for enhanced access security.
   * Implement **Single Sign-On (SSO)** to simplify user authentication while maintaining security.
   * Establish a comprehensive approach to identity lifecycle management for onboarding, updating, and offboarding users.
3. **Design and Implement PAM Solutions**:
   * Isolate and protect privileged accounts to reduce the attack surface.
   * Implement automatic password rotation and secure password vaulting for sensitive accounts.
   * Use **bastion hosts** and restricted administrative workstations to prevent direct administrative access.
4. **Monitor and Respond to Access Anomalies**:
   * Utilize logging and anomaly detection tools to identify suspicious activities.
   * Integrate the system with **Security Information and Event Management (SIEM)** tools for real-time alerts and monitoring.
   * Automate incident response using scripting languages like **PowerShell** and **Python**.

**2. Methodology**

**2.1 Research Approach**

This research is structured into four main phases:

1. **Risk Analysis**: Understanding and identifying the potential threats to identities and privileged accounts in Active Directory.
2. **IAM/PAM Strategy Design**: Designing robust solutions for identity and privileged account management.
3. **Tool Implementation**: Implementing the selected tools and technologies for IAM and PAM solutions.
4. **Automation and Monitoring**: Developing scripts and integrating monitoring tools to enhance security and provide real-time detection and response.

**2.2 Tools and Technologies Used**

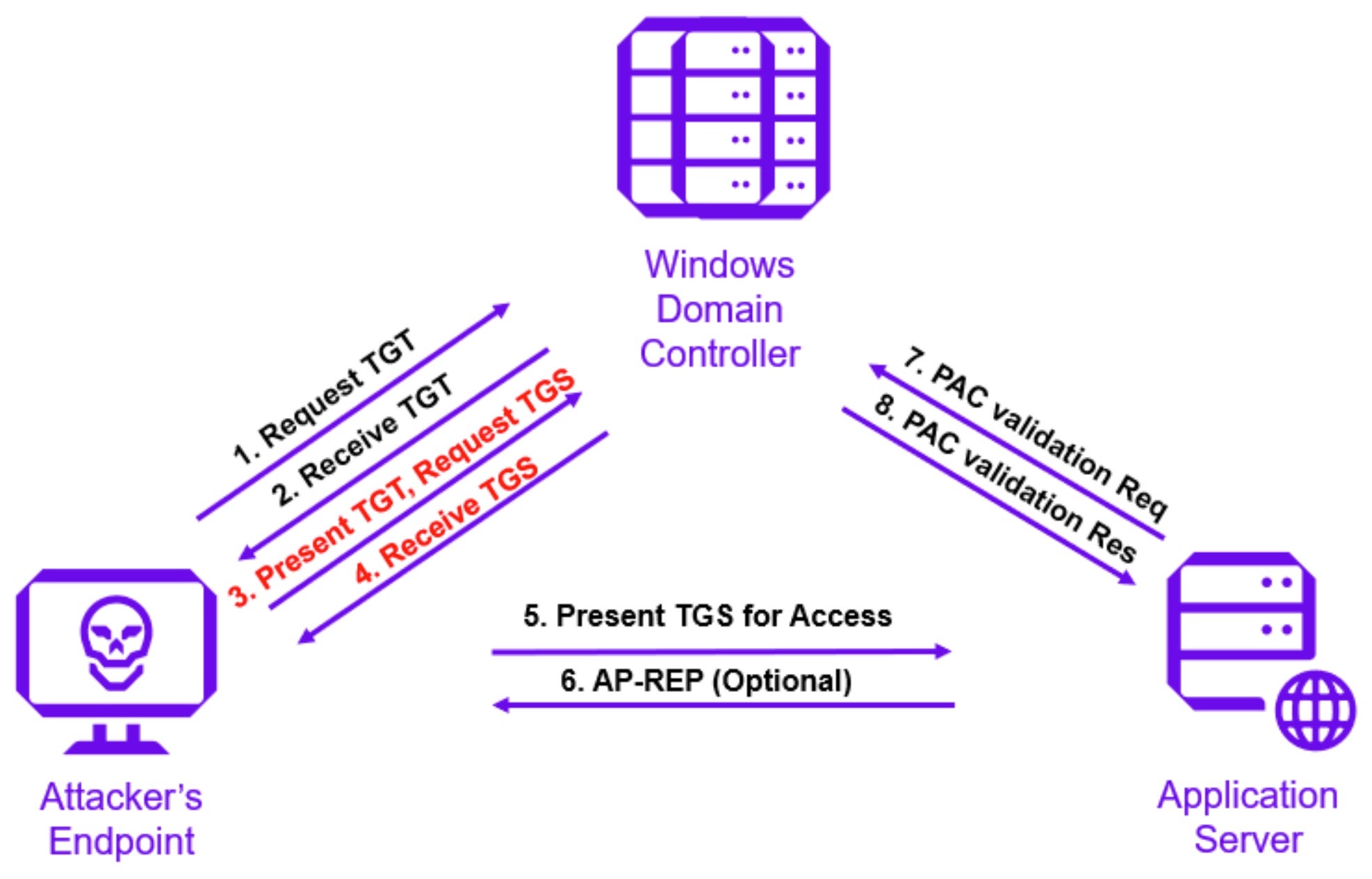
* **IAM Solutions**: Microsoft Azure AD, Keycloak, Okta, Ping Identity.
* **PAM Solutions**: CyberArk, BeyondTrust, HashiCorp Vault.
* **SIEM Solutions**: Splunk, ELK Stack, Wazuh.
* **Automation**: PowerShell, Python (LDAP3, PyAD libraries).
* **Monitoring**: Wazuh, Microsoft Sentinel, Splunk.

**3. Common Active Directory Attacks and Vulnerabilities**

**3.1 Kerberoasting Attack**

**Kerberoasting** is a technique where an attacker requests service tickets for service accounts in the network using the Kerberos protocol. These service tickets are then cracked offline to retrieve plaintext passwords of service accounts.

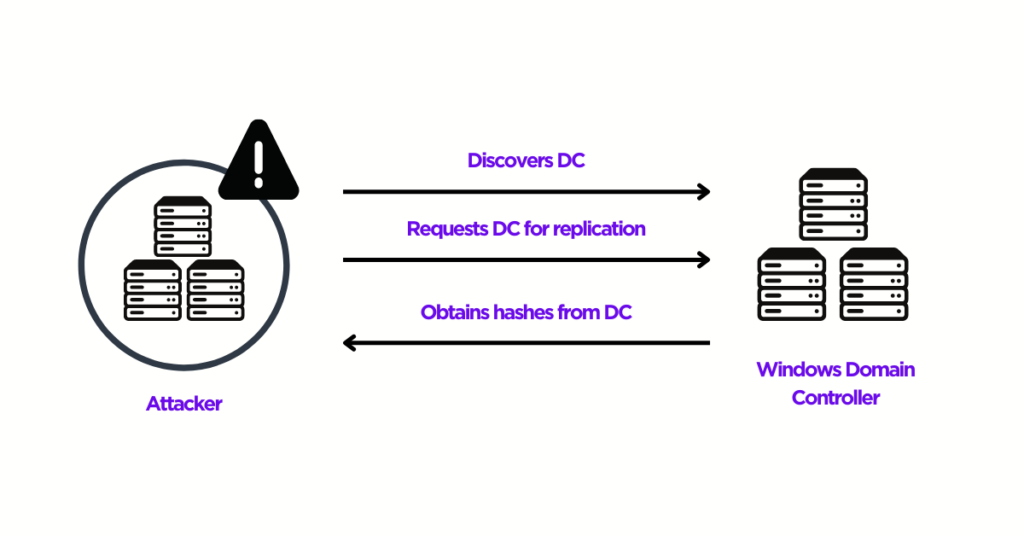
* **Definition**: The attacker takes advantage of the fact that service accounts often have weak passwords and requests **service tickets** from the Key Distribution Center (KDC). The service ticket is then offline-cracked to obtain the service account’s plaintext password.
* **Prevention**:
  + Enforce strong, complex passwords for service accounts.
  + Use **Managed Service Accounts (MSAs)**, which are automatically managed and rotated by AD.
  + Ensure service accounts use **AES encryption** for Kerberos tickets instead of weaker encryption types.



**3.2 DCSync Attack**

**DCSync** is a method used by attackers to simulate the behavior of a domain controller to retrieve Active Directory replication data, specifically **password hashes** for high-privilege accounts, such as domain administrators.

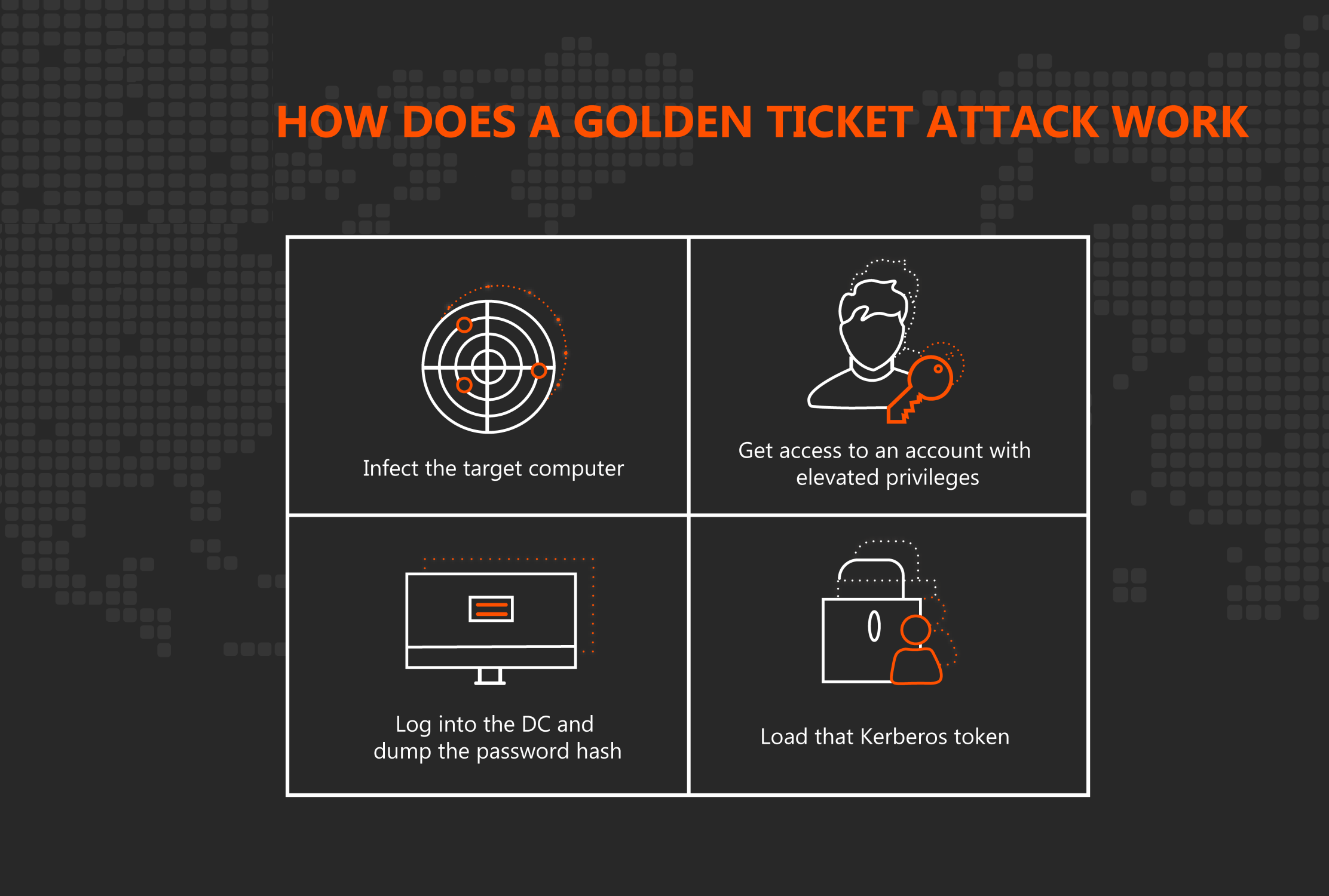
* **Definition**: The attacker impersonates a domain controller, requesting the **NTLM hash** of passwords for all domain users or specific high-privilege accounts, thereby compromising domain administrator credentials.
* **Prevention**:
  + Limit **replication permissions** to prevent unauthorized domain controllers from requesting replication data.
  + Regularly change the **KRBTGT account** password to reduce the effectiveness of forged Kerberos tickets.
  + Use **Credential Guard** on endpoints to prevent credential theft.



**3.3 Golden Ticket Attack**

A **Golden Ticket** is an authentication ticket forged by an attacker using the **Kerberos Ticket Granting Ticket (TGT)** to gain unauthorized access to all services within the domain.

* **Definition**: In this attack, the attacker forges a **TGT** (using the KRBTGT account’s NTLM hash) and uses it to impersonate any user in the domain, essentially gaining unlimited access to all domain resources.
* **Prevention**:
  + Regularly change the **KRBTGT password** to invalidate any potential golden tickets.
  + Implement **multi-factor authentication (MFA)** for all sensitive accounts to prevent unauthorized access.
  + Monitor and audit the creation of **Kerberos tickets** to detect potential ticket forgery attempts.

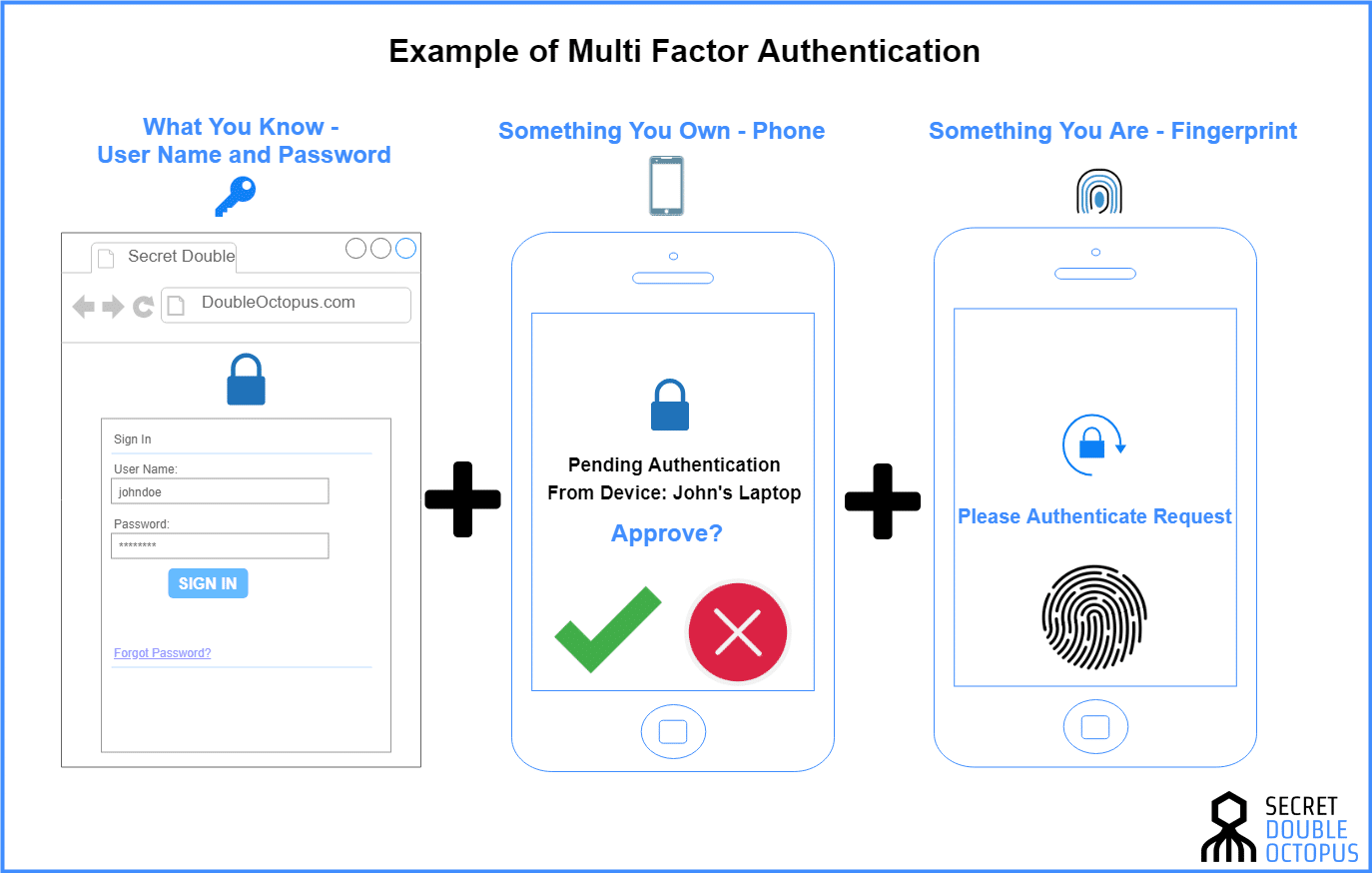


**4. IAM Strategy for Active Directory**

**4.1 Multi-Factor Authentication (MFA)**

**MFA** adds an additional layer of security by requiring more than just a username and password for authentication. This typically involves a combination of

* **something the user knows (password)**
* **something the user has (security token or mobile device)**
* **and/or something the user is (biometric).**

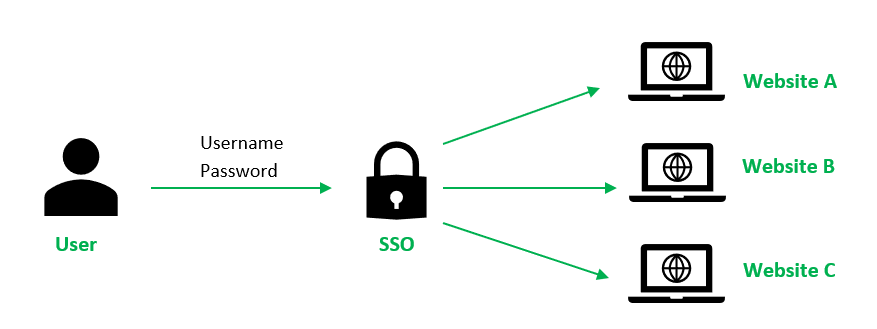


* **Definition**: MFA mitigates the risk of account compromise by making it harder for attackers to gain unauthorized access even if they have stolen the password.
* **Implementation Steps**:
  1. Integrate **Azure AD** MFA or a third-party MFA provider into Active Directory for all sensitive accounts.
  2. Set up **Conditional Access Policies** to enforce MFA for high-risk activities or sensitive resources.
  3. Require MFA for **administrative accounts** and use **MFA-enabled service accounts** for critical applications.

**4.2 Single Sign-On (SSO)**

**SSO** allows users to authenticate once and access multiple systems without needing to re-enter credentials for each system. This approach reduces the number of credentials a user needs to manage, improving both security and usability.

* **Definition**: SSO enables users to authenticate once with a central identity provider and gain access to multiple related systems.
* **Implementation Steps**:
  1. Set up **SSO** using **SAML**, **OAuth 2.0**, or **OpenID Connect** protocols with **Azure AD** or **Keycloak** as the Identity Provider (IdP).
  2. Integrate AD with third-party applications (e.g., cloud-based SaaS services) to provide seamless access without requiring repeated logins.
  3. Ensure **session management policies** are in place to minimize session hijacking risks.



**4.3 Identity Lifecycle Management**

Managing identities throughout their lifecycle—from onboarding to offboarding—ensures that users have appropriate access to resources only for as long as needed.

* **Definition**: Identity lifecycle management involves the automated creation, update, and deletion of user accounts based on changes in employment status or job role.
* **Implementation Steps**:
  1. Automate the **onboarding and offboarding** processes using tools like **Microsoft Identity Manager** or **Okta** to ensure that user access is provisioned and deprovisioned in a timely manner.
  2. Assign access rights based on **roles** and **responsibilities**, ensuring that users only have access to the resources necessary for their tasks (principle of least privilege).
  3. Regularly review access permissions to ensure they remain appropriate.

**5. PAM Strategy for Active Directory**

**5.1 Privileged Account Segregation**

Privileged accounts (those with administrative access) must be isolated from standard user accounts to minimize the risk of privilege escalation.

* **Definition**: **Privileged account segregation** involves restricting the use of highly privileged accounts to specific individuals and systems, ensuring that these accounts are only used when absolutely necessary.
* **Implementation Steps**:
  1. Use **Role-Based Access Control (RBAC)** to assign administrative privileges.
  2. Implement **Just-In-Time (JIT)** access, where privileged access is granted only when needed and for a limited time.
  3. Regularly audit privileged accounts and their activities to detect any misuse.

**5.2 Password Rotation and Management**

Effective **password management** practices, such as **automatic password rotation** and secure password storage, are essential for minimizing the risk associated with compromised credentials.

* **Definition**: **Password rotation** ensures that passwords are changed regularly, reducing the likelihood of long-term exposure in case of a breach.
* **Implementation Steps**:
  1. Use tools like **CyberArk** or **BeyondTrust** to automatically rotate passwords for privileged accounts at regular intervals.
  2. Securely store passwords in **HashiCorp Vault** or a similar enterprise password management solution.
  3. Require **complex password policies** for all privileged accounts to reduce the risk of password guessing or brute force attacks.

**6. Detection and Response to Access Anomalies**

**6.1 Anomaly Detection Using Logs**

Logs from Active Directory events (such as **failed logins**, **changes to privileged accounts**, or **suspicious login times**) provide valuable insights into potential malicious activities.

* **Definition**: **Log-based anomaly detection** involves analyzing event logs for patterns that indicate unusual or unauthorized access attempts, often leveraging machine learning or behavioral analytics.
* **Implementation Steps**:
  1. Enable **detailed logging** for all authentication and authorization events in Active Directory.
  2. Use tools like **Wazuh** or **Splunk** to parse and analyze logs in real-time for signs of anomalies.
  3. Set up **alerting systems** to notify security personnel of suspicious access events.

**6.2 SIEM Integration for Active Directory**

**SIEM** solutions provide real-time monitoring, aggregation, and analysis of security-related data, which can be crucial for detecting and responding to threats.

* **Definition**: **SIEM (Security Information and Event Management)** tools aggregate security event data from across the organization, providing centralized visibility into potential threats.
* **Implementation Steps**:
  1. Integrate **Active Directory logs** with SIEM tools like **Splunk**, **ELK**, or **Wazuh** for continuous monitoring.
  2. Develop custom **SIEM rules** for detecting known attack patterns like **Kerberoasting** or **DCSync**.
  3. Establish real-time **alerting mechanisms** to facilitate rapid responses to security incidents.

**7. Automation and Incident Response**

**7.1 PowerShell and Python for Automated Response**

Automation of incident response tasks is vital to ensuring a rapid reaction to potential threats, minimizing the damage caused by security incidents.

* **Definition**: **Incident response automation** involves using scripts and workflows to automatically detect, contain, and mitigate security incidents without human intervention.
* **Implementation Steps**:
  1. Use **PowerShell scripts** to automatically lock accounts or change passwords after detecting abnormal login attempts.
  2. Leverage **Python** for advanced incident detection and response automation, such as running queries against logs or triggering automatic workflows.
  3. Integrate with **SIEM systems** to trigger automated actions based on predefined alerts.

**8. Conclusion**

This project has successfully demonstrated how an effective **IAM and PAM strategy** can be designed and implemented to secure Active Directory environments. By addressing key issues such as **Kerberoasting**, **Golden Ticket attacks**, and **DCSync**, and leveraging IAM and PAM solutions like **MFA**, **SSO**, **privileged account segregation**, and **automated password management**, this strategy significantly reduces the risk associated with identity and privileged access management. Continuous monitoring through **SIEM tools** and **incident response automation** ensures that organizations can detect and respond to potential threats quickly and effectively.