Damage control in Azure Data Plane

Measuring the blast radius of users and NHIs

Author: Christophe PARISEL

Version 1.0.2

Introduction

While control plane risks in cloud IAM have been using the WAR norm[1], data plane permissions lack an analogous quantitative measure of their risk spread. In this article, we propose a model that quantifies the *blast radius* of a principal's data plane permissions and let score principals by order of risk. Our method leverages the strong triangle inequality inherent in ultrametric spaces to identify the maximum separation among any principal's permissions. This metric ranges from 0.0 (infinitely small radius) to 1.0 (tenant wide radius). We then compare our metric with WAR's, clarifying their respective roles in Azure data and control plane risks assessment.

Problem statement

1. Problem Statement on the Data Plane

1.1 Data Exfiltration (Confidentiality)

Unauthorized data access via `dataActions` read permissions enables a principal to extract sensitive information. If a principal's permissions span independent organizational zones, the potential for exfiltration increases significantly.

1.2 Data Forgery (Integrity)

Similarly, write `dataActions` permissions allow a principal (if compromised) to modify, overwrite, or forge malicious or fraudulent data, posing direct risks to data integrity.

1.3 Prioritization of 'dataActions' Permission Types

Azure Data Plane permissions feature 4 main types:

- Wildcards, granting all data plane actions to the resource scoped by the assigned role
- Write, granting the capability to create, delete, or edit customer data
- Read, giving access to customer data
- Action, typically used for listing customer objects (like blobs, key names, or containers)

Our model emphasizes that:

1. Both read and write `dataActions` are high-risk because they directly impact data confidentiality and integrity.
2. The combination of read and write `dataActions` represents a higher risk than read or write alone, since both data exfiltration and data forgery attack scenarios can be exploited simultaneously.
3. The action `dataActions` is a lower risk, thus not accounted in the blast radius scoring.
4. The risk of wildcard `dataActions` is equivalent to read AND write permissions, since the risk of data theft or data manipulation is already maximum.

1.4 The key importance of Organizational Context

We collapse scope assignments below the subscription level, as the risk for customer business data is effectively the same whether a principal is granted access at, for example, a database level holding customer data or at the whole resource group or the whole subscription level.

However, the Management Group (MG) hierarchy is strictly preserved because it captures key organizational boundaries aligning with most customer's data privacy and integrity concerns:

- The production versus non-production boundary is often reflected in customer's MG hierarchies. They directly impact the data exfiltration and forgery risk levels.
- The geography (regional boundaries) delineates regulatory data hosting and processing requirements. When implemented by MG, they directly impact the data exfiltration and forgery risk sensitivities.
- Business units (BU) have their own rules and regulations (called 'Chinese Walls' in the finance industry). Nearly all customer model BU and subdivisions as nested MG. This model directly impacts and localizes the data exfiltration and forgery risk appetite.

2 Comparison with Control Plane Needs

2.1 Control Plane Context

Scoping

Unlike in the Data plane, the impact of control plane operations can be significant at "deep" resource level scopes: therefore, the WAR metrics considers all permissions down to the minutest resource and sub-resource levels.

Conversely, the impact of control Plane operations tends to even out as scopes widen because they reach a critical mass: for that reason, the WAR metrics collapses all MG into one single level. (Causing configuration harm to a single MG is very similar to causing configuration harm to several MG in the Tenant).

Permission types

In WAR, `Actions` permission types obey the following ordering:

'Write' > 'Action' > 'Read'

reflecting the takeover and escalation potential of the 'Write' type and the marginal importance of the 'Read type' in causing configuration harm.

What's more, in the Control Plane, wildcard `Actions` lead to arbitrary operations which cannot be fully grasped by a risk assessment and are subject to future changes. Therefore, at subscription level or higher, wildcards are mightier than 'Write' permissions:

'*****' > 'Write' > 'Action' > 'Read'

2.2 Why a Separate Data Plane Metric?

Data-Centric Focus: Our blast radius concept isolates the direct impact on confidentiality and integrity. Control plane metrics focus more on configuration errors leading to accidental endpoints exposure or to security guarantees relaxation (eg: disabling encryption at rest, audit logs, …).

Scope Modeling: We collapse resource-level details into subscription-level boundaries while preserving the significant organizational structure provided by Management Groups.

Permissions ordering: compare the WAR permissions ordering

'*' > 'Write' > 'Action' > 'Read'

with our metric:

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

Methodology: Whereas WAR measures superincreasing differences between principals, the blast radius is defined within a single principal's set of data permissions.

Risk Implication: a high data plane blast radius indicates that a principal's data access spans multiple organizational domains—hence posing a significant risk of data exfiltration or forgery. This is quite different from configuration errors in the Control Plane.

3. Scoring the blast radius of an Azure Principal

3.1 The Distance Model

We leverage the natural ultrametric distance between two `dataActions` permissions p1 and p2 in the Azure Tenant hierarchy [2]:

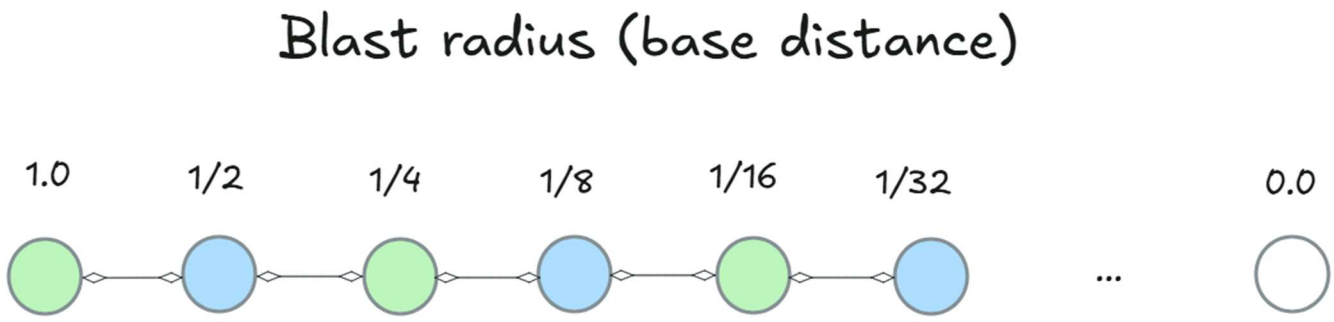
- Depth 0: Tenant (Root)
- Depth 1+: Management Groups (MGs), up to 6 officially
- Final depth "d": Subscription (treated as the leaf, with all sub-resources collapsed upward)

So depth ranges from 0 to a maximum of 7.

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

Base_distance(s1,s2)= $\frac{1}{2^{2d+1}}$

The base distance ranges from 0.0 to 1.0 It defines measures (in blue) and holes (in green) in the $\frac{1}{2^n}$ series: measures appear at odd powers in the series, and holes at even ones:



3.2 Weighting by Permission Type

To reflect the permissions ordering that we set in paragraph 2.1, we *ignore all permission pairs containing an 'Action'* [3]

For the remaining pairs, we define an 'impact' coefficient:

Impact = 2 if the pair contains a wildcard at LCA depth, or if both 'Write' and 'Read' atomic permissions are assigned to the principal at LCA depth by two pairs scoped at s1 and s2.

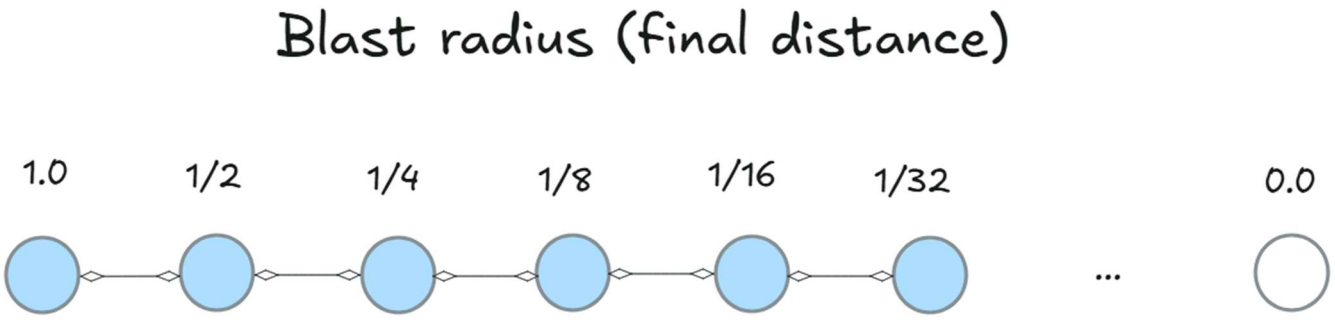
Impact =1 if the 'Read' or 'Write' permission is assigned to the principal at LCA depth.

The final ultrametric distance δ is:

$$\delta(s1,s2) = \frac{\text{impact}}{2^{2d+1}}$$

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

This nuance defines a hierarchy within the hierarchy: it "fills in" the $\frac{1}{2^n}$ series holes in an orderly fashion: for identical scopes, high risk permissions at even locations have a higher measure than lower risk ones at odd locations:



3.3 Defining Blast Radius

We leverage a useful property of ultrametric distances to define the blast radius: the strong triangle inequality

distance(a,c) ≤ max(distance(a,b),distance(b,c))

We define the blast radius of a principal as *the maximum ultrametric distance between any two data plane permissions they hold.* This distance represents the worst-case spread of their ability to exfiltrate or corrupt data:

For a given principal P with given permissions {p1, … pi, …, pj, …, pn}

Blast Radius(P) = max δ(pi,pj) for all i and j

This radius characterizes the full dispersion of the principal's data permissions. A higher blast radius indicates that the principal's access spans organizationally diverse domains and thus represents a higher data risk.

Conclusion

We have introduced an ultrametric-based model for quantifying the blast radius 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.

References & notes

[1]: A theory of de-escalation, page 5, the WAR distance
https://github.com/labyrinthinesecurity/automatedReasoning/blob/main/controls/IAM/DeEscalation.pdf

[2]: Via Nebulosa newsletter issue 52: https://www.linkedin.com/pulse/adversarial-lateral-motion-azure-paas-we-prepared-christophe-parisel-hyrpe/

[3]: 'Action' can help adversarial reconnaissance in the data plane. To monitor this risk and assess its blast radius, a possible solution is to implement a dedicated ultrametric distance. Therefore, we don't mix it up with the data exfiltration and data forgery concerns, which should remain the prime focus.

Appendix: Blast radii examples

Example 1: Single Permission at the Management Group Level

Scenario: a principal is granted a single 'Write' data plane permission that is scoped at management group level. The MG sits at depth 3 (in our collapsed tree: Tenant is depth 0, MGs follow, and Subscription is the leaf).

Calculation: With only one permission, there is no pair to compare and no LCA. The permission is 'Write', so the impact coefficient is 1. This permission belongs to a MG cluster at depth 3, so the blast radius is confined within this cluster:

Blast_radius(P)= $\frac{1}{2^{2*3+1}}=\frac{1}{2^7}$=0.0078125

Example 2: Two Permissions Within the Same Management Group Branch

Scenario: a principal holds two data plane permissions:

- The existing 'Write' permission in the MG considered in example 1 (sitting at depth 3).
- A wildcard permission in a subscription sitting at depth 5 under the same MG branch (effectively still governed by the MG sitting at depth 3).

Calculation: Since both permissions share the same MG as their LCA, the LCA depth is the initial MG's depth d=3. The wildcard permission could entail impact=2, however this permission in not at LCA depth: it stands at depth 5, which is higher. Since the permission at LCA depth is only 'Read', impact is 1.

So, Blast_radius(P)= $\frac{1}{2^{2*3+1}}$=0.0078125

The blast radius is the same as in the previous example, which is not a surprise given that the newly added permission, while riskier, stands deeper (higher) into the same clustering hierarchy.

Example 3: Two Permissions Spanning Different Management Group Branches

Scenario: a principal holds:

- The existing 'Write' permission in the MG considered in example 1 (at depth 3).
- A 'Read' permission sitting at depth 2 in a different MG, called MG'.
- The two permission trees relate such that their Lowest Common Ancestor is MG' parent, sitting at depth 1.

Calculation: LCA is 1 and both permissions have an impact of 1.

So, Blast_radius(P)= $\frac{1}{2^{2*1+1}}=\frac{1}{2^3}$=0.125

The blast radius is rather large because the permitted resources stand far apart in the Tenant hierarchy.

Example 4: Replacing Read permission with Write+Read and moving to the Tenant root

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, they can be collapsed into a 'Write+Read' permission. Since this collapsed permission is at LCA depth, and since it represents a high risk, its impact is 2. The LCA between MG and MG' is 0.

So, Blast_radius(P)= $\frac{2}{2^{0*1+1}}=\frac{2}{2^1}=1.0$

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.