Application Technical Accounts

Is this one of the biggest

Cyber Threat Risks in the Industry?

LAL-IT Consulting

www.lal-it.co.uk

info@lal-it.co.uk

Author Background

Lalit Choda is an industry leader with over 30 years' experience, dealing with risks around Technical Accounts, from advising C-Level management to establishing and managing successful global regulatory risk programs for numerous Top Tier Global Investment Banks.

Lalit started his career at a major US Investment Bank, building and managing highly scalable Global Front and Middle Office Equity & FX Trading and Order Management and Allocation systems – during this period he was involved in leading and remediating significant audit issues around Technical Accounts on both Operating Systems and Databases. In addition, as Global Risk Officer for the Equities Division, he developed skills in IT Controls, IT Security, Operational Risk, Regulatory and Audit Risk.

Over the last 15 years, Lalit has driven several large Global Regulatory Information Security Programs for multiple Global Investment Banks covering Access to Production, Technical Accounts, Privilege Access Management, Segregation of Environments and Segregation of Duties.

Partnering with Global CISO/IAM & IT Security heads he has been responsible for establishing the overall Global Strategy and Risk-Based approach for these programs, collaborated with Central Infrastructure teams to define/design the supporting platform/infrastructure capabilities needed, as well as setting up and driving the overall Risk Reduction activities.

Abstract

Application Technical Account risks pose one the biggest Cyber and Insider threat risks to any organisation. If you look at most of the biggest cyber incidents, they all have one thing in common i.e. the discovery and use of highly privileged Technical Account Credentials to access and compromise systems/data.

Regulators and Auditors around the world are now very focussed on this significant exposure that exists in most organisations. Several years ago, their primary focus was on Privileged Access Management, Segregation of Duties and Access to Production Controls, but are now raising significant audit points against numerous financial institutions.

Organisations cannot ignore or underestimate this risk and sheer scale and effort required to get these risks under control, to avoid being the next company facing a major cyber or internal threat incident.

This white paper will focus on core foundational control principles organisations will need to establish to understand the risks around Technical Account Credentials and the key things they will need to consider from a core capability standpoint to manage the risks in an effective and sustainable way.

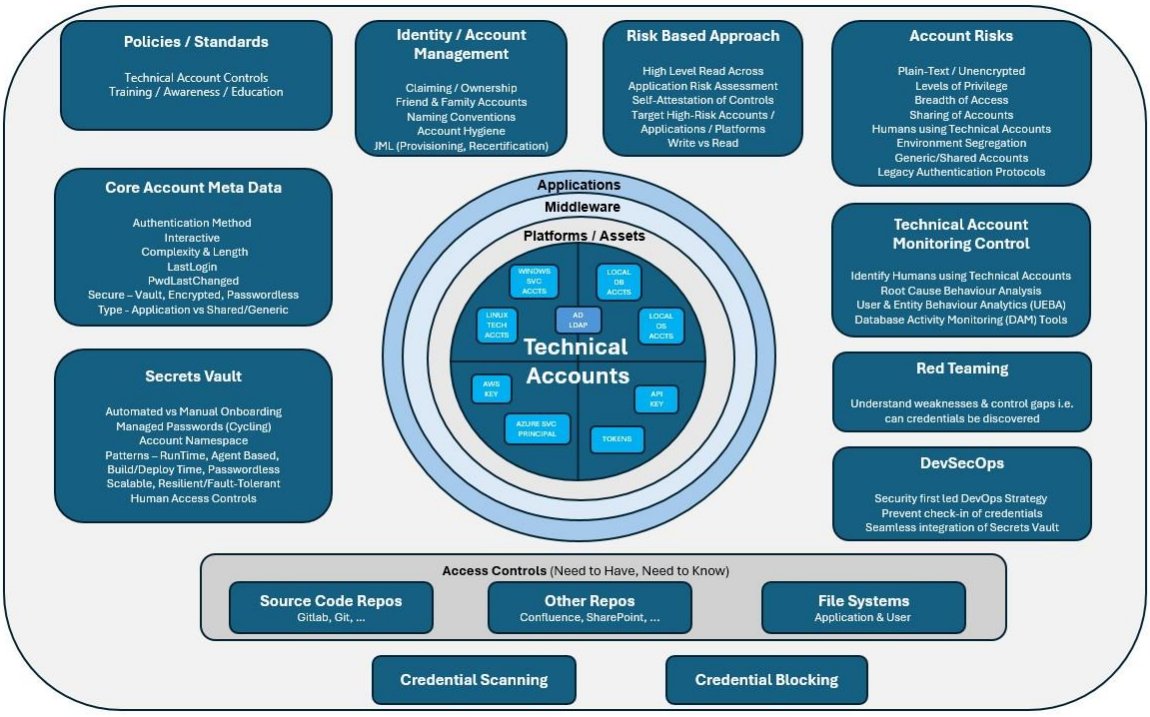
If you are an organisation that has faced a cyber incident relating to the exposure of technical accounts; been audited and major risks identified or want to understand more about the risks and how to go about establishing a risk program to manage and remediate the risks, contact us at **info@lal-it.co.uk**.

Introduction

As an organisation embarks on establishing a risk program around Application Technical Accounts (also known as Service/System Accounts) there are several core foundational control principles that will need to be established to understand the size and scale of the risk exposure that exists.

There are also several things an organisation will need to consider from a core capability standpoint to manage the risks in an effective and sustainable way.

The below high-level diagram attempts to summarise the key areas that an organisation will need to factor into any risk program they establish.



The following sections of this white paper explain in detail the key areas highlighted above.

1. Account Inventory and Ownership

A critical part of establishing any program around managing risks around technical account credentials starts with establishing an inventory of all technical accounts. This can be one of the biggest challenges, as technical accounts do not always exist in authoritative repositories and organisations may not have a global centralised Identity Management system to inventory and manage accounts/identities.

Directory based technical accounts are much easier to inventory and manage (including associated JML processes) vs local accounts defined directly on a platform (e.g. operating system, database etc). Locally defined technical accounts will typically require custom feeds to be built, that can take a significant amount of time and effort to develop.

Once an inventory of accounts is established a key part of driving any risk reduction program requires that technical accounts be owned by an application/owner, so it is clear who owns the risk and accountable for driving any risk reduction activities

Through the process of claiming/owning technical accounts, an organisation will uncover that many accounts are orphaned and have no clear owner or could be legacy accounts that are not in use. This gives organisations an opportunity to clean-up/remove these accounts and thus reducing the overall surface area of risk.

2. Understanding the Controls & Risks around Technical Accounts

- **Authentication Method –** does the credential use passwords for authentication or other forms of authentication e.g. Certificates, Kerberos, Tokens

- **Levels of Privilege** – what access does the account provide, is it full administrative access on the domain/asset or specific read or write privileges. This can help understand the relative risks of the account from a privilege standpoint. Understanding the levels of privilege for a technical account is not always that straightforward, as requires complex integration of the entitlement models on each platform.

  Another challenge is that you will find technical accounts can be over privileged – in many cases we see that the levels of access granted can be excessive i.e. given write access when in fact the account only needs read-access.
- **Breadth of Access** – what is the breadth of access for the account – is the account entitled to just one asset/component or is it entitled to hundreds/thousands of assets/components - clearly the more assets it has access to the higher the risk of the account.
- **Interactive Accounts** – platforms/accountTypes that support interactive logins pose significant risk to an organisation and a clear attack vector for a hacker or insider threat. A human can effectively impersonate and become the technical account and gain access to highly privileged accounts, allowing a threat actor to access and compromise systems/data.
- **Environment Segregation** – does the account only have access to one environment e.g. Prod, UAT, Dev or has the account got access to multiple environments – accounts with a lack of environment segregation increase the risk of lateral movement, something a hacker will compromise.
- **Password Discovery** – a huge challenge faced by the industry is that application credentials/passwords are not always secure, and in many cases, passwords are held unencrypted in plain-text, within source code repositories and other places e.g.

Confluence, SharePoint, File Systems, User Home Directories ... This is easy picking for threat actors and Red-Team groups.

Dealing with the unencrypted credentials is a significant undertaking for most organisations as it touches on JML process, Secrets Vaulting, Detect and Protect controls to identify these risks and stop future exposures from occurring. The effort to remove unencrypted credentials can be a significant effort for most application teams.

- **Password Complexity** – whilst this can be challenging to measure, it is important to understand if there are accounts that have non-complex passwords, as these will be prone to brute-force attacks.
- **Password Cycling** – industry best practice is to cycle technical account passwords on a regular basis. The longer the gap in a password being cycled the greater the risk of people knowing and using the password. There are many benefits to passwords being cycled as it reduces risks around transfers, leavers, mitigates/removes credential exposure in legacy code/scripts, helps uncover unknown dependencies and sharing of credentials etc.

  Password cycling is incredibly challenging given the above risks - if the dependencies are not known in advance i.e. all scripts/code that use the credential (including sharing of the credentials by other applications), there is an elevated risk of operational impact of applications breaking, when a password is cycled.

  Cycling cannot be considered in isolation – there needs to be a sustainable solution for automating the cycling activities (manual cycling does not scale or repeatable) and tightly coupled to an organisations Secrets Vault capability that needs to support cycling.
- **Account Usage** – if an organisation can determine whether the account is in use, this can drive hygiene activities to remove legacy/inactive accounts and achieve quick wins in removing driving down the surface area of risk. For some platforms/accountTypes this is

quite straightforward as account usage data is readily available (but needs to be extracted), but for others quite is quite complex or does not exist.

- **Account Sharing** – unfortunately, we have seen in many organisations, application credentials being shared across teams/applications, which breaks the need-to-have/need-to-know and segregation principles. This can result in excessive privileges being granted and makes the securing and cycling of passwords securing incredibly challenging, given the risk of operational impact and coordination required to segregate the accounts.
- **Humans using Technical Accounts** – the biggest risk around weak technical account controls, is the discovery and ability to compromise the credentials by both external and internal threat actors. When humans use technical accounts (including shared/generic accounts) it can be hard to detect and there is a lack of repudiation i.e. you may not know who accessed/updated the data, which is a huge exposure for any organisation and could have major implications for the integrity of a firm's books and records.

  One of the biggest challenges however, is “seeing the wood from the trees” – there could be both valid and inappropriate reasons why someone is using a technical or generic/shared account. Some teams use these accounts to compensate for weaknesses in Privileged Access capabilities, lack of automated Deployment tools and some use them as an easier way to perform BAU activities, rather than going through formal approved PAM processes. Users can also show up as indirectly using a technical account when running support scripts (that internally connect using a technical account) which can lead to false positives being reported.

A key part of any Technical Account risk remediation program is to understand the above control gaps, to help assess the overall size and scale of the risk exposure and define a risk-based approach.

So how does an organisation start to tackle this huge 'Elephant in the room' ? :

I was once asked by a CISO at a leading investment bank, why we cannot fix this risk in 1-2 years and explained fixing both the current exposure as well as preventing future risks from repeating, touches on the whole technology stack and processes in any organisation from IAM / JML, SLDC / SecDevOps, Application Security, Secrets Vaulting, Credential Scanning etc.

Unfortunately, there is no 'magic bullet' here, you cannot just buy a product that will take care of this for an organisation (it will be many products/processes), you cannot hire a few key SMEs/Consultants to quickly fix this – for most large global organisations this will be upwards of a **$20-50M** spend/investment.

Organisations will need to consider several things as they embark on their journey in tackling one of the biggest and most complex security risks, around technical accounts:

- **High Level Platform Read-Across** – try to determine where the biggest risks exist, by performing a high-level control effectiveness assessment for each platform. This will help senior management understand the key exposures that exist and help define your initial risk-based approach, whilst you can establish a much better view of the risks.
- **Account Inventory/Claiming** - investment in an identity management capability, to manage an inventory of all technical accounts including the ability to claim ownership of the account is foundational to any program to measure and drive remediation accountability as well as drive clean-up of orphaned/unused accounts.
- **Red-Teaming** – we would recommend that you establish regular red-team testing of your environment, to see if technical account credential exposures can be discovered.

- **Policies/Standards/Controls** - organisations will need to ensure that their policies/standards and controls are robustly defined and clearly highlight what is expected around management of technical account credentials.
- **Training/Awareness/Education** - there needs to be a robust training, awareness and education campaigns so the organisation is aware of the risks around technical accounts and that it is imperative that teams meet core control requirements around technical accounts being secure, cycled, segregated etc. It should also be made clear that technical accounts must not be used by humans and if detected, could lead to a disciplinary procedure, up to and including termination.
- **Control Effectiveness –** it is critical to be able to measure the controls in place against technical accounts. This can be challenging to do given:
    - the vast array of platforms where technical account credentials that may exist in the environment, each with their own level of control maturity.
    - each platform/accountType will have custom meta data around the account e.g. interactive, authentication method, password length/complexity, passwordLastChanged, lastLogin etc
    - for some platforms e.g. where directory services are used, some of this data can be easy to collect, but for others where local accounts exist on the platform, custom connectors will need to be developed to extract this data, which can take significant effort and time.
- **Attestation –** given some organisations may not have mature account inventories and may not be able to measure control effectiveness, they may need to consider some level of self-attestation processes, to initially establish control effectiveness, until this data can be sourced via authoritative sources.

- **JML Processes** – these are key to dealing with implementing controls in a sustainable manner and in part are the root cause for technical account issues with an organisation i.e. provisioning processes should ensure that any new credentials/passwords are not known by humans and feed directly into a Secrets Vault capability and therefore ensuring credentials are secured by design.
- **Secrets Vault** – a core part of any remediation strategy will involve delivery of a Secrets Vault capability. This is no small undertaking and will need to handle global scale, high volumes, and provide a resilient/fault tolerant capability. There are various integration patterns that need to be supported given an organisation will have both strategic and legacy applications i.e. Run Time, Agent Based, Build/Deploy Time patterns etc, each with pros and cons from a control effectiveness and cost/effort standpoint.

  Other key considerations include how credentials are onboarded onto the vault, i.e. via centralised account inventory feeds or manually by application teams. Account namespace definitions will need to be established, to appropriately describe the credential type/instance being onboarded, to support future processes like managed password cycling, reporting etc.
- **Naming Conventions / Environment Segregation** – an organisation will need to consider naming conventions for accounts e.g. separate accounts for each environment e.g. account1_prod, account1_uat ...
- **Automated Password Cycling Capabilities** – managed automatic cycling capabilities is not something you typically get out the box from most Secrets Vault solutions and will require significant investment in delivering this capability per platform/accountType. There are clearly risks with automating the cycling process, as there could be edge cases with synchronisation between the end point, the secrets vault and the application, that the application would need to handle e.g. through retry connection logic.

- **DevSecOps –** organisations will need to take a security first led DevOps strategy i.e. automating the integration of security at every phase of the SDLC from development, testing and deployment. These would typically include seamless integration of technical account credentials with the Secrets Vault capability and then maturing to include prevent controls, to stop check-in of unencrypted credentials into source code repos.
- **Credential Scanning** – an organisation will need the ability to scan and identify credential leaks e.g. unencrypted plain-text passwords in source code or other repositories. This is non-trivial exercise given the number of false-positive leaks that can be identified and the ability to scan many repositories e.g. Gitlab, Git, SharePoint, Confluence, File Systems etc. Processes will need to be put in place to mature the scanning capability, including managing false-positive leaks through attestation processes.
- **Preventing Credential Check-In of Credentials in Source-Code –** to create a sustainable set of controls, you will need to stop new credentials from being checked into source code repositories. Very few organisations have achieved this level of maturity and requires a significant investment in the SecDevOps capability, including a separate implementation for each source code repository being used.
- **Passwordless vs Password Strategy** – to stop the bleeding you need to consider moving away from passwords to passwordless credentials e.g. certificates, tokens etc. Note whilst this addresses many of the risks associated with passwords, it does not mitigate all the risks, given these accounts can also be compromised.
- **Account Hygiene** – one of the quickest and most effective forms of risk reduction is to drive a global hygiene program, to remove legacy/inactive accounts. Whilst in principle this sounds quite straight-forward, it is quite challenging, as depends on having account usage data available, for some platforms custom connectors will need to be built.

- **Legacy Authentication Protocols –** remove the use of legacy protocols in key applications, particularly any internet facing applications, as they do not support conditional access or multi-factor authentication and can easily be compromised.
- **Account centric driven remediation vs Scanning centric driven remediation** – we would advise organisations to use a hybrid approach of account centric led assessment / remediation alongside credential scanning-based assessment and remediation.
- **Monitoring Controls to Detect Human use of Technical Accounts** - this is one of the most challenging areas to deliver a strong and effective control, but one of the most critical, given an external/internal threat actor will always find a way to discover and misuse a technical account credential.

  Unfortunately, there is no one size fits all product that you can just onboard and deploy to your environment. There are challenges around sheer volume of events, multitude of platforms/accountTypes you need to monitor, each with their own maturity/availability of meta data and biggest challenge by far is, "seeing the wood from the trees."

  Organisations should consider Intelligence and Behaviour Analytics capabilities (i.e. UEBA tools) to develop scalable and sustainable solution, as it is not possible or practical to look at every single event that could be a violation and instead look at outliers/anomalies as a more effective way to manage the risk.

  There will be more fundamental challenges an organisation will need to deal with around why humans are using technical accounts e.g. cases deemed by teams to be valid/BAU which need to be understood, rooted out/stopped, otherwise it becomes an impossible task to see the wood from the trees.

  Organisations will need to deliver strategic capabilities, to migrate away from old legacy procedures/processes, where technical accounts are used to perform BAU activities.

Closing Remarks

In closing, addressing risks around technical accounts is likely to be one of the most complex and challenging security vulnerabilities to address in any organisation. It touches on all aspects of IT processes and controls, from IAM (Identity Management, JML), SDLC/DevSecOps, Secrets Vaulting, Scanning, Monitoring etc, many of which will require significant strategic investments to deliver the required infrastructure capabilities to support risk reduction and sustainable controls to “stop-the-bleeding".

Unless an organisation already has mature controls and capabilities for managing technical account credentials, they are likely to uncover that thousands of unencrypted plain-text credentials exist in the environment (that can be easily discovered or already known). There are likely to be many passwords that have not been cycled and a sizeable percentage of legacy/inactive accounts, increasing the surface area of risk. It highly likely that these credentials are being used by humans and therefore could pose risks around the integrity of a firm's books and records. Monitoring controls will need to be established, to reduce and identify any inappropriate activity.

The effort to remediate these risks will be significant, as teams will need to migrate unencrypted credentials onto a Secrets Vault solution or passwordless credentials. This will introduce significant change into the environment and operational risk, where applications could break due to unknown dependencies e.g. when passwords are cycled.

If you are an organisation that would like to understand more about the risks around technical accounts and how to go about establishing a risk program to manage and remediate the risks, contact us at **info@lal-it.co.uk**.