# Decision Point for Postmodern Security Zones

Published 19 November 2019 - ID G00451287 - 46 min read

By Analysts Joerg Fritsch

Initiatives: Security of Networks and Endpoints for Technical Professionals

Many organizations reproduce legacy designs — based on network segmentation paradigms — in new environments, leading to suboptimal architecture. Security and risk management technical professionals should use the model and architecture examples in this report to group and segregate digital assets.

## Overview

### Key Findings

- A security zone is defined by a combination of the service type and its communication relations, the identity of the people using the service, and the sensitivity of the data that is processed. Minimum or arbitrary maximum zone size constraints do not make sense from compliance, business or risk perspectives.
- The number of security zones implemented must be appropriate for the risk that you are trying to mitigate, and the tools used to manage those risks. In traditional architectures, the “sweet spot” range is five to 20 security zones. In microsegmentation architectures, you can go much higher.
- While traditional zone enforcement technologies (ZETs) segment your network tier by tier, microsegmentation technologies segment your network application by application. The best products offer the capability to do both at the same time and let you retain traditional paradigms.
- In modern software-defined data centers, firewalls are only one of many ZETs. Examples of other ZETs include identities and entitlements, as well as hypervisor-based segregation.

### Recommendations

Technical professional engaged in network security should:

- Strengthen agility by adopting a zone definition model that blends with DevSecOps and DevOps. Use automated workflows to streamline processes that document and approve rule changes.
- Establish a clear human-defined segmentation goal, and let tools help you to achieve or fortify it. Manually define five to 20 major security zones, and use automated rule builders and machine learning tools to add microsegmentation.
- Plan for newer architectures, such as hybrid IT multicloud hexagonal architectures, and do not overrate the importance of network segmentation in such architectures.
- Leverage your software-defined data center like a best-in-class technology company by relying on software-defined ZETs such as hypervisors, data partitioning and identity management. Traditional firewalls will soon become obstacles rather than helpers.

## Decision Point Question

Which security zones should we utilize to protect our enterprise?

## Decision Overview

Security zones are fundamental to reducing the attack surface of infrastructure and application assets. They do so by enforcing policies for, and creating visibility into, interasset network communications. But deciding on the number and size of zones, as well as their asset contents, is not a trivial undertaking.

- Too few zones lead to large segments with an unclear role, coarse security controls and an increased risk of lateral damage in the event of a compromise.
- Too many zones are difficult to manage and can slow network-related changes and/or add significant cost.

In traditional security zoning, the available enforcement technology options — such as firewalls — drive many organizations to want to minimize the number of zones. Microsegmentation-based [1] approaches offer enhanced flexibility and protection, but the necessary architectures and best practices are still emerging.

Gartner observes that security zoning for modern application architectures cannot be described with traditional network security zoning paradigms. Many applications are hugely interconnected, without necessarily having their own presentation or persistence layer. Some companies' "compute farms" (which are largely stateless entities with no presentation layer) are larger than the rest of their computing estate combined. Applications that enable digital business and the algorithmic economy will consist of various entry points for streaming data and have many tiers of processing, as well as channels to make automated (algorithmic) decisions and to provide data to partners. The application

architecture, or tiering, is built around the business function as opposed to any notion of a perimeter-based security model that reflects the capabilities and limitations of the security controls.

To address this, this Decision Point enables security architects to define and evolve security zones using a consistent model that:

- Allows more granularity than familiar architectures
- Is backward-compatible with traditional standards
- Easily adapts to modern application architectures and paradigms

## Traditional Security Zones

Gartner defines traditional security zones as network security zones that are configured with technologies that require network traffic to be “steered” across them. This includes next-generation firewalls in routing mode, bridging mode or virtual wire mode, but also access lists on switches or routers.

It has been a best practice in the past to make one-to-one mappings of business logic and application logic to the physical infrastructure design and the capabilities (or limitations) of firewalls. Logical security zones overcome this. Although traditional firewalls may still play an important role for macrosegmentation, clients can now use new technologies to map many logical zones to the small number of physical zones.

## Logical Security Zones

Gartner defines logical security zones as security zones that are determined by business structures and processes or application architecture that can be configured without changes to the underlying physical infrastructure design or system configurations.

The concept of logical security zones has been around for years, but implementations have not gone beyond traditional architectures that mimic the capabilities and layout of a legacy physical infrastructure. Rampant data breaches have provided the imperative for action. Many enterprises already implement logical security zones in their virtualized data centers and digital assets.

Two key technologies — frequently combined — are in use to create logical security zones:

- **Microsegmentation technologies:** Microsegmentation creates network security zones that do not address segregation of components, such as storage, compute or databases.
- **Software-defined security zones:** Software-defined security zones do not stop at the network level, but segregate storage, compute and networking on the same hardware pool.

### Microsegmentation Technologies

Gartner defines “microsegmentation” as the ability to insert a security service into the access layer between any two workloads in the same broadcast domain or x86 host.

Microsegmentation provides visibility and control over east-west traffic. It can also increase security and transform the way networks are segmented — from the traditional coarse-grained, static approach into a granular, dynamic application of a security policy that is, for example, based on security tags. Microsegmentation can create logical security zones without changes in the physical infrastructure design — for example, in virtualized environments or computing clouds — without the need to change system Internet Protocol (IP) addresses, virtual LANs (VLANs) or default gateways. That ability is especially useful in virtualized environments.

For more information about microsegmentation, see “Architectures and Paradigms of Microsegmentation Products” and “Solution Comparison for Microsegmentation Products.”

### Software-Defined Security Zones

Software-defined security zones are segregated by zone enforcement technologies that deploy and implement the required security policy across all levels of a software-defined (or virtualized) data center (illustrated in Figure 1). The ZET may involve traffic routing, such as firewalls, but it does not need to. For example, computing clouds require ZETs that are not based on traffic routing because the emulation of legacy experiences, such as a large number of subnets controlled by a traditional firewall, would be unmanageable. For example, software as a service (SaaS) solutions are multitenant applications that mainly use application layer or data layer ZETs to create the illusion of a single instance of a single-tenant application, thereby achieving unprecedented scalability and efficacy. Traditional network-based ZETs have little relevance in SaaS environments, illustrating the importance of logical ZETs that are not network firewalls.

Table 1 lists controls that are frequently deployed as ZETs.

**Table 1: Selected Zone Enforcement Technologies**

| Layer | Example |
|---|---|
| All | Surveillance and monitoring |
| All | Cloud-based account or subscription management (illustrated in Figure 2), such as Amazon Web Services (AWS) Control Tower, AWS Landing Zone or Microsoft Azure Policy (With cloud service providers (CSPs) pushing for accounts or subscription to be your perimeter, in public clouds, this ZET has quickly become equally important to traditional IP-based ZETs.) |

| Layer ↓ | Example ↓ |
|---|---|
| Network | Multiprotocol Label Switching (MPLS), Ethernet VPN (eVPN), VLAN, software-defined network (SDN) overlays such as Virtual Extensible LAN (VXLAN) and CSP security groups |
| System | Hypervisors and host firewalls |
| Protocol | Traditional perimeterization technologies that involve traffic routing (the "good old" firewall) — ranging from ports and protocols to protocol inspection and awareness |
| Application | Containers and authentication and authorization (authNZ) — for example, as built into databases (RBAC) and multitenant applications, such as SaaS solutions |
| Data | authNZ, RBAC, encryption and enterprise digital rights management (EDRM) |

Source: Gartner (November 2019)

Gartner observes that, in virtualized architectures, best-in-class enterprises have started using a mix of ZETs (illustrated in Figure 1) to segregate the compute, data and network layers that belong to an application, thereby creating logical security zones. For example, some clients leverage hyperconverged infrastructure and hypervisors to zone applications or tenants. Other clients may have data analytics requirements where relying on traditional ZETs, such as enterprise firewalls (aka next generation firewalls [NGFWs]), for security zoning is problematic. For example, they will need to duplicate the database infrastructure (including the amount of data and network traffic for ingesting data) to make the data available in several perimeter-based traditional network security zones. For more information about data-centric use cases, see Gartner's "How to Successfully Design and Implement a Data-Centric Security Architecture."

Figure 1 illustrates how software-defined data centers (SDDCs) enable the creation of cross-cutting logical security zones that are not limited to network separation. Pooled hardware exposes the software-defined storage, compute and networking components. The segmentation of applications is provided by software-defined controls that are part of the software-defined component, such as:

- Network microsegmentation
- Hypervisor isolation of instances and accounts
- Storage, such as virtual SAN (vSAN)

Although IP-based segmentation prevails for many use cases, CSPs recommend identity-based security zones to their clients, and also provide the tools to configure, automate and maintain the

implementation of identity-based security zones.

Figure 2 illustrates the initial state of an AWS Landing Zone environment consisting of several AWS accounts. IP-based controls, such as microsegmentation, can be used to control east-west traffic or to create subzones inside the accounts.

Figure 1. Implementation of Software-Defined Security Zones in an SDDC

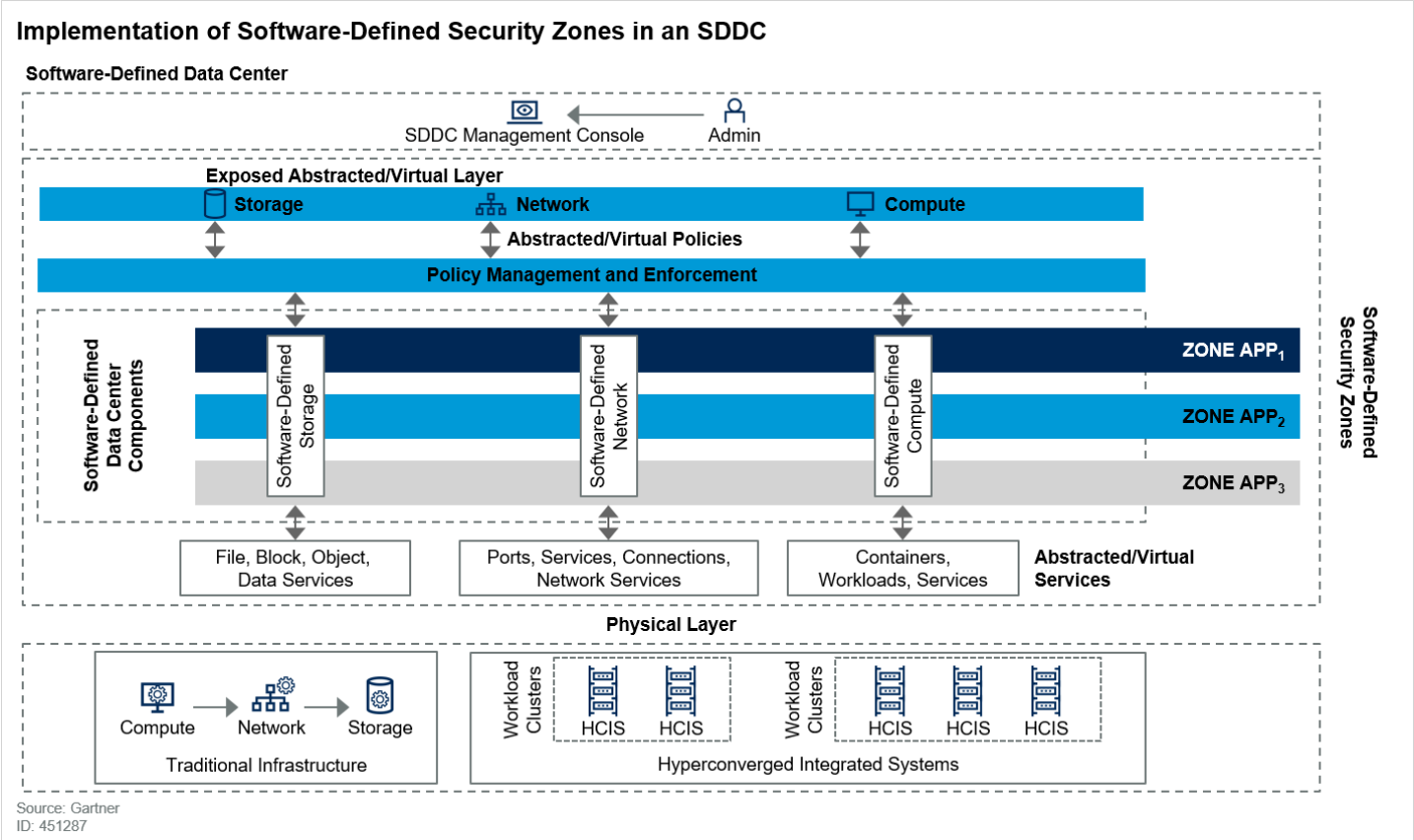
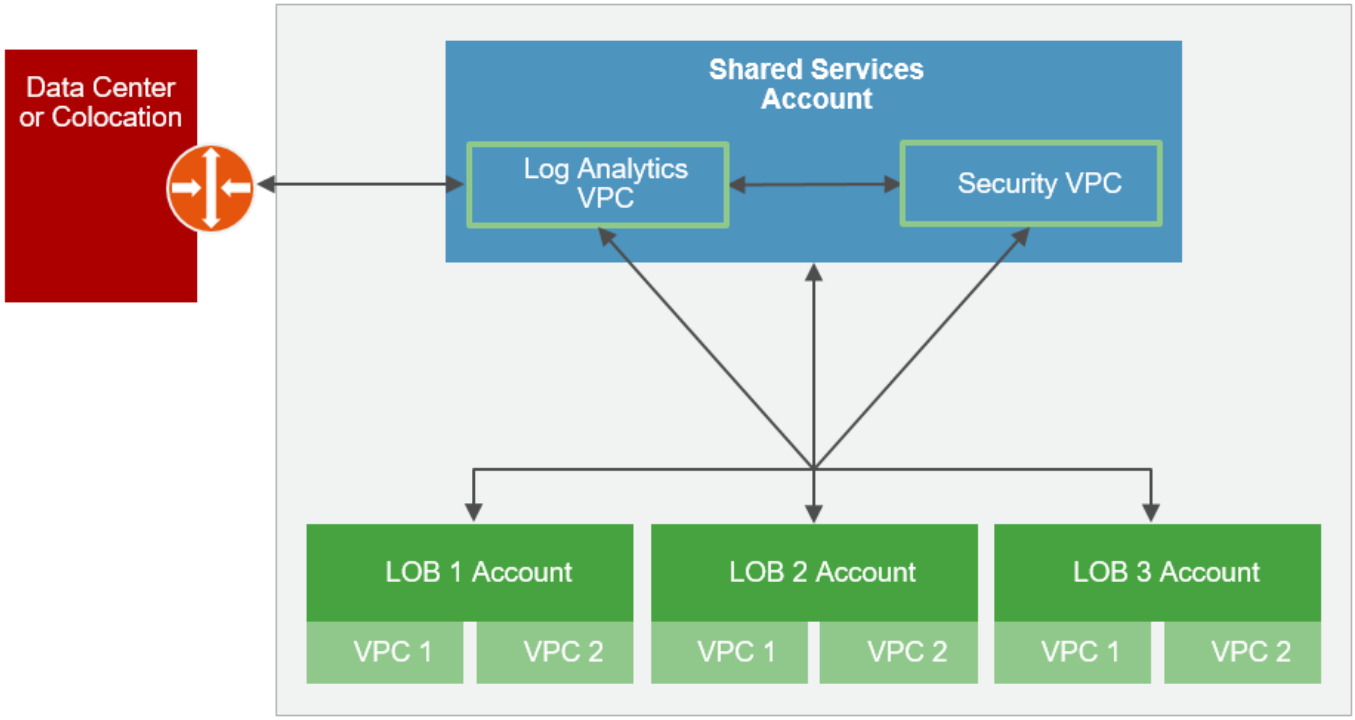



Figure 2. Multiaccount Infrastructure Leveraged for Segmentation

**Multiaccount Infrastructure Leveraged for Segmentation**




LOB = line of business (such as development, production and operations)
Source: Gartner
ID: 451287

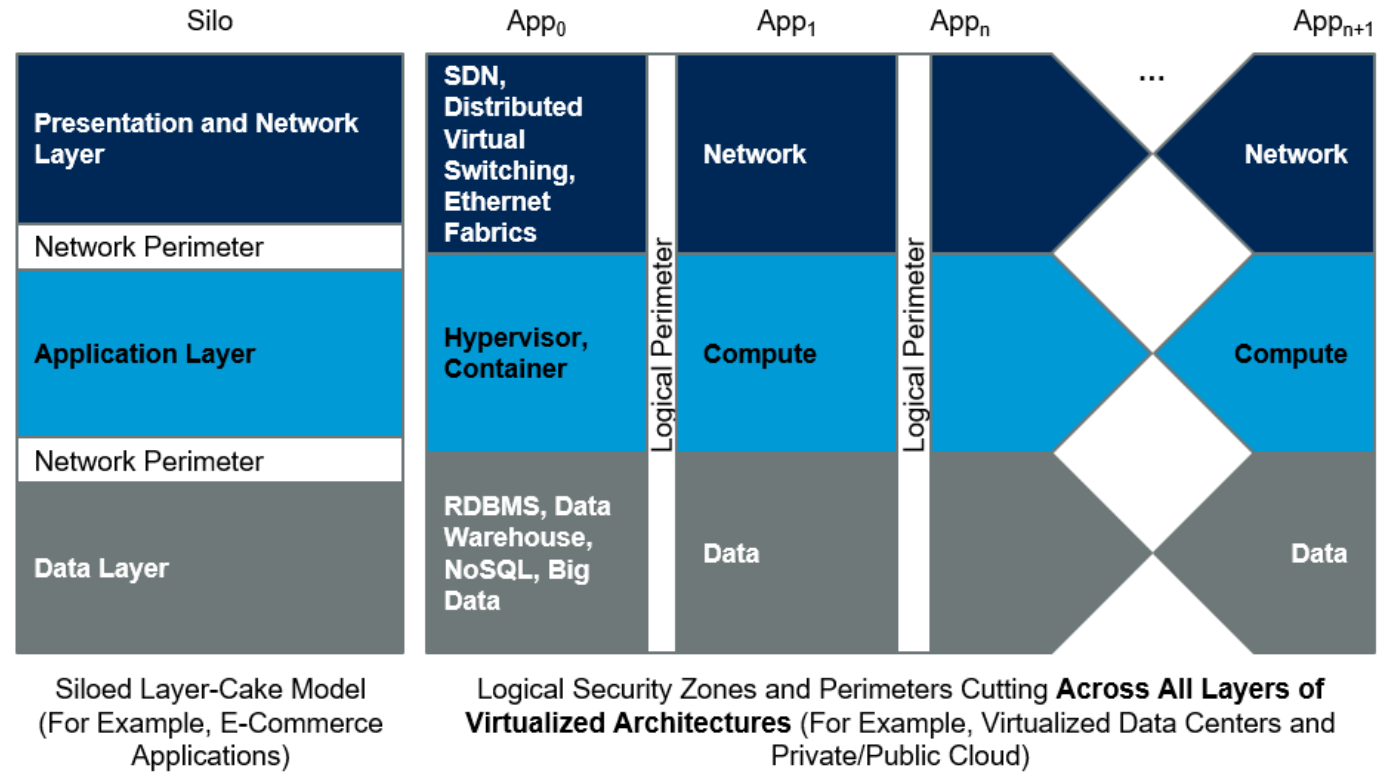
### Tier-by-Tier vs. Application-by-Application Security Zoning

The architecture and layout of security zones is at a turning point, from zoning monolithic applications tier by tier to zoning newer distributed applications on an application-by-application basis.

#### Tier by Tier

Applications are traditionally designed as monolithic three-tier silos featuring a presentation tier, an application tier and a data tier (illustrated on the left side of Figure 3). Network firewalls were configured to physically ensure that tiers can only interact with the clearly defined interfaces on servers in other tiers. The separation between these tiers is not necessarily done because of security, but so that the rule base of traditional firewalls stays understandable and manageable. For example, it is easier to detect possible attacks if the tiers interact in defined ways only, and changes within one layer, such as a database cluster, require only small changes to the network perimeter.

Figure 3. Evolution of the Perimeter




### Application by Application

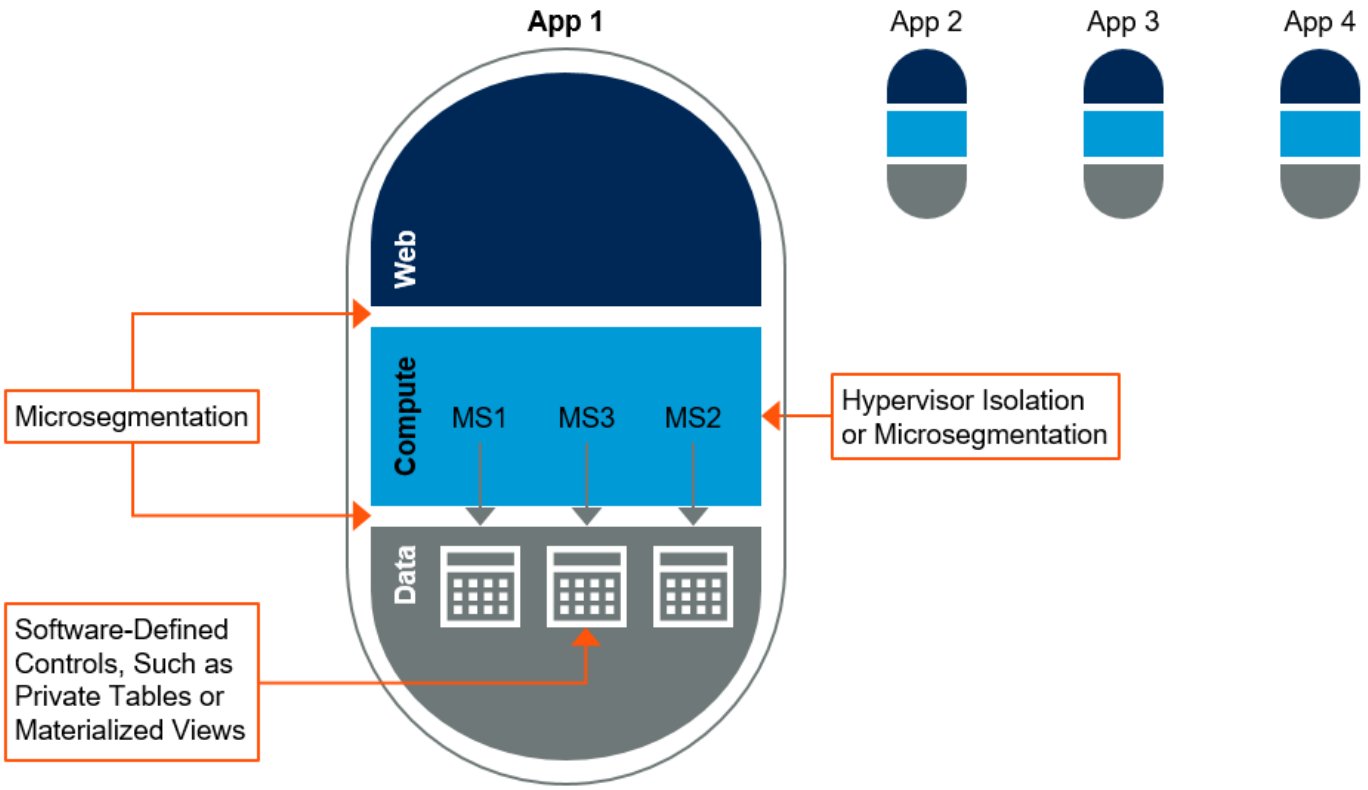
Tier-based security zoning was the prevailing paradigm when applications were monolithic and *rarely updated*, and they needed to interact with only one kind of entity at either the presentation layer or the data layer. The drawback is that newer applications, such as microservice-based applications, are distributed, and changes to the required communication paths may occur at any of the next *incremental updates,* which come at high pace.

Products that enable logical security zoning are designed to segregate application by application (illustrated on the right side of Figure 3). This certainly helps the security for distributed applications, especially because products are capable of providing increased visibility on the required communication flows inside traditional tiers.

However, Gartner clients frequently want or need to retain tier-based zoning because they find it difficult to buy into a new paradigm that still has to prove itself and that has not been exposed to the field for long. For critical systems and architectures, clients frequently combine several ZETs and retain tier-based zoning while adding application-based zoning (illustrated in Figure 4).

**Figure 4. Combination of ZETs to Retain Tier-Based Zoning While Adding Application-Based Zoning to It**

**Combination of ZETs to Retain Tier-Based Zoning While Adding Application-Based Zoning to It**




Source: Gartner
ID: 451287

### Reality Check: Do You Have a Case for Software-Defined Security Zones?

Although the goal to stand up an SDDC where only software-defined security zones are used is ambitious, you will probably discover that you are already using modern ZETs for selected use cases.

Gartner observes that clients are frequently running a small number of pockets that have SDDC functionality, but they do not have a complete SDDC. For example, clients deploy hyperconverged integrated systems (HCIS) by application I/O profiles — virtual desktop infrastructure clusters, file share clusters or database clusters to name a few — to optimize clusters for specific applications. They do so because mixing diverse workloads will degrade overall performance. This way, there will be “natural” boundaries and security zones forming around applications that are deployed on SDDC technologies.

What is missing is a unified software-defined management plane that would bring security zoning into a better position to support the rapidly changing needs of digital business, and to provide effective protection in a rapidly changing threat environment. For comparison, public cloud offerings are frequently specific to application I/O or compute requirements (for example, block storage, object storage or graphics processing unit [GPU] computing) and delivered from specialized hardware, but they do have a unified management plane.

## Decision Tool

Gartner defines a security zone as a *logical* grouping of IT resources that may reside at multiple locations but have similar protection requirements. Systems are grouped into zones by three attributes:

- **Service — types of (network) services:** This is the communication relationship to other services, applications and zones as required by the application, the service or the architecture that is needed by the application/service "owner." In the context of zones, services have traditionally been defined by network ports. This grouping simplifies policy management because traditional firewall rules were defined by network ports and addresses. Modern ZETs, such as microsegmentation products, define services at the application layer rather than the network port. Service groups should be aligned with, for example, the IT service catalog or the configuration management database (CMDB) where available. As such, we suggest the following basic service groups as a starting point for zone enumeration:
  - Web
  - Databases
  - Infrastructure servers
  - None/workstations

- **Identity — types of users accessing the systems:** Identity encompasses the business requirements and structure (for example, community of interest [COI], location, access type or function). User roles have always been a consideration when defining security zones. Trusted users can be granted privileges appropriate to the trust afforded by their role. Conversely, untrusted users must be excluded from restricted systems. The granularity of user roles varies by organization and often changes over time.

  However, technical professionals need to open their minds and expand their views of identities and interactions to Identity of Things (IDoT). IDoT brings the identities of systems, devices and an expanded view of interactions with applications as an additional level of complexity. Planners can utilize formal business-defined identity roles, or they can develop a hybrid that best balances the benefits of increased segmentation with implementation limitations that are specific to the business.

### Gartner recommends that clients use their Active Directory (AD) or Linux IPA structure to find the most appropriate ID_class. For example, you can use AD

**groups, organizational units (OUs) or containers as starting points to structure your ID_class and Service_class. CSPs frequently assist with the creation of (default) identity-based zones according to their best practices. Examples are AWS and Google Cloud Platform (GCP) landing zones.**

- **Data class — types of data on the system:** This is the sensitivity or classification of the data. Data can be sensitive for a number of reasons. For example, it can be confidential (such as personally identifiable information) or subject to integrity or availability concerns (such as data for operational technology [OT] environments). Security zones must enable data protection, prevent sensitive data from leaking, and ensure its integrity and availability. Therefore, Gartner leverages data classification as a component to create zones. (For more information about data classification, see "Improving Data Security Governance Using Classification Tools.")

  Enterprises that do not use data classification should consider the sensitivity of the data as a data class attribute. When possible, planners should exploit documented classification categories, but they should be cautious if there are more than three or four data classifications. Most organizations have no more than four data classifications, and too many classifications will greatly increase the number of zones. [2] Suggested starting data classification groups are:

  - Public
  - Internal use only
  - Confidential
  - Regulated

A security zone is formally defined by the triplet (the set of the three attributes) illustrated in Figure 5.

Figure 5. Formal Zone Definition

**Formal Zone Definition**

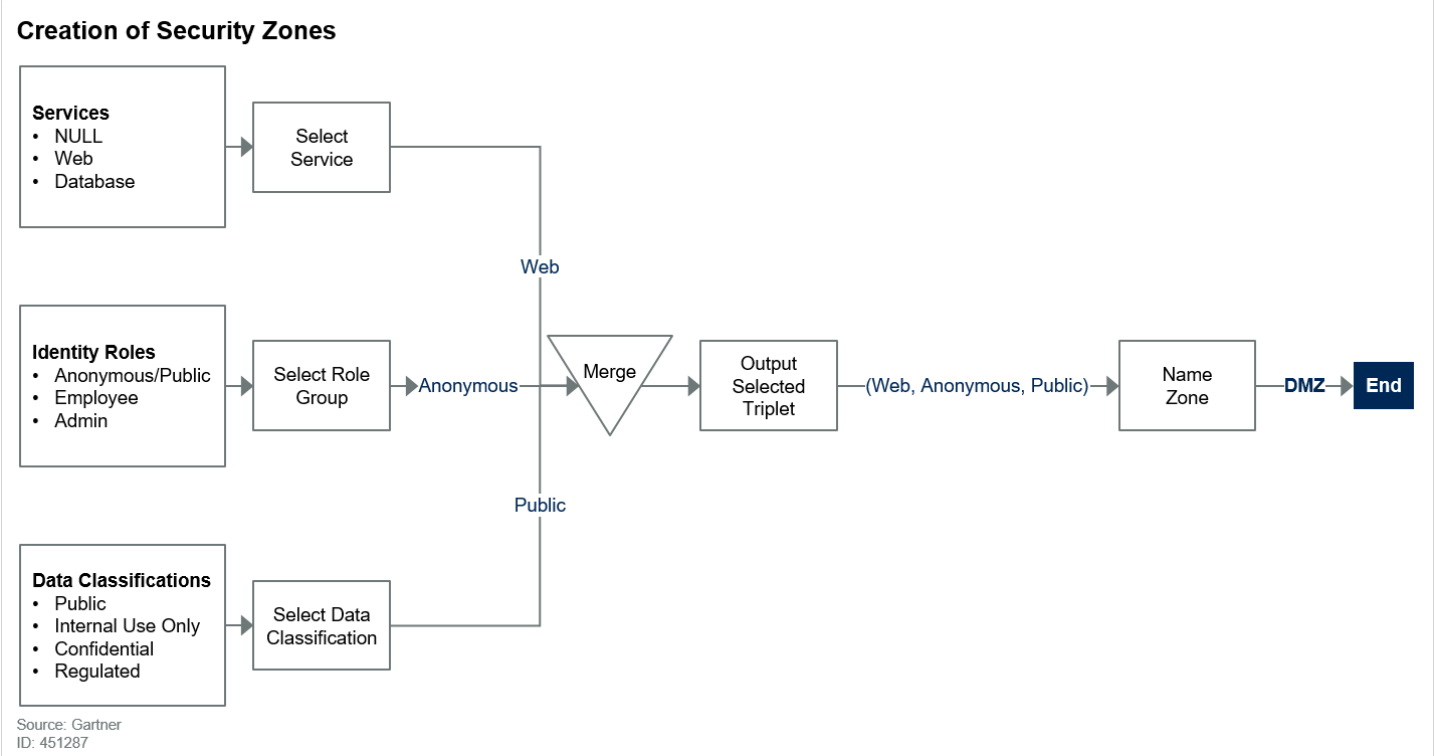
```
(service_class, MIN(id_class), data_class)
```

Source: Gartner
ID: 451287

These resulting zones implicitly group systems by the services running on the systems, the users who access the systems and the data processed on the systems. These attributes will drive zone creation and system placement. Utilizing these three attributes, this Decision Point provides a process to define logical network security zones and determine system placement. By adjusting the granularity of the groups, the zones can incrementally transition from existing legacy architectures or to microsegmentation.

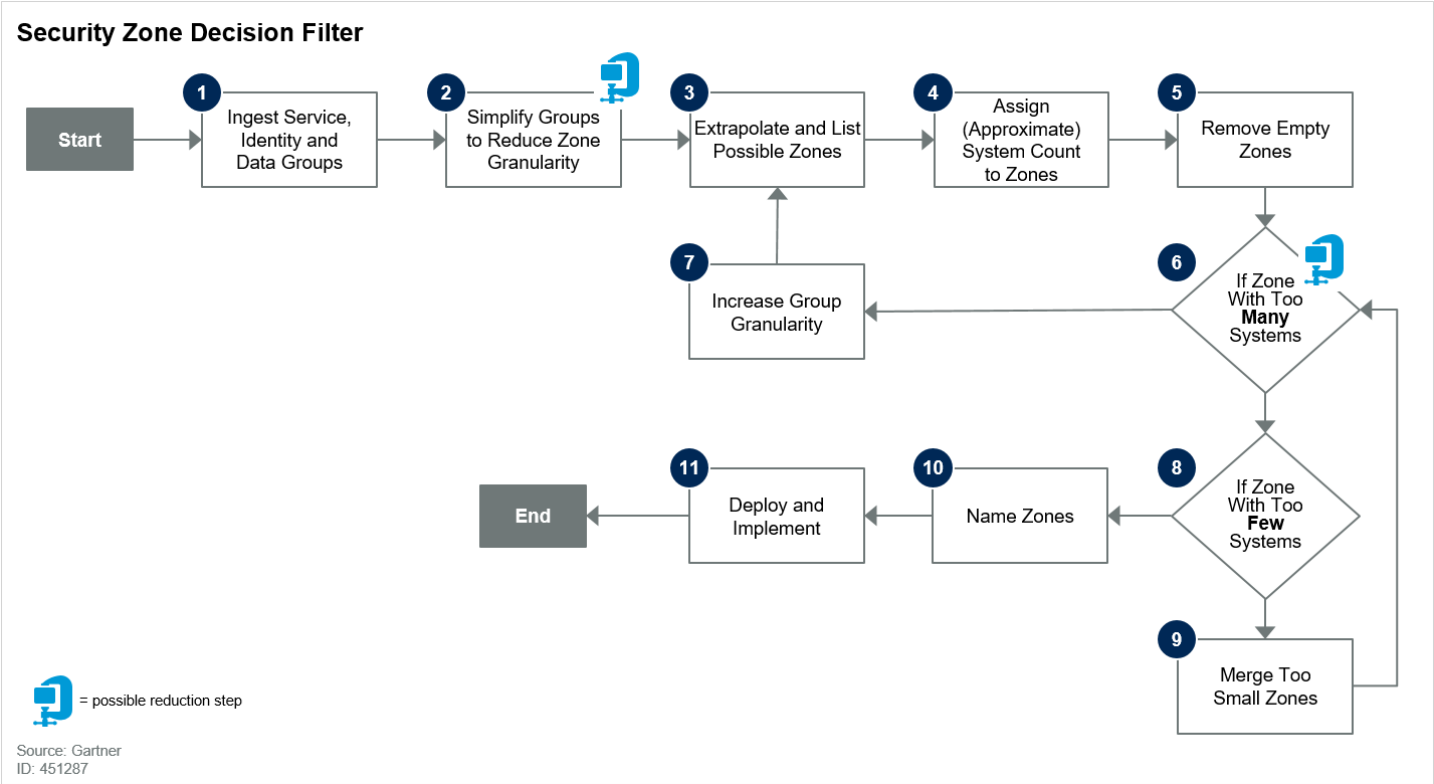
The flowchart in Figure 6 illustrates the creation of security zones (triplets) from the example attributes that are described above. For an implementation in pseudo code, see Note 1 at the end of this document. The flowchart in Figure 7 illustrates how to validate, verify and reduce a potentially large number of zones (Step 2 and Step 6). For an implementation in JavaScript (JS) that can be run (for example, with JS.do [3]), see the example provided at the end of Note 1. More detail on how to evaluate the reduction steps is provided in the Requirements and Constraints section.

Figure 6. Creation of Security Zones




Because this enumeration potentially delivers a large number of security zones (candidates), Gartner recommends not trying to invent any aptronyms (that is, telling names) but simply to use the composition of the triplet together with a delimiter that is accepted by your ZETs. For example, a traditional demilitarized zone (DMZ) that is described by the triplet ( `web server` , `anonymous` , `public` ) may get the name `web_server:anonymous:public` (see Figure 7).

Figure 7. Security Zone Decision Filter




A brief description of each step in the flowchart:

- Step 1 and 2: Business definitions and protection requirements must drive security. Utilize these business definitions as a starting point to begin defining network security zones.

  Security controls might not be appropriate or possible in all cases. Group definitions should be modified to fit this scope. Roles for purposes of security zoning are not necessarily the same ones that your identity and access management (IAM) or ERP system may use for RBAC and entitlement management purposes. Many clients have more roles in their IAM model than employees and would end up with a ridiculous number of triplets if they were to base this purely on their IAM roles. Thus, before the triplets are created, clients must determine a “sweet spot” of roles/identities that balances business alignment with security and zoning requirements.

- Step 3: Possible security zones are enumerated by extrapolating each unique set of identity, data classification and service groups. These sets can be represented as a triplet, such as “(web servers, untrusted users, public data).” Although this step reads as rather simplistic in practice, it will be a challenge to manually create all triplets. An example of Java code that will enumerate all triplets for your environment is provided in Note 1 at the end of this document.

- Step 4: Depending on the ZET, zones can be too small to merit their implementation, or they can be too large (which can aggregate too much risk). Assign approximate system counts to zones so that these factors can be considered. Gartner advises that clients should set zone size constraints only in traditional network architectures or government network architectures.

- Step 5: The enumeration process will likely create zones that are not necessary and that have no systems that would be placed into them. Remove these empty zones.
- Step 6: The deployment of too many systems into a zone might create too much risk. For example, risk may be aggregated because the systems are not protected from each other, thus allowing a worm or virus to spread across all systems in the network security zone without controls that can contain the attack. If the number of systems in any zone surpasses this threshold, the granularity of the groups must be refined and the possible zones re-enumerated.
- Step 7: Increase group granularity to increase the number of zones.
- Step 8: Zones with too few systems can be too costly to create and maintain.
- Step 9: When zones are too small to create or maintain, merge the two smallest zones with the most similar identity, data classification and service groups.
- Step 10: Many organizations name zones to enable easier communication.
- Step 11: Deploy and implement.

## Zone Size Constraints

The goal to introduce minimum or maximum zone sizes is understandable from the perspective of practicality. One example of this is the overhead and cost created by having too many zones in traditional zoning models with traditional ZETs, where a central team implements the zoning and policy definitions.

**The maximum number of feasible zones is largely a function of the ZET.**

Although traditional ZETs had a very tight sweet spot concerning the maximum number of feasible security zones, new ZETs frequently do not have these constraints, and large numbers of zones are feasible. For example, traditional ZETs infer that a central team implements the zoning and policy definitions. On the other hand, new ZETs are developed with an API-first paradigm in mind, thereby enabling self-service aspects by abstractions and the need for dynamic provisioning in today's cloud-based data centers.

Clients who need to limit the number of zones will find the practicality attribute in the pseudo code in the "zone reduction" portion of Note 1.

### Minimum Zone Size Constraints

Minimum zone size constraints do not make sense from a compliance, business or risk perspective. For example, HR typically has very limited IT deployments that require separation. These deployments are often implemented as separate zone environments, which are difficult for IT to monitor and manage. Traditionally, HR is a perfect candidate for zoning, yet it is very small. Another example is limited deployments covered by strict compliance requirements, such as Payment Card Industry Data Security Standard (PCI DSS; see also the Enclaves and Subzones sections).

#### Maximum Zone Size Constraints

Arbitrary maximum zone size constraints also make no sense. Gartner advises that clients should use maximum zone size constraints only to account for risk reduction (for example, to limit lateral spread after a successful compromise). Clients who use the Gartner formal model and find that some of the resulting zone sizes are too big should first note whether the services are differentiated enough.

For example, coarse service definitions, such as `web_server` , may lead to a result where many web servers are in the same zone. Granular definitions where the type of web server is used as input will lead to a better result.

If the intent is to separate production from development environments (a very common requirement, and both environments are potentially huge), then clients should be neither surprised at nor afraid of the resulting zone sizes.

## Principles, Requirements and Constraints

### Principles

The purpose of a security zone is to limit network communications and provide visibility.

Security zones:

- Reduce the risk of compromise by reducing the attack surface
- Reduce the scope of compliance (such as PCI DSS) by strictly segregating systems that are in scope of a regulation from systems that are out of scope
- Limit data exfiltration by reducing the number of data exfiltration paths and creating visibility on the network traffic
- Limit lateral spread in the event of a compromise by controlling east-west data center traffic (It must be noted that security zoning is not an appropriate defense against the spread of malware.)
- Enforce (or re-enforce) access privileges by prohibiting network communications from unauthorized sources

- Segregate environments (such as production and development) or multiple tenants

## Requirements and Constraints

Creating a large number of triplets and potential zones is interesting, but it gives you no information on how you can apply the Gartner formal model of digital security zones to your environment. This section looks at common and emerging network architectures and explains how you can adapt and apply the result of the decision tool.

The following architectures and paradigms determine if and how the number of security zones should be reduced:

- **Traditional architectures,** such as hub-and-spoke topologies with firewalls in the center
- **Government networks** built around a national security classification scheme
- **Hexagonal architectures,** [4] such as high-frequency trading (HFT) platforms
- **Container-based architectures,** which often use microsegmentation
- **Hybrid IT and cloud-based IaaS architectures**

Key considerations for each of these architectures are described in the sections below. If you are not familiar with a certain type of architecture, you are likely to find the most helpful information at the end of each section.

### Traditional Architectures

> **Gartner recommends that these clients rightsize the number of zones they deploy to a sweet spot that, for most enterprises, is between five and 20 manually defined zones.**

Clients should seek the minimum number of communities of interest or roles to limit the number of security zones (illustrated in Figure 8). The data classification attribute is mainly used to create enclaves, such as database servers that are in scope of PCI DSS, to exclude the remaining environment from the scope. Table 2 lists some example zones that most traditional enterprises will have (also see Figure 8).

**Figure 8. Zone Reduction in Traditional Networks**

**Zone Reduction in Traditional Networks**

*(service_class,* ***MIN(id_class),*** *data_class)*

Group zones according to data_class; create required additional zones according to MIN(id_class).

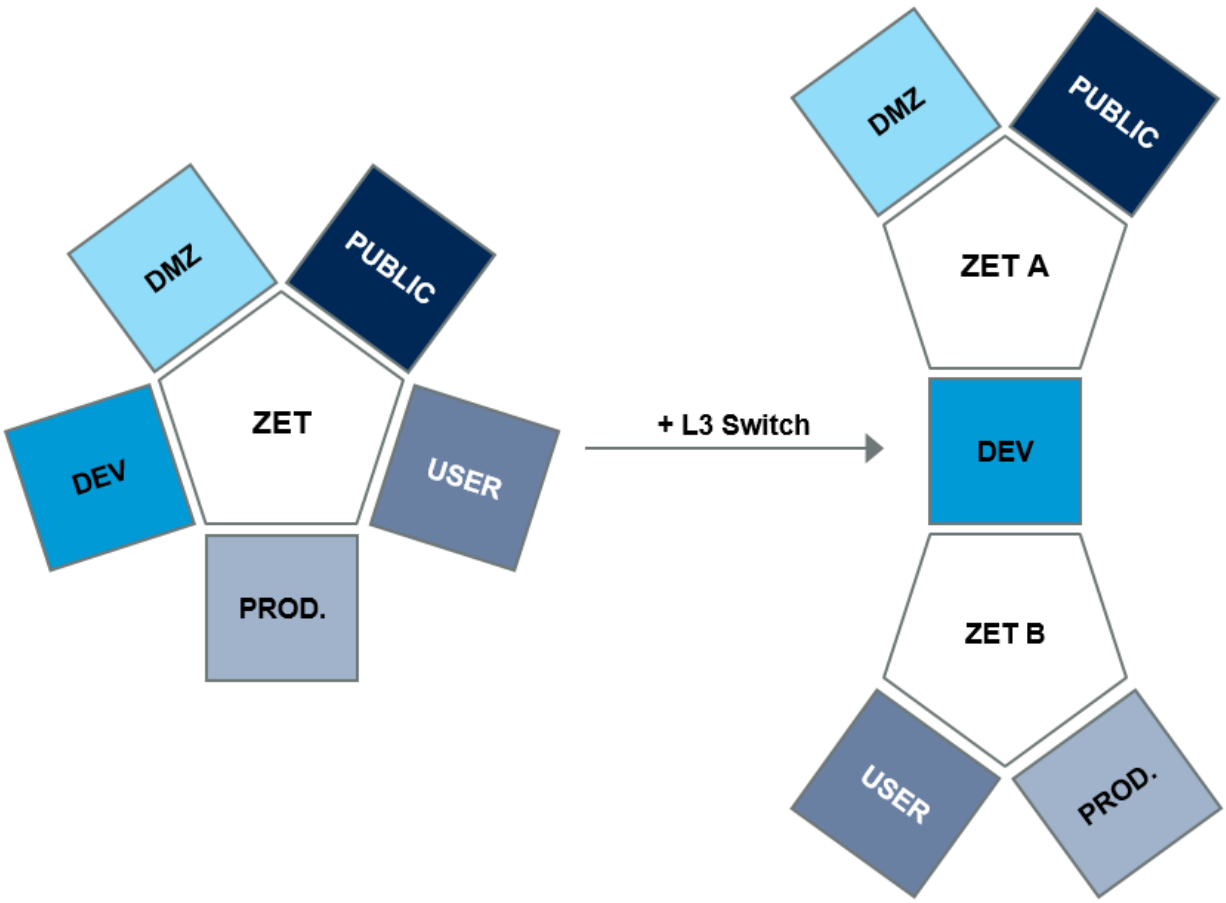
Source: Gartner
ID: 451287

### Traditional Network Properties

- The use of architecture and segregation is employed to reduce the attack surface and, where applicable, the scope of compliance.
- Development and production are frequently the main zones (in size).
- Security zoning frequently involves traffic routing via a common ZET — for example, an enterprise firewall that is reused for many purposes and that is an important (central) hub of the topology. See Figure 9 for a view of traditional enterprise network architecture.

**Figure 9. Traditional Enterprise Network Architecture**

**Traditional Enterprise Network Architecture**




Source: Gartner
ID: 451287

The concept of multiple parallel zones that are enforced by a central firewall (illustrated in Figure 9, left) has been in use at small and midsize businesses (SMBs) for more than 15 years. Depending on the enterprise size, the architecture is frequently implemented with an additional internal firewall (illustrated in Figure 9, right) where the development network is a zone between two firewall layers, which gives more flexibility to open communication requirements for developers.

The emergence of enterprise firewalls, with their many security features, has encouraged end users to reuse these expensive devices for as many security purposes as possible and has emphasized their position as central (security) hubs in traditional network architectures. Gartner recommends using traditional hub-and-spoke architectures in SMBs with up to 200 employees. Gartner recommends a combination of hub-and-spoke architectures with a layered architecture (illustrated in Figure 9, right) for midsize and large businesses, and businesses that have critical OT for industrial operations, such as manufacturing equipment, attached to their network.

As firms grow, the easiest way from the left to the right in Figure 9 is often to put the development network — where more lenient security policies are required — in the middle layer and introduce a routing capability between both layers. One way to do this is to use a Layer 3 network switch. This way, clients can uplift their architecture without the need to change IP addresses or systems.

## Table 2: Traditional Security Zones and Features

| Zone Name | Features and Triplet |
|---|---|
| Corporate users | Trusted business users on workstations without services: `workstation`, `employee`, `internal` |
| Development | Untrusted business systems running services for developers: `server`, `developer`, `none` |
| DMZ | Servers that run web services available to untrusted users: `web_server`, `anonymous`, `public` |
| Partner network | Services and data provided to business partners: `server`, `partner`, `internal` |
| Privileged users | Trusted business users on workstations with privileged access or restricted data: `workstation`, `admin`, `internal` |
| Production | Trusted business systems running services for trusted users: `database`, `employee`, `protected` |
| Public | Any system that is outside the traditional perimeter (for example, end users) |
| Regulated | Trusted business systems with data or services regulated by the credit card industry: `server`, `employee`, `pci` |
| Restricted servers | Trusted business systems running services for privileged users: `erp`, `employee`, `sensitive` |

Source: Gartner (November 2019)

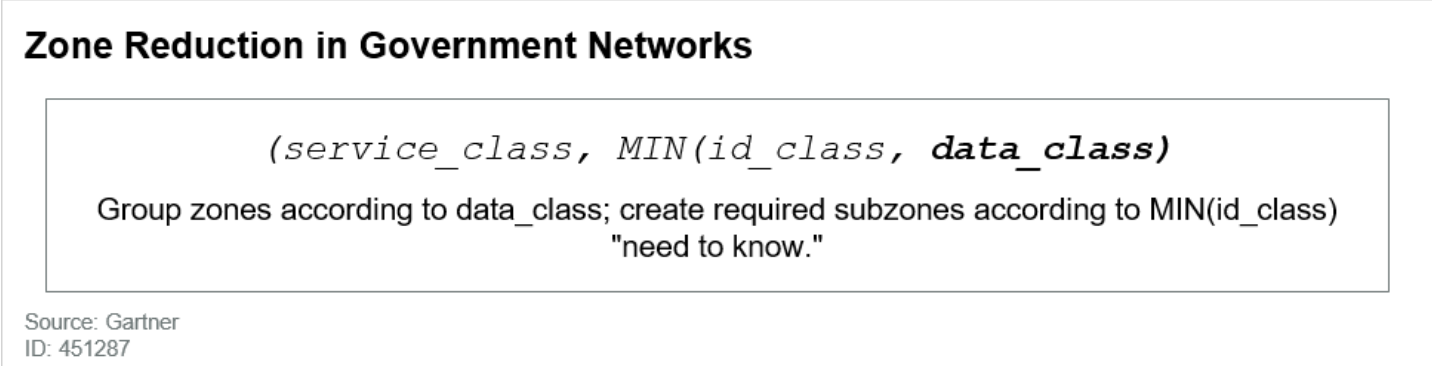
### Enclaves

Enclaves are defined in a number of standards, such as Committee on National Security Systems Instruction (CNSSI) 4009, U.S. Department of Defense (DoD) 8570.01 and several DoD Instruction (DoDI) standards. An enclave is generally defined as a "collection of computing environments connected by one or more internal networks under the control of a single authority and security policy, including personnel and physical security." [5] Strictly speaking, all traditional network security zones are enclaves. Enclaves can be network security zones or subzones (see the Subzones section). However, the term is mostly used for traditional network security zones that are created because of external expectations that must be met (for example, PCI DSS).

### Government Networks

Clients who have this type of architecture need to focus on the data classification attribute and the identity attribute (see Figure 10) that correspond to the "need to know" of defined COIs or roles that are present in the organization. Gartner recommends that clients expect the main security zones to grow very large. Clients should seek the minimum number of COIs or roles to limit the number of subzones (see the Subzones section) inside the main security zones.

Figure 10. Zone Reduction in Government Networks



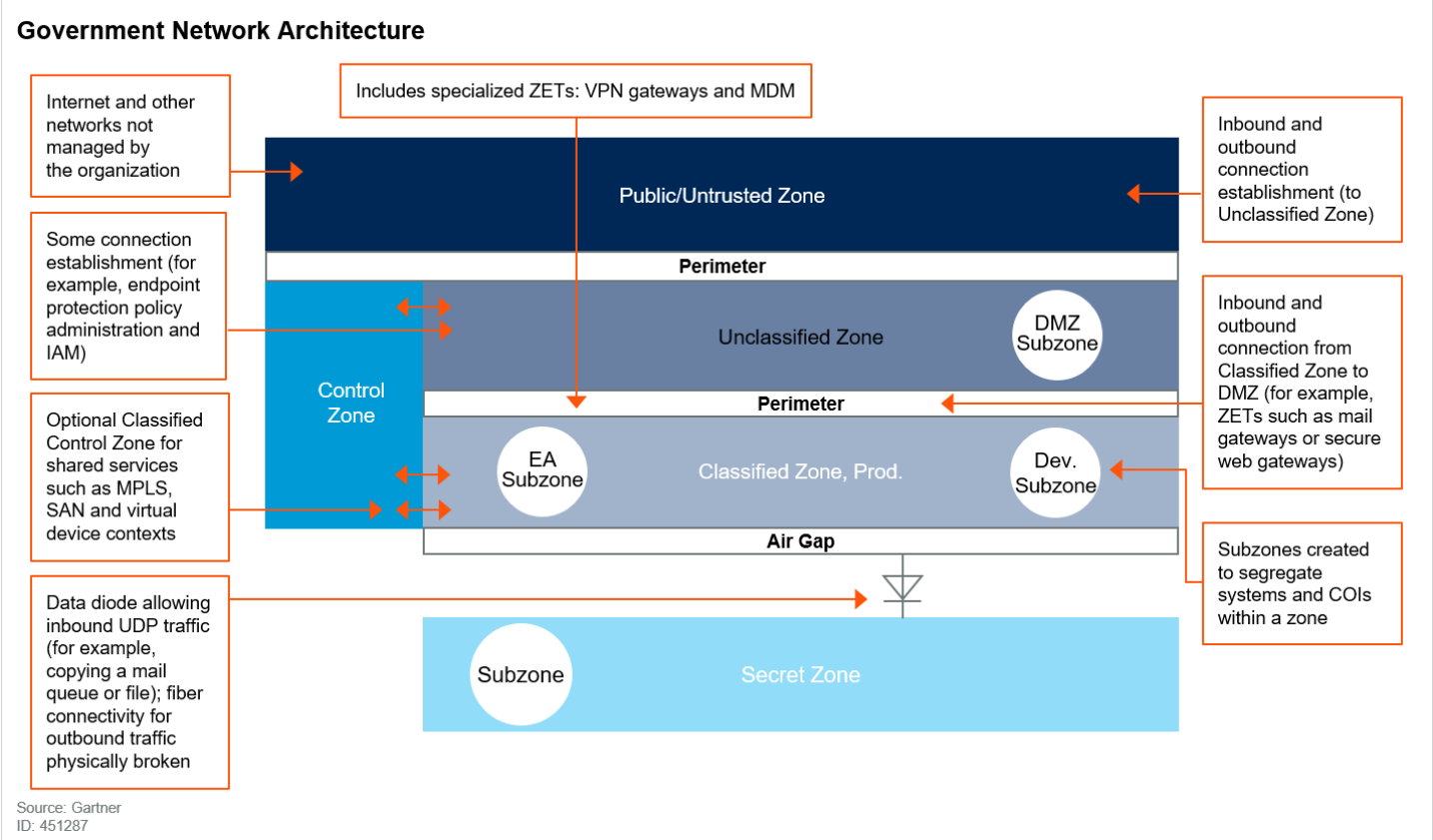

### Government Network Properties

- It is built around a defined classification scheme — not enterprise applications.
- A maximum zone size constraint makes no sense because all classifications are run on dedicated infrastructure that can potentially be huge.
- Classification comes first; identities ("need to know") come second.
- There is segregation of identities or applications via subzones.
- The government network prevents data exfiltration from security zones with a higher data classification into security zones with a lower data classification.

- The data classifications define the layers, or network security zones, each with a distinct perimeter (see Figure 9). The perimeter that surrounds each layer defines the network traffic that is allowed to pass into and out of the network security zone. The layers are loosely interconnected to allow essential business services, such as email/SMTP or web browsing/Hypertext Transfer Protocol (HTTP). Common ZETs are:
  - Commercial off-the-shelf NGFWs
  - Information exchange gateways (IEGs) that are a combination of proxy and filtering technologies tailored to this market
  - Data diodes that allow only User Datagram Protocol (UDP) data to be sent from the lower classified side (called “low side”) to the higher classified side (called “high side”)

  Other communications are made physically impossible (see Figure 11).

## Figure 11. Government Network Architecture




- Business requirements for additional security zoning (flowing from the so-called “need-to-know principle”) are implemented using subzones. A subzone is a subset of a network security zone demarcated by a network perimeter or a virtualized perimeter solution. Systems in one subzone may not communicate with systems in another subzone without crossing some type of security

mechanism. Figure 11 illustrates a number of subzones. For example, data up to a defined classification (for example, “Restricted”) can be processed in the darker blue zone (Classified Zone, Prod.). Enterprise applications, such as ERP, are ring-fenced in a subzone of the classified zone.

### Subzones

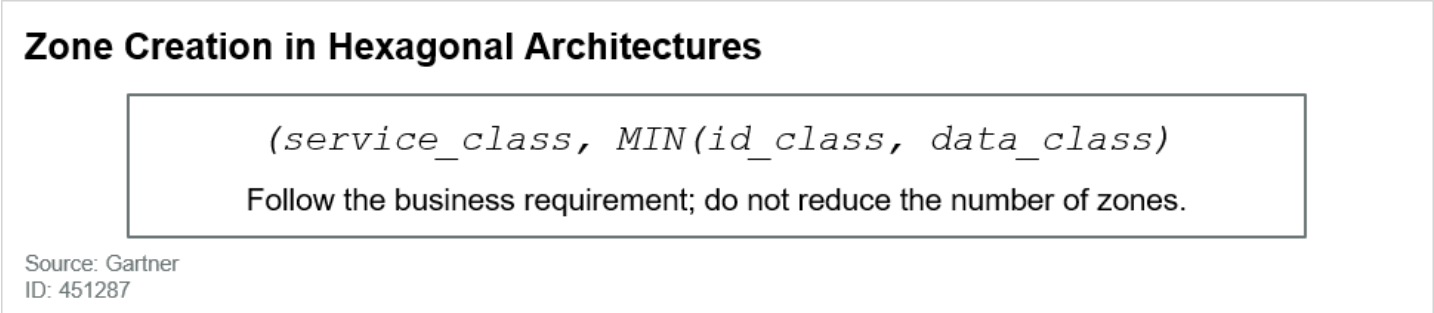
A “subzone” is defined as “a subset of a network security zone demarcated by a network perimeter or a virtualized perimeter solution.” Subzones need to be separate from other parts of their parent network security zone but should still maintain the same basic connectivity characteristics as the rest of the zone. For example, the DMZ subzone in Figure 11 will allow some connections from the untrusted zone, but the systems in the subzone are to be limited in terms of the connections they can initiate into the “Unclassified Zone.”

There may be any number of subzones within each network security zone, depending on how the organization sees fit to separate systems. However, the number of subzones is tied to the costs associated with perimeter protection mechanisms. The larger the number of required perimeters, the higher the cost will be in terms of mechanisms and management. Organizations should examine their systems to identify the most appropriate way to divide them into subzones in order to balance risks and costs.

### Hexagonal Architectures

Clients who have this type of architecture need to follow the business requirements. Gartner recommends that clients initially split the triplets (that is, the enumerated zones) into services and nonservices (such as corporate user workstations), identifying all systems or (micro)services [6] that logically belong to the central shared-service pool. To identify services and enclaves — such as systems in scope of PCI DSS — that need to be segregated with further ZETs, clients do not limit themselves to a maximum number of security zones. Nor do they limit the number of attributes used when enumerating the security zones (illustrated in Figure 12).

Figure 12. Zone Creation in Hexagonal Architectures

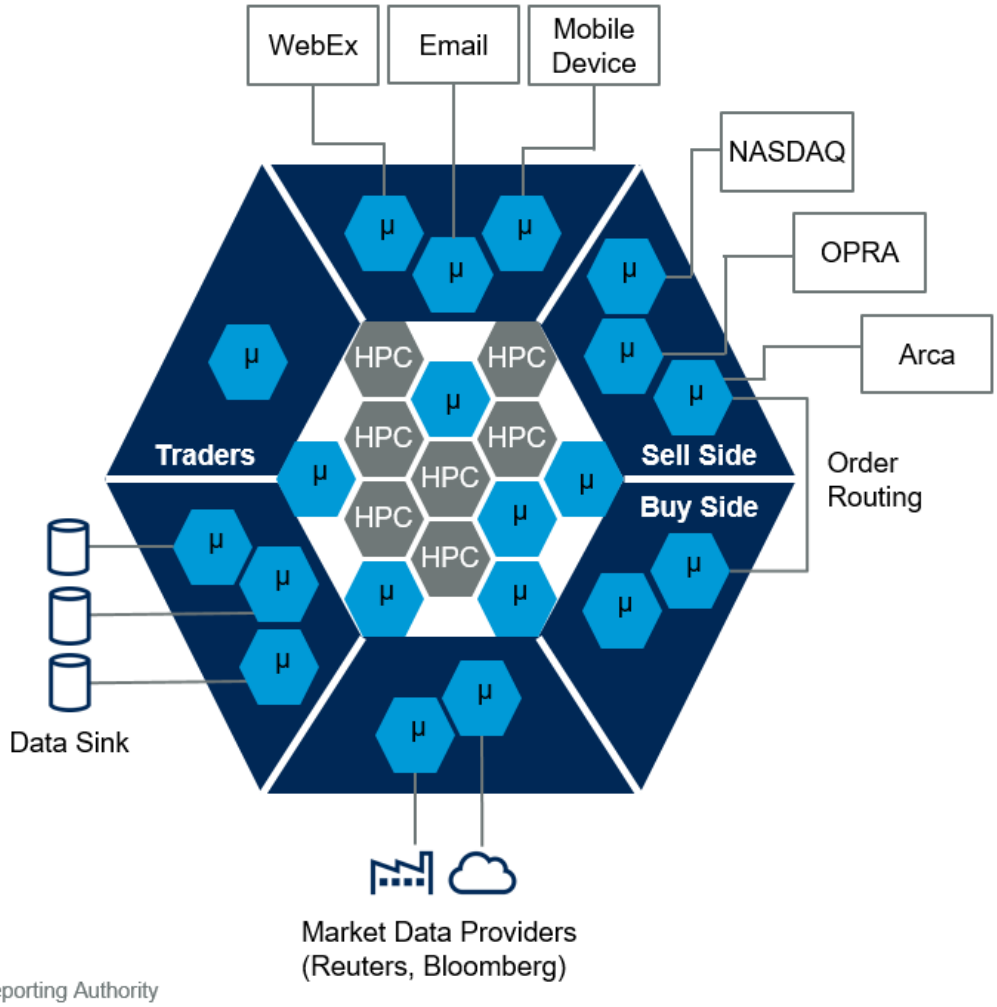



### Hexagonal Network Properties

- There is a boundary (“perimeter”) between the application/compute domain and the rest of the world.
- It builds around business function.
- There are various entry points of (streaming) data.
- The size of central services, such as high-performance computing (HPC) or microservices, exceeds the size of the noncentral services.
- Microservices are hugely interconnected without their own presentation layer.
- HPC nodes or Model-View-Controllers (MVCs) are members of multiple security zones.
- Hexagonal network architectures draw a boundary between the application domain (the main business logic) and the rest of the world (see Figure 13). [7] There are no separate presentation, application and data tiers. Instead, there is a perimeter around the central shared services that can belong to multiple security zones. Frequently, a certain amount of bleeding of microservices out of the central service pool cannot be avoided. For example, an HFT platform may consist of a central, well-differentiated compute platform with many ways in and out and auxiliary business (communication) services — such as email, portals and IP telephony — in the outer layer. This emphasizes the need for ZETs that support logical security zones independent of system or service placement (for instance, microsegmentation approaches that are less resource-intensive in highly complex environments). Microsegments also make sense from a compliance, business and risk perspective because, in hexagonal architectures, we find small enclaves that are covered by strict requirements.

Figure 13. Hexagonal Network Architecture of an HFT Platform

**Hexagonal Network Architecture of an HFT Platform**




OPRA = Options Price Reporting Authority
Source: Gartner
ID: 451287

- The use and access of data need special considerations. In contrast to microservices, data is subject to a "gravity pull" and is not portable. Thus, data stores frequently need to be shared across security zones while preserving the quality of the zones, which makes data-based ZETs invaluable. (For an overview, refer back to Table 1).

For more information about microsegmentation, see "Architectures and Paradigms of Microsegmentation Products" and "Solution Comparison for Microsegmentation Products."

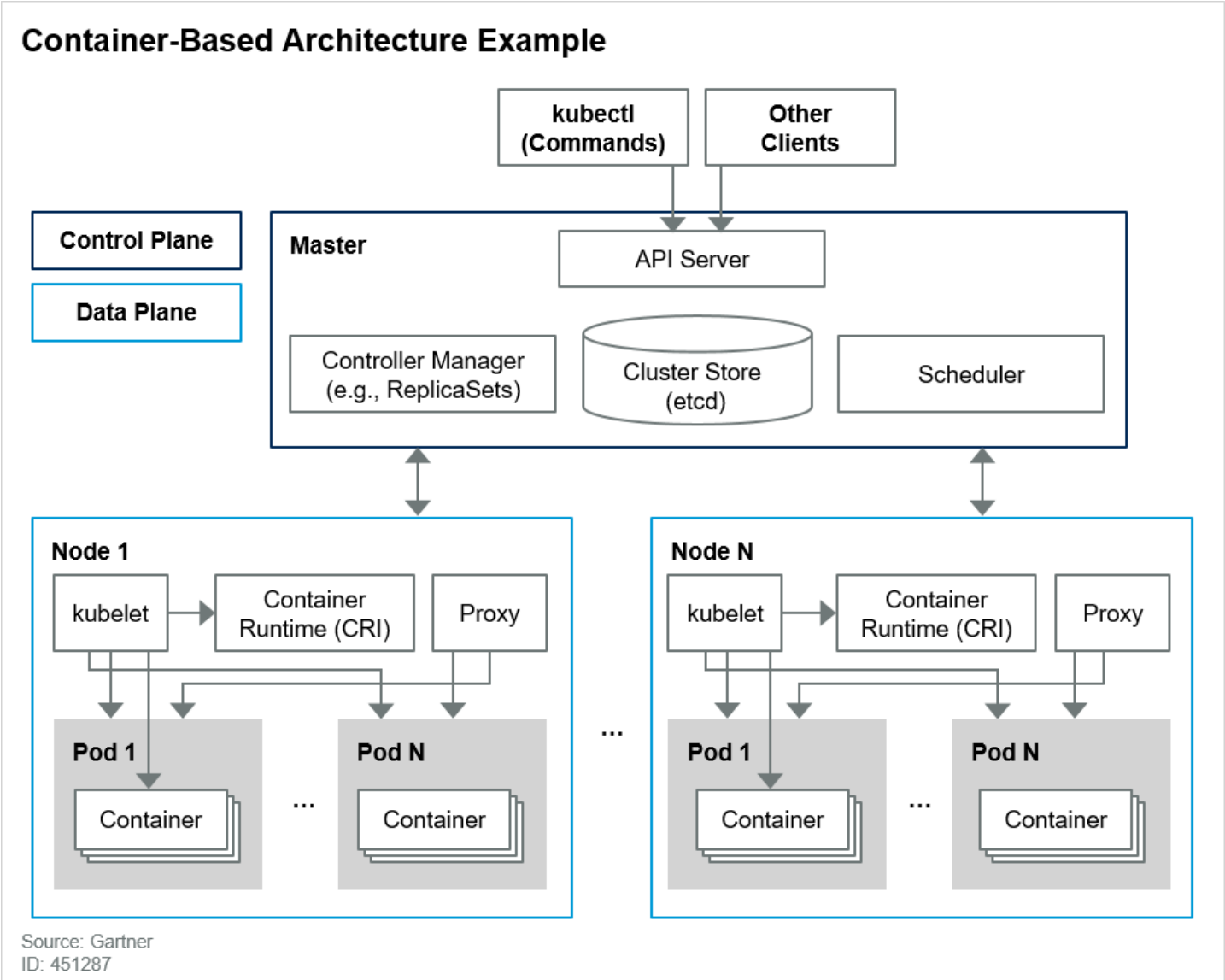
### Container-Based Architecture

Container-based architectures (see Figure 14) are an emerging technology area for which different segmentation approaches may be used (see Table 3). Traditional network and security architects often feel the need to "force fit" a traditional zoning model onto the new technology. On the other hand, proponents of DevOps, DevSecOps and modern application architectures often believe that traditional security zones are counterproductive for containers, and that more modern, microservices architecture (MSA) approaches should be used.

**Regardless of the approach used, it is important for organizations to have as few security zones as possible in a container-based architecture.**

To compensate for this, organizations may implement ZETs that are on higher OSI layers, such as web application firewalls or the Istio service mesh (see Using Higher-OSI-Layer Controls for Container Segmentation section).

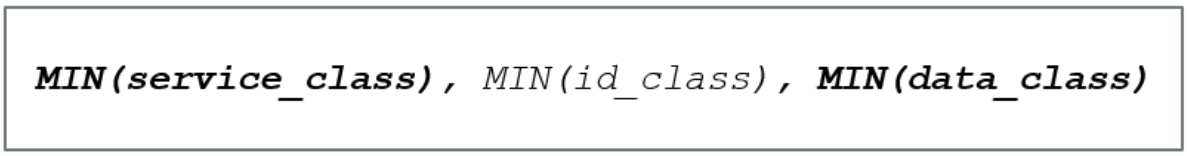
Figure 14. Container-Based Architecture Example




In container-based architectures, a zone may be formally defined by the triplet shown in Figure 15.

Figure 15. Zone Creation in Container-Based Architectures

**Zone Creation in Container-Based Architectures**




Source: Gartner
ID: 451287

In this case:

- “service_class“ may be seen as services in the DMZ or for testing (with the term “service” describing the application in a wider sense).
- “id_class” could indicate the physical location, such as connected to an external system, or part of development/testing or the application.
- “data_class” here could refer to the regulatory framework.

### Using Network Segmentation for Containers

Organizations often use network segmentation for containers to achieve the following goals:

- To separate development/testing from production
- To separate application services exposed to untrusted environments from applications and services that are consumed from the trusted environment only
- To limit the scope of compliance frameworks, such as PCI DSS or ISO 27000

The last point will become increasingly critical as more container-based environments are deployed that contain data subject to audits related to compliance frameworks. For example, PCI DSS security requirements apply to all systems within or connected to a defined customer data environment (CDE), which means organizations must accurately determine the scope of the CDE under review. In a container environment, the simple solution would be to define an entire Kubernetes cluster as “in scope.”

However:

- The broader the defined scope is, the higher the expense will be (e.g., due to auditing or management costs). Also, a broader scope may increase the risk that the CDE will be exposed to vulnerabilities and other unwanted issues.
- On the other hand, if scope is defined narrowly, costs will likely decrease — but the risk rises that some cardholder data may be accidentally excluded from your CDE.

Clients who deploy network-based zoning for container-based architectures will face questions similar to those that arose in the past with the advent of virtualized infrastructures. For example, is an environment provisioned with containers good enough to segregate commingled containers with different security levels? The short answer is technically yes, but several factors related to cost and practicality will likely make it inadvisable (see the Gartner blog "About Commingling Virtual Machines"). Other issues involve whether container-based data with differing security levels must be separated through network zoning, or whether this can be achieved mapping Kubernetes pods to different security levels and sorting the data into different pods.

These and other issues are addressed in Table 3, which describes several of the integration points where segmentation can be enforced in a container-based architecture.

**Table 3: Potential Integration Points for Network Segmentation in Container-Based Architectures**

| Level of Segmentation | Explanation |
|---|---|
| **Container-Level Network Segmentation** | This can be achieved using offerings such as:<br>■ General microsegmentation products, such as Illumio, VMware NSX-T Data Center and ShieldX<br>■ Container security products that extend into the DevOps pipeline beyond the network, such as Aqua Security, NeuVector or Palo Alto Networks-Twistlock<br>■ SDN plug-ins for Kubernetes, such as Tigera or Weaveworks |
| **Logical Segmentation at the Pod Level** | In this approach, logical segmentation is achieved by aggregating containers with similar security requirements into pods. By default, Kubernetes gives every pod an ingress network security policy filter. Most clients deem this approach to be sufficient for their needs. (See "Container Security — From Image Analysis to Network Segmentation, Options Are Maturing.") |
| **Physical-Node-Level Separation** | Nodes maybe provisioned on VMs or bare-metal hardware. The advantage here is that you can specify what pod runs on what node and achieve quasi-physical separation between pods. |

| Level of Segmentation | Explanation |
| --- | --- |
| **Segmentation at the Kubernetes Level** | Segmentation can also be achieved at the Kubernetes level — for example, by using different Red Hat OpenShift Container Platform (RHOCP) clusters. Clients frequently used dedicated RHOCPs to separate various types of environments (e.g., development/testing vs. production, on-premises vs. cloud, or PCI vs everything else). However, this dedicated approach may become less popular in favor of a new option that RHOCP recently introduced — the possibility to create stretched clusters, where only certain components are duplicated into the other security zones or the physical zone. |

Source: Gartner (November 2019)

### Using Higher-OSI-Layer Controls for Container Segmentation

The approaches discussed above involve applying network segmentation to containers. This is appropriate for “lift and shift”-type or monolithic applications (and virtually all applications on Docker Hub are such lift-and-shift types). However, organizations using modern approaches, such as MSA, typically try to do less with network-based security and compensate with higher-layer controls. These include:

- Web application firewalls
- API gateways (sometimes deployed in every Kubernetes node) to mediate traffic to APIs (Together with web application firewalls — or perimeter firewalls in front of the Kubernetes cluster nodes — this approach is often considered to be sufficient for network security.)
- The Istio service mesh, a secure proxy that mediates network and service connections between microservices

### Recommendations for Using Segmentation in Container-Based Architectures

Network security for containers is a nascent field where no clear best practices have developed (with the exception of Kubernetes ingress pod policies). Given this, Gartner offers the following recommendations:

- Address compliance requirements first, and aim to limit the scope of any compliance framework. Contact your internal auditor or a qualified security assessor (QSA) to discuss your architecture implementation plan before proceeding.
- Start small and expand gradually into larger environments. Wherever possible, perform a comparison between at least two products.

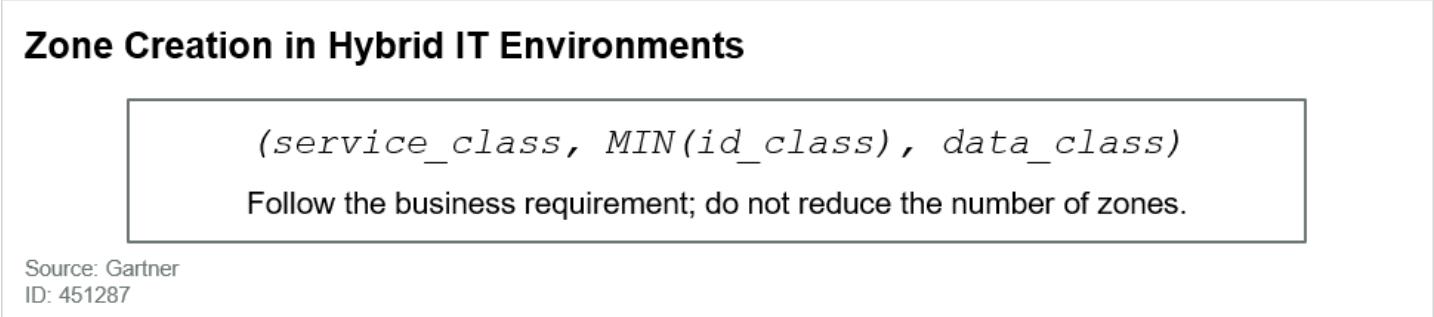
- For everything else, take a risk-based approach. For example, do not implement tier-based zoning simply because this is what you've done in the past. Go back and question what risks you may need to mitigate.

### Hybrid IT and Cloud-Based IaaS Architectures

Clients who have hybrid IT architectures need to decouple security zoning and workload placement. Gartner recommends that clients do not limit themselves to a maximum number of security zones and do not reduce the number of zones. Instead, they should focus on the efficacy and manageability of the deployed ZET.

A triplet for zone creation in hybrid IT environments is shown in Figure 16.

Figure 16. Zone Creation in Hybrid IT Environments




When using cloud-based IaaS architectures, clients have the choice between several ZETs. Although this choice poses advantages to organizations by allowing them to tailor the segmentation architecture to their use case, it is also a source of uncertainty.

ZETs that are available in IaaS include:

- **Cloud security groups,** such as AWS security groups or Microsoft Azure network security groups (NSGs)
- **Third-party microsegmentation products,** including microsegmentation products that are agent-based (from vendors such as ColorTokens, Guardicore or Illumio) or VXLAN-based (from vendors such as ShieldX)
- **Virtual versions of traditional network enterprise firewalls** (such as Check Point Software Technologies or Palo Alto Networks), which can also be used in many public clouds
- **ZETs that are not IP-based** — for example, segmenting functional areas through accounts (in the case of AWS) or subscriptions (in the case of Azure)

**If your CSP offers strong identity-based segmentation, such as AWS Landing Zone or AWS Control Tower (see Figure 2), the non-IP-based segmentation has priority over the traditional IP-based segmentation. In this case, first deploy the Landing Zone and look into IP-based segmentation to control east-west traffic or to create subzones in large accounts.**

### Common Questions About Security Zoning in IaaS Architectures

In inquiries with Gartner clients, several questions about IaaS security zoning arise. The two most common of these questions are answered below.

#### Do I need an IP-level firewall between my cloud deployment and my on-premises data center?

Gartner finds that the answer to this question often depends on the interests and perspectives of various groups within and outside the organization. For example:

- Developers and DevOps-minded groups often perceive each IP-level control to be an inconvenient stumbling block that slows them down, and therefore argue that no firewall is needed.
- Security operation center (SOC) and infrastructure operators often argue in favor of IP-level firewalls to limit the "blast radius" of any given deployment, or to create chokepoints for the creation of audit trails.
- Cloud service providers (CSPs) or consultants will often point out that you have a contract with the CSP that already guarantees you will benefit from world-class security. It is important to bear in mind, however, that this does not account for any configuration errors or flaws that may exist within your own custom applications.

#### Where is the right place to put a firewall between the public cloud and my on-premises data center?

To segment off their on-cloud deployment from their on-premises data center, organizations will often insert a traditional network enterprise firewall in either of the following two locations:

- **Into a colocation hub:** Colocation network hubs and exchanges — such as CyrusOne, Equinix, Digital Realty and QTS Data Centers — offer facilities where multiple carriers meet and exchange internet traffic. Not every enterprise uses interconnect services that have colocation hubs. For those that do, however, it can be beneficial to have a presence in a colocation hub and to use its high-bandwidth connectivity with carriers and internet providers. Cloud providers often have private, high-bandwidth connectivity with colocation hubs, and sometimes host some of their own services in such facilities. In addition, getting high-bandwidth, low-latency connectivity from

multiple carriers and cloud providers through a colocation hub is inexpensive because crossover connectivity is usually available, bypassing the physical network access cost.

- **Into the WAN access edge:** The WAN access edge is the attachment point between the organization and its external networks, which include both the private WAN and the internet. It provides physical connectivity to a communications service provider's MPLS services, metro Ethernet services and internet service providers, as well as logical connectivity via mechanisms such as IPsec tunnels.

> **It is hardly possible to talk about defense in depth and microsegmentation while not using the obvious boundaries for segmentation.**

In addition to the two examples discussed above, another common situation for the placement of a traditional network enterprise firewall involves PCI compliance. In this case, the solution typically involves placement of a virtualized traditional firewall in a position that controls where the network traffic of systems within the scope of PCI DSS may pass. This is often the only way to implement the intrusion detection system/intrusion prevention system (IDS/IPS) requirement of PCI DSS.

#### Use Cases and Options for Using Cloud-Native ZETs Instead of Firewalls

Cloud-native ZETs should be used in many cases. In fact:

> **Practically the only use cases that are appropriate for the placement of a traditional enterprise firewall are the colocation hub, WAN edge firewall and PCI DSS compliance cases discussed above. In other cloud architectures, you should choose cloud-native ZETs.**

For these other use cases, Gartner offers these recommendations for evaluating cloud-native ZETs:

- Use accounts (AWS) or subscriptions (Azure) for coarse-grained segmentation of large functional areas, such as development/testing or geographic regions. This approach is also known as using "landing zones."
- Use CSP-provided microsegmentation, such as cloud security groups.

- If CSP-provided microsegmentation does not scale for your teams (for example, you need multicloud-ready or hybrid-IT-ready segmentation), look into agent-based third-party MSEG products. For more information about microsegmentation, see "Architectures and Paradigms of Microsegmentation Products" and "Solution Comparison for Microsegmentation Products."

- If you need additional perimeter security (e.g., for global web applications), look into perimeter security provided by content delivery networks (CDNs), such as Microsoft Azure Front Door or third-party CDNs.

#### Leveraging Microsegmentation in "Brownfield" Approaches

Clients that have implemented traditional architectures frequently ask Gartner how they could acquire a modern architecture — such as a hexagonal microservices architecture, service-oriented architecture (SOA) or hybrid IT architecture — but in a "brownfield" approach. Although modern network zoning paradigms can completely replace traditional architectures, they do not need to. They can also be used as add-ons to existing architectures where they greatly increase the security posture.

Gartner recommends creating five to 20 manually defined major zones that are, for example, based on the decision model for traditional networks (see Figure 9) or your existing zones.

Inside the major zones, clients can then identify areas that will benefit from microsegmentation and have a tool, such as Cisco Tetration Analytics or Illumio Policy Compute Engine, to identify applications and protect them with microboundaries. It is important to note that, although the term microsegmentation seems to imply that every server is protected with a laser-cut policy, the products generally support application-by-application segregation of systems or containers. They usually do so by discovering and grouping communication relations that most likely belong to the same application.

## Alternatives

The logical zoning architecture presented in this Decision Point is the de facto standard. In all digital environments, zoning will be present in one way or another, and there are no alternatives — especially when we do away with the thought that traditional firewalls are the only and best ZET.

In some niche use cases, organizations do not want or cannot implement logical security zones and must consider other architectures, such as:

- Open architecture

- OT networks

### Open Architecture

The early internet is a good example of an open architecture — systems connected directly to the internet with little or no protection against malicious activity. Each host or network resource was open to any type of communication (assuming that the service existed on the system), and the security of the system was reliant on the security mechanisms inherent in the operating system itself.

Some organizations have substantial components that require the open-architecture approach. For example, institutions that research new protocols may be adversely affected by perimeter protection systems whose security mechanisms might prevent the use of experimental protocols. In other cases, an organization may require that a large number of previously unknown individuals be granted access to a majority of systems within the organization's network. Internet groups and other COIs are good examples of this. In this instance, managing perimeter configurations (other than those provided by a service provider) may be too costly or inflexible for an organization. This type of environment may also be described as a "collaboration stew," where anything that is not specifically prohibited is allowed.

In an open-architecture environment, each system must be able to protect itself from attack, or the organization must be able to survive the failure of such systems with no significant consequence. Similar to microsegmentation approaches, this architecture can be described as each system being in its own zone, protecting itself from malicious attacks and from accidental violations of security. Whether to allow the transmission of sensitive information must be carefully considered. It is likely that any transmission of unprotected sensitive material will be intercepted by unauthorized individuals.

## OT Networks

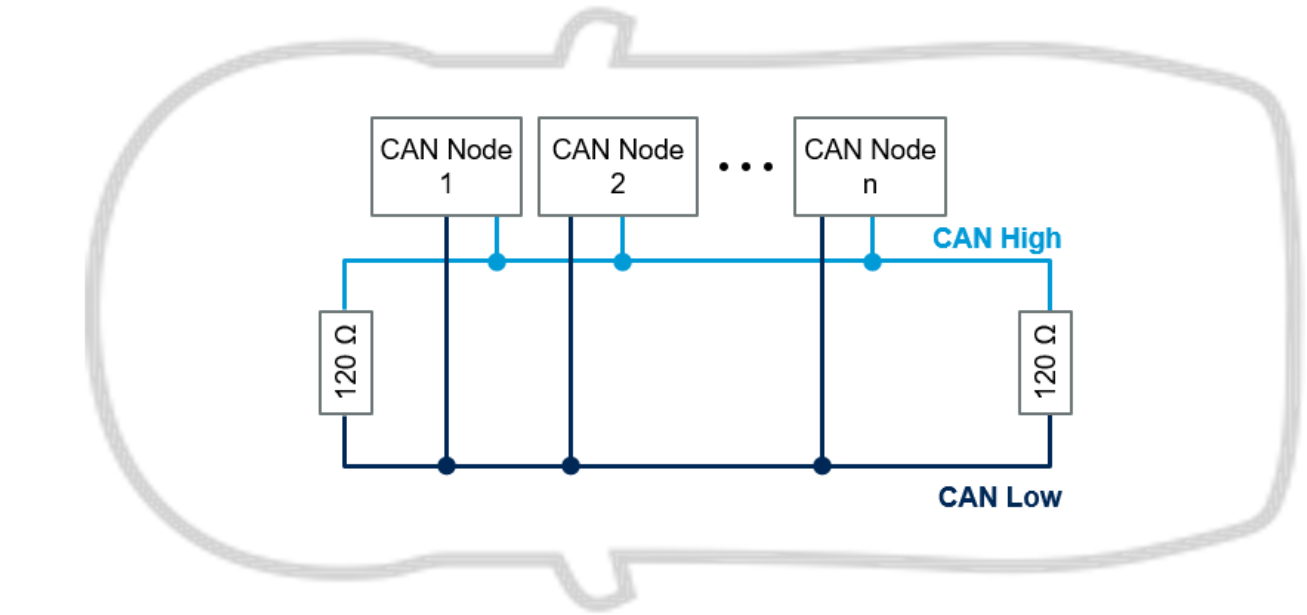
OT networks are specialized networks, including controller area network (CAN) buses and industrial Ethernet (IE). Security (zoning) standards for OT, such as ANSI/International Society of Automation (ISA)-99 or International Electrotechnical Commission (IEC) 62443, are traditional architectures or not present at all. OT environments are separated from IT environments because of integrity and availability concerns. Traditionally, this has caused little consternation because there were no permanent or high-bandwidth communication requirements for the rest of the IT infrastructure. However, with the push for the Internet of Things (IoT) and Industry 4.0, and the increased desire to collect and utilize data, this is currently changing.

### Controller Area Network Buses

CAN buses are used in vehicles. A CAN bus is a closed serial network that has two main zones — a high-speed bus and a low-speed bus (see Figure 17). In this environment, every connected system is considered equally trustworthy, and every component has implicit access to every other component. Generally, systems on the bus cannot be accessed from outside the closed community — such as a vehicle. Perimeter security is handled primarily through physical controls, and the environment is not checked for unauthorized or compromised systems.

A single compromised device has sufficient access to compromise all systems. For example, a compromised MP3 player that is connected to the low-speed CAN bus of a car can also compromise the brakes and the engine. That MP3 player may have been compromised via a short-range communication channel, such as Bluetooth. On the other hand, the multimaster, controllerless CAN bus is a very good starting point for further exploring microzoning strategies.

Figure 17. Vehicle CAN Bus




## Future Developments

Because security zoning is so fundamental to IT security (and functional) architecture, it takes years to change. Thus, the future holds increased implementation of already-mature technologies and ZETs, such as software-defined networks, virtualized data centers and microsegmentation.

### Automated Provisioning and Microsegmentation

Software-defined networking and microsegmentation are rapidly emerging technologies. Vendors are increasingly offering the ability to integrate with identity management systems and virtualization technology to deploy customized security controls that are specific to the user or system instance.

### IoT and Industry 4.0

IoT and Industry 4.0 platforms have a dynamic perimeter that frequently extends to CSPs (for example, to use a cloud-based IoT platform or services such as AWS Lambda) and end customers. Securing and zoning the new perimeter require the successful combination of new security paradigms; appropriate, adaptive security architectures; and human actors.

IoT and Industry 4.0 are not monolithic blocks that clients can fence off with a static perimeter and later plug the gaps with a third-party security product. IoT and Industry 4.0 security means addressing appropriate levels of security in constantly evolving, distributed multicomponent systems that expose a very large attack surface.

All components must be resistant to attack, tolerant of failure, flexible, scalable, adaptive and adaptable. To get a firm grip on IoT and Industry 4.0 security, that security must evolve into a first-class citizen in all components. To achieve this, vendors and end customers need to reapply what they have learned and already do well in traditional enterprise security applications and data, but plan for much greater scale and agility without a significant "security penalty."

## Evidence

[1] Gartner defines microsegmentation as architectures where the security policy can be applied at the workload (virtual machine, bare metal or container), as opposed to the network or layer and regardless of the topology and location of the workloads.

[2] "Survey Results: Data Classification Success Factors"

[3] "JS.do Online JavaScript Editor," JS.do

[4] Hexagonal architectures were first mentioned by Alistair Cockburn (co-author of the Agile Manifesto) in the year 2005. Various versions of this design pattern can be found in practice.

[5] "Information Assurance (IA) Implementation," Department of Defense Instruction.

[6] Independent, deployable services that are part of an application.

[7] Hexagonal architectures were first proposed in 2005 by Alistair Cockburn. Although lacking a formal name, HFT platforms have been using similar architectures since the inception of SOA.

## Note 1
## Zone Enumeration and Reduction

**Zone enumeration** is most easily understood as the selection of every possible group triplet. In pseudo code, this could be implemented as shown in Figure 18.

Figure 18. Zone Enumeration

**Zone Enumeration**

```
security_zones = []
(system_classes, user_classes, data_classes) =
                                        getClassifications()
for sc in system_classes:
       for uc in user_classes:
              for dc in data_classes:
                     security_zones.append((sc, MIN(uc), dc))

#remove overly small (too expensive) security zones
reduce(security_zones)
print security_zones
```

Source: Gartner
ID: 451287

**Zone reduction** is often required for certain ZETs to match the architectural requirements and constraints. In pseudo code, this could be implemented as shown in Figure 19.

Figure 19. Zone Reduction

**Zone Reduction**

```
#reduce(security_zones)

#general practicality factors

If size(zone) < MIN_ZONE_SIZE #if it's too small to be practical

If size(ZONE) > MAX_ZONE_SIZE #is the zone too big? The more
systems in a zone, the greater the risk to lateral systems


#if the policies between zoneX and all other zones is the same as
zoneY and all other zones, then merge them unless the result is
too big


If policy(zoneX, :) = policy(zoneY, :) and
      policy(zoneY, :) = policy(zoneX, :) and
      size(zoneX + zoneY) < MAX_ZONE_SIZE then:
            merge(zoneX, zoneY)
```

Source: Gartner
ID: 451287

A JavaScript tool can be used to enumerate possible zones, as shown in Figure 20.

Figure 20. Enumerating Possible Zones in JavaScript

## Enumerating Possible Zones in JavaScript

```javascript
<script>

var input = [
  ["web_srv", "anonymous", "Public"],
  ["lan_srv", "everyone", "Restricted"],
  ["", "employee", "Protected"],
  ["", "priv_user", ""]
];

for (i = 0; i < input.length; ++i) {
  for (j = 0; j < input.length; ++j) {
    for (k = 0; k < input.length; ++k) {
        document.write("[ '" + input[i][0] + "' , '" + input[j][1] + "' , '" + input[k][2] + "']");
        document.write("<br>");
    }
  }
}

</script>

```

Source: Gartner
ID: 451287

## Document Revision History

Decision Point for Postmodern Security Zones - 11 September 2017

Decision Point for Postmodern Security Zones - 16 March 2016

Decision Point for Network Security Zones - 10 July 2012

Zones - 22 October 2010

Zones - 5 December 2007

## Recommended by the Author

Architectures and Paradigms of Microsegmentation Products

Solution Comparison for Microsegmentation Products

Container Security — From Image Analysis to Network Segmentation, Options Are Maturing

Improving Data Security Governance Using Classification Tools

## Recommended For You

Mitigating the Risk of Phishing When Technical Security Controls Fail

Best Practices for Choosing Network Security Controls Between EFW, SWG and CASB

Security Operations for Technical Professionals Primer for 2019

2020 Planning Guide for Security and Risk Management

Building the Foundations for Effective Security Hygiene