

Scoring Azure permissions with metric spaces

Christophe Parisel*

April 21, 2025

Abstract

In this work, we introduce two complementary metrics for quantifying and scoring privilege risk in Microsoft Azure. In the Control Plane, we define the *WAR distance*, a superincreasing distance over Write, Action, and Read control permissions, which yields a total ordering of principals by their configuration power.

In the Data Plane, we present a *blast radius distance* for measuring the maximum breadth of data exfiltration and forgery, leveraging the natural ultrametry of Azure Tenants clustering hierarchy

Together, these metrics offer a unified framework for proactive IAM analysis, ranking, lifecycle monitoring, and least privilege enforcement.

1 Introduction

In modern cloud environments like Microsoft Azure, overseeing the security of privileged access is a complex and critical challenge, especially when it comes to *Non Human Identities* represented by Application SPNs and Managed Identities. As organizations move to the cloud, the need for robust systems to measure and mitigate risk associated with identity and access management (IAM) has become more pressing. One of the most significant risks is the potential for excessive or misconfigured privileges granted to NHIs aside of the standard groups membership process that human principals typically follow, which can lead to unauthorized data exfiltration, integrity breaches, or unintended privilege escalation.

Azure provides fine-grained access control through role-based access control (RBAC) [1], allowing principals (users, service principals, or managed identities) to be assigned permissions on resources across various scopes such as tenants, management groups (MG), subscriptions, resource groups (RG), and individual resources. However, understanding the exact risk posed by these permissions, particularly when they are distributed across different scopes and types, is a non-trivial problem.

*Email: ch.parisel@gmail.com

This paper attempts to solve that problem leveraging natural metric spaces found in Azure. We begin with an overview of the WAR distance metric, explaining how it combines scope and permission type to measure control-plane privilege. We then present the data-plane ultrametric for assessing risks related to data exfiltration and forgery.

2 Measuring Control Plane Privilege: The WAR Distance

2.1 Motivation

Control plane permissions in Azure are assigned through role definitions containing **actions**. These define what operations a principal can perform on resources, including reading configurations, starting virtual machines, or deleting services. Each action is granted over a *scope*, such as a subscription or a resource group.

To measure the *privilege strength* of any given Azure principal from a control-plane perspective, we introduce a distance function that reflects both the **scope** of permissions and their **type**. However, the space formed by combining scope and type is not totally ordered. The notion of a **distance** allows us to circumvent this limitation by imposing an opinionated but consistent ordering.

2.2 WAR: A Scope-Type Distance

We define the **WAR distance**—short for Write/Action/Read—as a function that quantifies the distance between two sets of control-plane permissions. In particular, we are interested in computing the distance between a principal and the **null origin**, i.e., a principal with no assigned permissions.

The WAR distance lives in the Cartesian product of two independent spaces:

- **Scope space:** Ranging from tenant-level to sub-resource level:

Tenant > MG > Subscription > RG > Resource > Sub-resource

- **Permission type space:** We assume a control-plane risk-based order:

Write > Action > Read

2.3 The Distance Model

To define a distance that incorporates both scope and type, we assign numerical weights. Let each role assignment be modeled as a tuple $(w, a, r)_s$, where w , a , and r are integers representing the existence of write, action, and read permissions assigned at scope s ranging from Tenant to Sub-resource. The following

discrete sets are used for the numerical weights:

$$\begin{aligned} W &= \{600, 500, 400, 300, 200, 100\} \\ A &= \{90, 80, 70, 60, 40, 20\} \\ R &= \{9, 8, 7, 6, 4, 2\} \end{aligned}$$

The sets are chosen such that the minimum difference between any two elements in W (which is 100) is strictly greater than the maximum value in A plus the maximum in R (i.e., $90 + 9 = 99$). Similarly, the smallest gap in A is 10, which exceeds the maximum value of R (which is 9). This superincreasing [4] structure guarantees that the total score can be decoded unambiguously into its components.

Superadmin rights (denoted by wildcards in Azure permissions) are routinely assigned to built-in roles like Owner or Contributor, and also to custom roles. At subscription level and above, this entails much more powerful rights than the Write permission. Taking wildcards into account, the permissions order becomes:

Wildcard => Write > Action > Read

We also add three more weights to the Write set: one for Tenant superadmin (900), one for Management Group superadmin (800), and one for Subscription superadmin (700):

$$\begin{aligned} W &= \{900, 800, 700, 600, 500, 400, 300, 200, 100\} \\ A &= \{90, 80, 70, 60, 40, 20\} \\ R &= \{9, 8, 7, 6, 4, 2\} \end{aligned}$$

Whenever a wildcard is found at Subscription level or above, we replace the usual Write weight by its superadmin equivalent.

2.4 Privilege Norm and Principal Ranking

We define the WAR distance between two points $P_1 = (w_1, a_1, r_1)$ and $P_2 = (w_2, a_2, r_2)$ as:

$$d(P_1, P_2) = |w_1 - w_2 - a_1 - a_2 - r_1 - r_2|$$

This function, which we call the *L1-minus distance*, differs from the Manhattan (L1) distance. While the Manhattan distance is always a valid metric, the L1-minus is not point-separating in general. However, due to the superincreasing property of the sets W , A , and R , our version is indeed point-separating within our domain, and hence qualifies as a proper metric.

Given a principal $P = (w, a, r)$, its privilege level (or strength) is:

$$\|P\| = w + a + r$$

Norm $\|P\|$ induces by the WAR distance provides a scalar value to rank principals by privilege. The maximal norm is $999 = 900 + 90 + 9$, corresponding to a “superadmin” principal with full permissions at the tenant level. The null principal has $\|P\| = 0$.

2.5 Groups membership

So far, we have only addressed the case of roles being granted directly to a principal. Principals often belong to groups, so they enjoy roles inheritance. To measure the actual norm of a principal belong to groups, we start by measuring its raw WAR norm, as we have just done, and we measure the norm of all groups (including nested ones).

Consider the 3D vector of a WAR norm as a triangle with vertices W, A and R, centered at origin.

The actual norm of the principal is the convex envelope of all triangles of the groups the principal belongs to, plus the triangle of its raw norm. For a concrete implementation, see [2]

2.6 Benefits of the WAR Distance

The WAR distance is not only a tool for theoretical privilege comparison—it offers immediate operational benefits:

1. **De-Escalation Planning:** Principals can be sorted in decreasing order of $\|P\|$ to identify the most powerful and risky SPNs and prepare targeted de-escalation strategies.
2. **Target-Based Comparison:** Given a principal’s current tuple and a target (less privileged) configuration, the WAR distance quantifies the privilege gap. This supports quantitative privilege reduction campaigns.
3. **Lifecycle Monitoring and Anomaly Detection:** By tracking $\|P(t)\|$ over time, one can monitor the privilege evolution of any principal. Spikes or regressions in privilege level can be used to trigger alerts or to detect misconfigurations and privilege escalations.

The WAR Distance is implemented in Azure Silhouette [2], an open source tool developed by the author.

3 Measuring Data Plane Privilege: The Blast Radius Distance

3.1 Introduction

In the context of data plane security, two fundamental risks dominate the threat landscape: **data exfiltration** and **data forgery**. These risks arise from over-

privileged identities possessing `dataActions` on resources, often across organizational boundaries.

Data Exfiltration (Confidentiality). Read-level `dataActions` grant a principal the ability to access and extract sensitive information from Azure data resources. When such permissions are distributed across logically independent scopes—such as separate business units, regions, or environments, the risk of large-scale exfiltration escalates. In the event of a compromise, an attacker could harvest data across these boundaries, breaching confidentiality in a systemic fashion.

Data Forgery (Integrity). Write-level `dataActions` empower a principal to modify or overwrite existing data. This capability introduces a serious risk to data integrity: a compromised identity with write access could forge fraudulent data, inject malicious payloads, or corrupt operational information. The breadth of such permissions determines the potential impact of a forgery attack.

3.2 Understanding data exfiltration and forgery in the Azure context

Let's revisit permissions types and scope under the light of data plane security in Azure, and compare them with what we proposed in the control plane.

3.2.1 Prioritization of `dataActions` Permission Types

Azure's data plane permissions comprise four primary types:

- **Wildcard (*)** – Grants unrestricted access to all data plane actions within the scope of the assigned role.
- **Write** – Enables creation, deletion, or modification of customer data.
- **Read** – Grants visibility into customer data.
- **Action** – Typically allows enumeration operations, such as listing blobs, keys, or containers.

Our model introduces a custom risk-driven prioritization, emphasizing the potential for data breach scenarios:

1. **Read or Write** permissions are both high-risk, as they enable data exfiltration (confidentiality breach) and data forgery (integrity compromise), respectively.
2. The combination of **Read and Write** permissions elevates the risk further, enabling simultaneous confidentiality and integrity attacks. This combination defines the highest data-centric privilege.

3. **Action** permissions are comparatively low-risk in the data plane and are excluded from blast radius scoring due to their limited exploitability. They are still opening doors for reconnaissance, so they should not be ignored altogether. But as far as data security is concerned, they play a secondary role.
4. **Wildcard** permissions are treated as functionally equivalent to the union of Read and Write, as the latter already imply maximal impact on both confidentiality and integrity.

3.2.2 Preservation of Organizational Context

To focus on business-impactful privilege boundaries, we intentionally collapse all scopes *below* the subscription level. Whether a principal has access at the customer database, resource group, or entire subscription level, the risk to business-critical data remains fundamentally the same in the data plane.

In contrast, the hierarchy of Management Groups (MGs) is preserved *as-is*, due to its organizational relevance. This structure encapsulates real-world data segmentation practices and is important in modeling blast radii:

- **Production vs. Non-Production** – Customers frequently segment prod and non-prod environments via separate MGs. Cross-boundary access is a clear data risk and must be surfaced.
- **Geographical Boundaries** – Regional compliance requirements (e.g., GDPR, data residency laws) are often implemented using MG segmentation. Cross-region data access poses regulatory risks.
- **Business Units (BUs)** – Enterprises model internal departments or legal entities as nested MGs. In industries like finance, this ensures adherence to “Chinese Wall” policies. Access across BUs signifies a significant breach of data containment.

Preserving the MG hierarchy in our blast radius computation allows us to align privilege assessment with customer-specific risk profiles and real-world operational boundaries.

3.2.3 Comparison with Control Plane Needs

Control Plane Context Scoping In contrast to the data plane, control plane operations can have profound consequences even at fine-grained scopes such as individual resources or sub-resources. As such, the WAR metric captures permissions down to these granular levels. On the other hand, as control plane scopes widen, their marginal impact tends to plateau: once a principal can configure one Management Group (MG), they can likely affect others within the tenant in a similar fashion. Therefore, the WAR model simplifies the hierarchy by collapsing all MGs into a single abstract level, reflecting the observation that compromising one MG is operationally similar to compromising several within the same tenant.

Permission Types In the WAR model, permission types are ordered based on their potential to cause configuration damage or privilege escalation. The ordering is as follows:

*** >= Write > Action > Read**

This reflects the fact that **Write** permissions can be used to take over or reconfigure resources, while **Read** permissions alone pose relatively little configuration risk. Additionally, wildcard permissions (e.g., *****) in the control plane are particularly hazardous (especially for scopes at or above the subscription level) because they enable arbitrary actions and may implicitly include future actions not currently defined and potentially catastrophic breaking changes.

Threat focus Data protection focuses on direct threats to data confidentiality (exfiltration) and integrity (forgery). In contrast, control plane risks often center on misconfigurations, such as disabling audit logs or weakening encryption policies.

Scope Modeling In the data plane, we abstract away fine-grained resource details and evaluate risk at the subscription level. This choice reflects operational realities: access to a resource or its containing subscription typically results in equivalent data exposure. However, we retain the hierarchical structure of Management Groups to capture organizational segmentation (e.g., production vs. non-production, business units, or geographic regions).

Permission Prioritization Compare permissions prioritization in the control plane WAR model:

*** > Write > Action > Read**

with our data plane model, where confidentiality and integrity are equally critical:

*** = (Write and Read) > (Write or Read) > Action**

Risk Implications A high data plane blast radius signals that a principal holds read or write access across multiple independent organizational zones. This fragmentation amplifies the risk of data exfiltration or tampering, distinguishing it from the predominantly configuration-centric risks of the control plane.

3.3 The Blast Radius Distance

To capture and quantify data exfiltration and data forgery, we introduce a new purpose-built metric in Azure: the *Blast Radius Distance*. This distance measures how far a principal's data privileges extend across the Azure resource hierarchy. It is designed to reflect both the depth and dispersion of sensitive **read** and **write** permissions, enabling a structured assessment of data risk exposure for any given identity.

3.3.1 The Distance Model

We leverage the natural ultrametric distance [3] between two dataActions permissions p1 and p2 of the principal in the Azure Tenant hierarchy:

- Depth 0: Tenant (Root)
- Depth 1+: Management Groups (MGs), up to 6 officially
- Final depth “d”: Subscription (treated as the leaf, with all sub-resources collapsed upward)

So depth ranges from 0 to a maximum of 7.

To account for groups membership, we must consider all pairs of dataActions permissions inherited from the groups the principal belongs to.

3.3.2 The distance function

For any two scopes s1 and s2 of p1 and p2, let d be the depth of their Lowest Common Ancestor (LCA). We define the base (ultrametric) distance as $\text{Base.distance}(s_1, s_2) = \frac{1}{2^{2d+1}}$

The base distance ranges from 0.0 (no data plane rights or infinitely small blast radius) to 1.0 (Tenant wide radius) It defines measurements and holes in the $\frac{1}{2^n}$ series: measurements appear at odd powers in the series, and holes at even ones.

To reflect the permissions ordering, we ignore all permission pairs containing an ‘Action’. For the remaining pairs, we consider the shallowest scope depth of a pair of permissions as $\min(s_1, s_2)$ and define an ‘impact’ coefficient:

- Impact = 2 if the pair contains a wildcard at shallowest scope depth, or if at least ‘Write’ and ‘Read’ atomic permissions are assigned to the principal at shallowest scope depth.
- Impact = 1 if the ‘Read’ or ‘Write’ permission is assigned to the principal at shallowest scope depth, but not both simultaneously at this depth.

The final (ultrametric) distance is: $\delta(s_1, s_2) = \frac{\text{impact}}{2^{2d+1}}$

Concretely, for high-risk permissions, since $\text{impact}=2$ the final distance is $\frac{1}{2^{2d}}$ whereas for lower-risk permissions since $\text{impact}=1$ the final distance is $\frac{1}{2^{2d+1}}$

This nuance defines a hierarchy within the hierarchy: it “fills in” holes in the series in an orderly fashion: for identical scopes, high risk permissions at even locations have a higher measurement than lower risk ones at odd locations.

3.3.3 Defining Blast Radius

We define the data plane blast radius by leveraging a fundamental property of ultrametric distances: the **strong triangle inequality**, given as:

$$\text{distance}(a, c) \leq \max(\text{distance}(a, b), \text{distance}(b, c))$$

This property allows us to reason about the *data perimeter* of a set of permissions both conceptually (the data perimeter is the blast radius times $2\pi r$) and practically, in a stable and computationally efficient way.

Definition: For a principal P with a set of data plane permissions $\{p_1, \dots, p_n\}$, we define the blast radius as:

$$\text{BlastRadius}(P) = \max_{i,j} D(p_i, p_j)$$

where $D(p_i, p_j)$ is the ultrametric distance between permissions p_i and p_j .

This value represents the worst-case spread of P 's data access: how far apart the most distant permissions are, organizationally. A higher blast radius suggests that the principal's access spans diverse or unrelated organizational zones, thus increasing the risk of data exfiltration or data forgery.

3.4 Benefits

The blast radius provides a compact scalar summary of the dispersion of sensitive permissions. It can be used to prioritize review and remediation efforts, especially in large environments with complex identity and access configurations.

4 Conclusion

This paper presents a dual-metric framework for scoring privilege strength and data risk in Microsoft Azure environments.

We introduced the *WAR distance*, superincreasing over Write, Action, and Read permissions, which enables a total ordering of principals by their control-plane privilege. The metric balances scope and operation type in a principled way, providing a foundation for de-escalation strategies, anomaly detection, and least-privilege governance.

Complementing this, we have introduced the *Blast Radius*, an ultrametric-based metric for quantifying the maximum extent of Data Plane permissions in Azure cloud environments. By emphasizing that both 'Read' and 'Write' dataActions are most critical, and by collapsing granularity below the subscription level, we capture the risk that a principal's permissions span disparate organizational domains. This model complements control plane metrics such as WAR by providing a clear, mathematically robust measure of data-centric risks.

Together, these metrics form a coherent approach to reasoning about privilege in Azure—offering both mathematical rigor and practical applicability. By enabling structured comparisons, lifecycle monitoring, and risk quantification, they empower security teams to proactively reduce overprivilege, enforce least privilege, and build more resilient cloud infrastructures.

A note on the nature of scores: The WAR model produces a norm—induced by the WAR distance—between two principals, enabling ranking and comparative privilege analysis. In contrast, the data plane blast radius is computed over

the permissions of a single principal. It is not a norm but a scalar measurement, derived from an ultrametric distance within that principal's permission set. While two principals' blast radii can be compared, the result is not a formal distance between them.

Future work

In the Control Plane, as of today Azure makes no clear distinction between pure configuration operations and IAM operations (role assignments). We plan to separate scoring IAM operations and configuration operations using a new, dedicated norm for the former and ignoring role management operations for the latter.

The WAR norm of a single subscription-bound principal is the same as the WAR norm of a principal interacting with many subscriptions. We plan to offer a better resolution for distinguishing both, since the latter is more akin to a MG-bound principal.

5 Appendix A: WAR norm examples

5.1 The WAR norm table

Scope	Has superadmin permission	Has write permission	Has action permission	Has read permission
Tenant	+900	+600	+90	+9
Management group	+800	+500	+80	+8
Subscription	+700	+400	+70	+7
Resource Group		+300	+60	+6
Resource		+200	+40	+4
Subresource		+100	+20	+2

Figure 1: The WAR norm table

5.2 Examples

Here are a few examples of silhouette configurations based on the WAR norm table (by decreasing order of privileges):

1. 999 corresponds to Tenant admin
2. 888 corresponds to management group level superadmin
3. 777 corresponds to subscription level superadmin
4. 477 corresponds to subscription level for W, A and R
5. 466 corresponds to subscription level for W, resource group level for A and R

6. 377 corresponds to resource group level for W, subscription level for A and R
7. 367 corresponds to resource group level for W and A, subscription level for R
8. 346 corresponds to resource group level for W and R, resource level for A
9. 026 corresponds to subresource level for A, resource group level for R, and no W action
10. 000 corresponds to no control plane rights (except IAM roles management, as explained above)

6 Appendix B: Blast Radius Examples

6.1 Single Permission at the Management Group Level

Scenario: A principal is granted a single `Write` data plane permission scoped at the management group level. The management group resides at depth 3 in our hierarchical model (where `Tenant` is at depth 0, followed by `Management Groups`, and `Subscriptions` at the leaf level).

Calculation: With only one permission, there are no permission pairs to compare, and therefore no Lowest Common Ancestor (LCA) to compute. The impact factor for a `Write` permission is 1. Since the permission resides within a single cluster at depth 3, the blast radius is:

$$\text{BlastRadius}(P) = \frac{1}{2^{2 \cdot 3 + 1}} = 0.0078125$$

This value reflects localized motion, fully contained within the given management group hierarchy.

6.2 Permissions Within the Same Management Group Branch

Scenario: A principal holds two data plane permissions:

- A `Write` permission at a management group (depth 3), same as in the previous example.
- A `*` (wildcard) permission at a subscription under that same branch, located at depth 5.

Calculation: The Lowest Common Ancestor (LCA) of both permissions is the original management group at depth 3. Although the wildcard permission implies a potentially higher impact (up to 2), the impact is computed *at the shallowest depth*. Since only the `Write` permission is shallower than the wildcard, we retain the former hence the effective impact is 1.

$$\text{BlastRadius}(P) = \frac{1}{2^{2 \cdot 3 + 1}} = 0.0078125$$

The blast radius remains unchanged compared to the single-permission case, as the additional permission does not widen the organizational scope beyond the original cluster.

6.3 Permissions Spanning Different Management Group Branches

Scenario: The principal holds:

- A **Write** permission in a management group at depth 3.
- A **Read** permission in a separate management group at depth 2 .

The two permissions reside in distinct branches. Their Lowest Common Ancestor (LCA) is a shared parent at depth 1.

Calculation: Both permissions have an impact factor of 1, and the LCA sits at depth 1:

$$\text{BlastRadius}(P) = \frac{1}{2^{2 \cdot 1 + 1}} = 0.125$$

This significantly higher blast radius reflects the broader organizational spread of the principal's data access, signaling increased risk of data exfiltration or integrity compromise.

6.4 Replacing Read permission and moving to the Tenant

Scenario: A principal holds the usual 'Write' permission at MG depth 3. The principal has a 'Read' permission under the Tenant (recall in our model, the Tenant sits at depth 0). The principal is then granted a second permission under Tenant scope, this time it is a 'Write' permission.

Calculation: The principal has 3 permissions, but since the last two 'Read' and 'Write' are attached to the exact same scope, we collapse them into a 'Write+Read' permission. Since it is at shallowest depth zero, and since it represents a high risk, its impact is 2. Finally, the LCA between MG and MG' is 0. So,

$$\text{Blast_radius}(P) = \frac{2}{2^{2 \cdot 0 + 1}} = 1.0$$

Interpretation: A blast radius of 1.0 is the maximum possible in our model. This extremely high score indicates that the principal's permissions span the entire organizational boundary at the highest risk level for data exfiltration and forgery.

References

- [1] Microsoft, *Azure role-based access control (RBAC)*, <https://learn.microsoft.com/en-us/azure/role-based-access-control/overview>.
- [2] Christophe Parisel, *Azure Silhouette, a SPN sorter and roles minimizer*, <https://github.com/labyrinthinesecurity/silhouette>.
- [3] Etienne Ghys, *A Singular Mathematical Promenade*, page 37, <https://arxiv.org/abs/1612.06373>
- [4] Richard A. Mollin, *An Introduction to Cryptography (Discrete Mathematical and Applications)*, Chapman and Hall/CRC; 1 edition (August 10, 2000), ISBN 1-58488-127-5.