

ASSET NAME	
ASSET VERSION	v0.4 12.07.2023
ASSET OBJECTIVE	<ul style="list-style-type: none"> Help you understand the OCI Landing Zones landscape, with the solution characteristics and maturity. Provide you with technical guidelines on why you can consider using a solution.
ASSET CONTENTS	[1. WELCOME NOTE] [2. UNDERSTAND THE LANDSCAPE] [3. WHY USE AND CONSIDERATIONS BEFORE USING]

1. WELCOME NOTE

Welcome to the Select Your Landing Zones Solution asset. This asset presents the OCI Landing Zones solutions available with IaC, and IaC solutions to create OCI landing zones. Note the previous versions of the OELZ are not included.

2. UNDERSTAND THE LANDSCAPE

APPROACH		STANDARD / PRESCRIBED		TAILORED
AREA	TOPIC	OELZ v2	CIS LZ v2	CIS LZ v3 MODULES
SOLUTION STORY	DESCRIPTION	An OCI Product Management Landing Zone ready to start the OCI Journey, with hub & spokes, application-driven design with multiple workloads, using ORM and IaC.	A Secure Landing Zone to start the OCI journey with a layered approach, several network topologies (shared to hub & spoke), with top security CIS 1.2 compliance and validation, using ORM and IaC.	A framework to create new and all types of Landing Zones (from simple to complex ones) without requiring Terraform code customization. Includes CIS LZ v2 and its capabilities, top IaC modules, and several LZ configuration blueprints.
	DOCUMENTATION	Here and here	Here and here	Modules: [1] [2] [3] [4] [5]
	LICENSING	UPL v1.0	UPL v1.0	UPL v1.0
SOLUTION SUMMARY	CUSTOMIZATION	Code-Driven	Code-Driven	Configuration-Driven
	ENVIRONMENTS	Prod and Non-Prod	Multiple	Multiple, Configurable
	CIS SECURITY	CIS 1.2	CIS 1.2 (Embedded)	CIS 1.2 (Embedded)
	NETWORK	Hub & Spoke, no NSGs	Shared VCN, Hub & Spoke, with NSGs.	All, Configurable
	TARGET WORKLOADS	Three-Tier Applications and ExaCS	Three-Tier Apps and ExaCS	All, Configurable
	MULTIPLE WL	Yes ELZ Workload	Yes By Spokes	Yes Configurable
	IAC REPOSITORY	OCI OELZ	OCI CIS LZ	IAM , Network , Governance , Observability , and more are coming...
SOLUTION MATURITY	OPERATIONS INTERFACE	ORM, TF CLI	ORM, TF CLI	TF CLI, CI/CD, Config-as-Code (GitOps)
	BLUEPRINTS / PATTERNS	ExaCS	ExaCS	OCI CIS v2, OCI Open LZ Blueprint , and others.
	SOLUTION DESIGN RECOMMENDED? (approach type)	<input checked="" type="checkbox"/> YES (prescribed)	<input checked="" type="checkbox"/> YES (prescribed)	<input checked="" type="checkbox"/> YES (blueprints/tailored)
	SOLUTION IS CUSTOMIZABLE?	<input checked="" type="checkbox"/> YES (by code)	<input checked="" type="checkbox"/> YES (by code)	<input checked="" type="checkbox"/> YES (configurable)
	CUSTOMIZATION EFFORTS?	<input checked="" type="checkbox"/> MEDIUM-HIGH By Code / No TF Best Practices Not following Terraform's best practices pays a toll. The consequence of this it's less reuse and more effort into customizing a solution, with documentation gaps, duplication of resources, incomplete naming conventions, and the overall risk of adopting poor coding practices.	<input checked="" type="checkbox"/> MEDIUM By Code / TF Best Practices To customize you need to change the code, and this means developing efforts. This can be simpler, or more complex depending on the target solution. Regarding the code, it's well organized with a modular approach, coded with TF best practices , and can be extended with reuse. We have s using this solution as a starting point to create their OCI IaC assets. It's also a great entry point for the IaC journey.	<input checked="" type="checkbox"/> LOW By Configuration / No Coding This solution is a game change to customizations. By using a set of Terraform modules that receive HCL JSON values it can separate configurations from code development or customizations, meaning your solution is just a set of configurations as input to the generic code. With this approach, efforts are focused on configurations and not on coding . This means less time to value and few implementation efforts. You can configure your own solution in no time.

3. WHY USE AND CONSIDERATIONS BEFORE USING

APPROACH		STANDARD / PRESCRIBED		TAILORED
SOLUTIONS		OELZ v2	CIS LZ v2	CIS LZ v3 (MODULES)
WHY USE		<p>PM Ownership</p> <p>This is the solution proposed by OCI Product Management.</p> <p>Note that the licensing is UPL 1.0 and the support channel is GitHub Issues.</p> <p>Simple Applications Onboard</p> <p>The solution has an application-driven design, ready to onboard multiple workloads into OCI with a simple/flat landing zone structure.</p> <p>Each workload can have its own responsible team, with a dedicated compartment and a spoke VCN. The workload layers are isolated with Subnets.</p> <p>Prod/Non-Prod Environments</p> <p>The solution is ready for landing zone Non-production and Production environments.</p> <p>To confirm your understanding of this topic, please read "Approach to Environments" for OELZ v2 and</p>	<p>Worldwide Adoption</p> <p>It is a solution adopted worldwide in NA, EMEA, and JAPAC. The customer base is high and growing. The partner community is using this solution.</p> <p>Strong Security (CIS Embedded)</p> <p>Security-centric solution implementing CIS OCI Foundations Benchmark version 1.2. The solution implements the recommendations and contains a script to validate any tenancy against the recommendations on the benchmark.</p> <p>In terms of design, the solution ensures the segregation of duties for a set of pre-defined personas, delivering operationally ready-to-use environments. At the network level, there is strong network isolation with VCNs, Subnets, and NSGs. It also lets users choose OCI security services to enable.</p> <p>Three-tier & Segregation of Duties</p> <p>The basic design is suitable for three-tier apps, deploying one VCN with one public subnet and two</p>	<p>Focus on Configs, Not Code</p> <p>With this approach, efforts are focused on configurations and not on coding. This means less time to value and few implementation efforts. You can configure your own solution quickly.</p> <p>The solution uses a set of Terraform modules that receive HCL JSON values, separating configurations from code development/customizations, meaning your solution is just a set of configurations as input to the generic Terraform modules.</p> <p>Suitable for DC Exits</p> <p>DC Exits normally involve a high number of workloads landing on OCI, with different characteristics, with specific separations of duties across the organization, and very fine-tuned isolation of resources - possibly different by workload type. As this solution is configuration-driven, it makes it easier to set up the target solution with less effort focusing on configurations and little coding.</p>

APPROACH	STANDARD / PRESCRIBED		TAILORED
	<p>CIS LZ v2, as there are several different approaches to this feature.</p> <p>laC Modular Approach</p> <p>The modular design based on Terraform modules provides flexibility to optimize a Landing Zone design.</p> <p>With this model, this solution is a simple entry point for the laC journey, simple to learn and start coding.</p>	<p>private subnets. A public subnet is used for the load balancers and bastion servers. The application and database tiers are attached to separate private subnets. The solution can also create VCNs to support the deployment of specific workloads, like Oracle Exadata Database Service.</p> <p>The segregation of duties is implemented with several management groups, including groups for security, network, database, and applications, among others.</p> <p>Strong Network Isolation</p> <p>Several network topologies are supported, from shared to a hub and spoke, including using NSG.</p> <p>No Initial Consumption Cost</p> <p>The deployment of this solution creates resources that initially don't trigger consumption costs.</p> <p>The following optional and paid components can incur costs:</p> <ol style="list-style-type: none"> Object Store, it's not configured by default. OCI Logging, it's configured by default and will be paid after 10 gigabytes of Log Storage per month. OCI Notifications, the Service Connector is not configured by default. This service will be paid over 1 million Delivery Operations Per Month <p>Regarding 2 and 3, the deployment of these services configured doesn't incur an immediate cost as you will be far billable thresholds.</p> <p>Be aware that to use this solution you need a paid tenancy due to the limitations of the Always Free tenancies.</p> <p>Modular laC with Best Practices</p> <p>The code is well organized with a modular approach and coded with TF best practices and can be extended with reuse. There are several customers using this solution as a starting point to create their own OCI laC assets.</p> <p>It's also a great entry point to start or continue an laC journey.</p> <p>Multiple Environments</p> <p>This solution provides a simple approach to multiple environments: "1 environment - 1 deployment". To be able to do this, you will deploy a non-production environment in an enclosed compartment. Note that we recommend the production environment should be at the root level and deployed in a second iteration.</p> <p>A lot of our customers prefer this separation of environments by deployment, but there are several alternatives for this feature. For example, the OELZ v2 solution provides you with the landing zone environment concept.</p> <p>To confirm your understanding of this topic, please read "Caution with Environments" for OELZ v2 and CIS LZ v2, as there are several different approaches to this feature.</p> <p>Incremental Versioning</p> <p>The CIS LZ v2 is an incremental update over the v1, and it shares the same repository.</p> <p>This same solution is implemented in the same repository with CIS LZ v3 configurable approach.</p> <p>Multi-Region Deployment</p> <p>This solution provides you the ability to extend an existing landing zone into several more regions.</p> <p>IAM, IDCS & Identity Domains</p> <p>This solution works with OCI Local IAM (in combination with IDCS) or the OCI Default Identity Domain.</p>	<p>See the bullets below on organization and cloud operating models, as they are normally associated with large movements of workloads into OCI.</p> <p>Several Workload Types</p> <p>If you need different workloads types to land on OCI, meaning several resource structures, and potentially different security or network topologies, this solution is a great fit because it allows you to configure a landing zone with these workload patterns at a fast pace and reduce efforts - as you will not be required to code it.</p> <p>Onboarding Enterprise Organizations</p> <p>You need to onboard your organization into OCI (Operating Entities, OpCos, LoB, Departments, etc.) and you need to reflect that into the Landing Zone.</p> <p>Use the OCI Open LZ Blueprint for these cases.</p> <p>Customized Operating Model</p> <p>You need a highly tuned cloud operating model, having several teams operating different types of resources or areas of the landing zone, or even having shared areas where some resources are operated by one team and other elements are operated by another team.</p> <p>Operations at Scale - GitOps</p> <p>This solution is a perfect fit if you're operating at scale, with a high number of resource configurations, for different workloads and teams, and you apply a best practice on versioning and operating laC configurations, in a different lifecycle than the laC itself. This is also commonly known as GitOps.</p> <p>Use the OCI Open LZ Blueprint for these cases.</p> <p>Follow TF Best Practices</p> <p>laC solution is coded according to TF best practices, therefore it has the necessary ingredients to grow OCI laC skills in your organization.</p> <p>Strong Security (CIS Embedded)</p> <p>Security-centric solution implementing CIS OCI Foundations Benchmark version 1.2. The solution implements the recommendations and contains a script to validate any tenancy against the recommendations on the benchmark.</p> <p>Existing Patterns/Blueprints</p> <p>Several blueprints will be available for you to start using this solution. These should be used as a reference design, that you can use out-of-the-box, or tailor.</p> <p>There are already examples on each module [1] [2] [3] [4] [5] for you to start using.</p> <p>The CIS LZ v2 solution configuration will be available soon, in the same repository of the existing solution, with v3 tagging.</p> <p>The OCI Open LZ Blueprint is also available, and more patterns will be published soon.</p>
CONSIDERATIONS BEFORE USE	<p>No Terraform best practices.</p> <p>To customize or extend the solution you need laC Terraform skills. The code doesn't follow all TF best practices, with no documentation on several modules, duplication of resources, incomplete naming conventions, and no tagging.</p> <p>The consequence of this can be more effort into customizing a solution and higher code complexity as it gets more customized.</p> <p>Approach to Environments</p> <p>Caution when using the Landing Zone environments as these don't necessarily match non-production and production project/application environments. These two entities have normally different lifecycles.</p> <p>If you do not require two landing zone environments you can change the configuration to deploy only one environment. If you require a simple approach to environments, such as "1 environment - 1 deployment", you can also consider using OCI CIS v2.</p> <p>Note also that the Landing Zones environments might have also different lifecycles, different target tenancies, or target compartments, or even managing their configurations in other places. For this type of requirement, you can consider two options:</p> <ol style="list-style-type: none"> Customize the code with the environment and project/workload entities, for the proper operation of these elements. This implies coding efforts. If you require highly customized environments, from landing zones to project environments, we recommend using CIS LZ v3. The OCI Open LZ Blueprint presents an example to help you drive into these scenarios. <p>Approaches to Org. Structures</p> <p>This solution design is not prepared to reflect organization structures (LoB, BU, OEs, OpCos, Departments, etc.) in the separation of duties and operating model. To onboard organizations' structures you mainly have three approaches:</p> <ol style="list-style-type: none"> Use a simple approach and deploy one landing zone instance per organization. The drawback of this approach is higher maintenance because shared resources at the security and network levels have to be operated separately. Customize this solution with the organization's concepts at the compartment, groups, and policies level, having centrally operated shared resources for all organizations. Note that this requires customizations and coding efforts. Consider using OCI CIS v3 to onboard enterprise organizations and their organization structures into OCI focusing on configurations and not code customizations, reducing your efforts drastically. You can leverage the OCI Open LZ Blueprint to simplify this process. <p>Approach to Environments</p> <p>The approach to environments with the rule "1 environment - 1 deployment" is very flexible as it allows multiple landing zone environments. You can therefore decide the scope and content for each.</p> <p>If the above model is not enough and you require that a landing zone instance contains different project environments, you can have two options:</p> <ol style="list-style-type: none"> Customize the code with the environment and project/workload entities, for the proper operation of these elements. This implies coding efforts. If you require highly customized environments, from landing zones to project environments, we recommend using CIS LZ v3. The OCI Open LZ Blueprint presents an example to help you drive into these scenarios. <p>If you require a solution with OOTB two landing zone environments please review the two OCI OELZ v2 environments entries on the left.</p>	<p>Requires Terraform Skills</p> <p>To customize/extend the solution you need laC Terraform skills.</p> <p>If you have no laC skills and no partner responsible for this, consider using CIS LZ v3 terraform modules.</p> <p>Approaches to Org. Structures</p> <p>This solution design is not prepared to reflect organization structures (LoB, BU, OEs, OpCos, Departments, etc.) in the separation of duties and operating model. To onboard organizations' structures you mainly have three approaches:</p> <ol style="list-style-type: none"> Use a simple approach and deploy one landing zone instance per organization. The drawback of this approach is higher maintenance because shared resources at the security and network levels have to be operated separately. Customize this solution with the organization's concepts at the compartment, groups, and policies level, having centrally operated shared resources for all organizations. Note that this requires customizations and coding efforts. Consider using OCI CIS v3 to onboard enterprise organizations and their organization structures into OCI focusing on configurations and not code customizations, reducing your efforts drastically. You can leverage the OCI Open LZ Blueprint to simplify this process. <p>Approach to Environments</p> <p>The approach to environments with the rule "1 environment - 1 deployment" is very flexible as it allows multiple landing zone environments. You can therefore decide the scope and content for each.</p> <p>If the above model is not enough and you require that a landing zone instance contains different project environments, you can have two options:</p> <ol style="list-style-type: none"> Customize the code with the environment and project/workload entities, for the proper operation of these elements. This implies coding efforts. If you require highly customized environments, from landing zones to project environments, we recommend using CIS LZ v3. The OCI Open LZ Blueprint presents an example to help you drive into these scenarios. <p>If you require a solution with OOTB two landing zone environments please review the two OCI OELZ v2 environments entries on the left.</p>	<p>Customer Approach to laC</p> <p>If you have an existing strong laC practice, or predetermined way of using laC (e.g., already having existing core Terraform modules for OCI, or using a procedural approach such as Ansible, etc.) and you want to follow the same approach for landing zones, weight the pros & cons of using this solution as it can enhance or break you existing laC approach. In the case of a simpler Terraform approach in place, you can potentially reuse CIS LZ v2 modules to expand existing capabilities.</p> <p>Code Maintenance Requires TF Skills</p> <p>The TF skills required for just using the modules are very low as it's configuration-driven.</p> <p>The solution comes with highly structured Terraform written with best practices, but it's essential to be aware that to customize or change the modules code you should have proficiency in Terraform and teams responsible for those tasks.</p>

APPROACH	STANDARD / PRESCRIBED	TAILORED
	<p>2. Customize this solution with the organization's concepts at the compartment, groups, and policies level, having centrally operated shared resources for all organizations. Note that this requires customizations and coding efforts.</p> <p>3. Consider using OCI CIS v3 to onboard enterprise organizations and their organization structures into OCI focusing on configurations and not code customizations, reducing your efforts drastically. You can leverage the OCI Open LZ Blueprint to simplify this process.</p> <p>Identity Domains are Required This solution requires the use of OCI IAM with Identity Domains, and it will create resources on the default Domain and several resources related to workloads on the new Identity Domains. Be aware that the groups are created by Python scripts, which won't run with ORM, and will not be reflected in the terraform state using any of the available terraform interfaces. The main reason is the Identity Domain groups are not supported by the OCI Terraform provider yet.</p> <p>If you have an older tenancy without Identity Domains we recommend that you use CIS LZ v2.</p> <p>Caution with CIDRs There is a "Workload - VCN" match, so the more workloads, the more complex it gets to manage the CIDR blocks</p> <p>For highly complex landing zones for DC-Exits situations, we recommend using CIS LZ v3, with the OCI Open LZ Blueprint, which uses shared VCN and subnets for several workloads with NSG isolating each layer.</p> <p>Limited Network Security The use of Network Security Groups (NSG) is considered an OCI best practice, and NSGs are not available out-of-the-box with this solution.</p> <p>If you require higher levels of network isolation or higher security postures, you can customize the solution with these elements, or use CIS LZ v2.</p> <p>Requires Terraform Skills To customize/extend the solution you need IaC Terraform skills.</p> <p>If you have no IaC skills and no partner responsible for this, consider using CIS LZ v3 terraform modules.</p> <p>Home Region Deployment The Oracle Enterprise Landing Zone should be deployed to the tenancy's Home Region.</p> <p>Different Codebase from V1 This solution is a completely new approach in terms of design and code, and it doesn't propose an incremental update from OELZ v1. It has a dedicated new code repository from the previous version. In other words, If v1 is already in use, the v2 approach is disruptive.</p>	<p>Caution with CIDRs Spokes in this solution are one VCN with three subnets, one for each layer. Deciding on the spoke granularity is key to scaling the use of this landing zone for several workloads.</p> <p>Note that if the decision is "1 Workload - 1 VCN", the more workloads on the landing zone, the more complex it gets to manage the CIDR blocks.</p> <p>For highly complex landing zones for DC-Exits situations, we recommend using CIS OCI v3, with the OCI Open LZ Blueprint, which uses shared VCN and subnets for several workloads with NSG isolating each layer.</p>