Citrix and Consul-Terraform-Sync (CTS)

Achieve dynamic application services configurations
with Citrix ADC and Consul-Terraform-Sync

The Challenge

While mature DevOps teams have largely automated the majority of their CI/CD pipeline: continuous integration and continuous delivery, they have faced challenges in continuous deployment. Continuous Deployment (CD) remains the bottleneck, taking days or weeks per workload, while the previous stages are completed in minutes. CD is critical to business continuity because it configures the updated application for production, including service access, routing, and security functions. CD remains slow because of manual ticketing workflows used by networking and security teams.

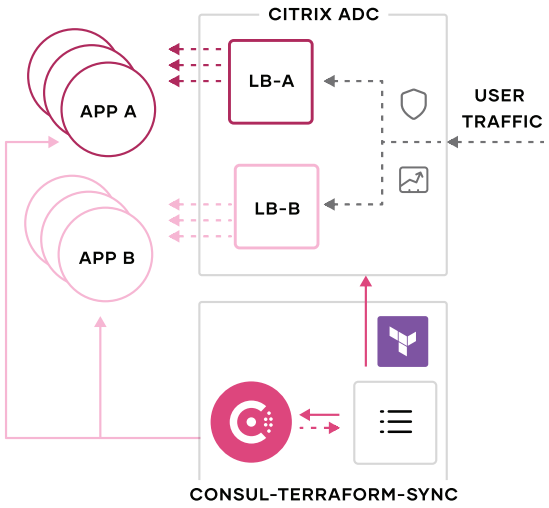
Product Integration

Now, application teams can rapidly self-configure routing, security, and service access in minutes, by using Hashicorp Consul-Terraform-Sync (CTS) with Citrix ADC! Using predefined policies and configuration code, developers can automatically update Citrix ADC to add or remove new instances of services. They no longer have to rely on creating manual tickets for platform teams to configure the deploy.

The solution also dramatically offloads monotonous workloads from network and security teams. Now, they can pre-define routing and security policies once and tailor configuration code provided by Citrix. Developers can then reuse the hardened code to create production-ready policy and config templates automatically through Terraform deployment workflows. Citrix provides the policy and configuration code for all routing and robust security functions, including Web Application Firewall, Bot management and API security. Once the configurations are deployed, CTS performs automated clean-up such as closing the open ingress gateway to ensure it doesn't become a vulnerability.

Finally, the integrated solution allows the entire platform team to focus on review and approval of the automated configurations and workflows using the CTS dashboard. CTS also provides a production system of record that centralizes all changes in one place, so they are visible and easily auditable. When customers see predictable results, some even opt for zero human involvement.

How it Works



- App teams add (update, delete) new instances of services
- Consul as a service discovery tool picks up changes related to services
- Consul-Terraform-Sync will auto-generate ADC configuration as Terraform resources
- Terraform will configure services, service group in Citrix ADC
- ADC will start routing traffic to new instances

Figure 1: Caption: Dynamically update service changes in ADC using Citrix ADC CTS modules

The Terraform module will create a service group for each service and then bind to it service members according to the number of instances for each service as sourced from the Consul service discovery. This will result in service-groups being in sync with the services sourced from Consul. Adding, deleting or modifying services in Consul will result in service groups and its servicemembers reflecting these changes.

Existing CTS modules configures the services and service group in Citrix ADC corresponding to backend applications. These modules can be enhanced as per your desired use case e.g you can modify the template to configure a new load balancing vserver in Citrix ADC on the fly if App owners want to expose new app to end users. Here is an example of ADC terraform configuration for the same that needs to be run separately.

Use Case 1: Agility of application updates

Most enterprises are refreshing their customer applications almost daily to drive more engagement. But the actual time to update a feature set or add instances for scaling, can still take days to weeks due to the complexity of ensuring availability, performance and security. The result is that multiple development and operations teams must work in parallel to deploy the many feature updates that are needed.

With Citrix ADC and Hashicorp Consul-Terraform-Sync and Terraform, enterprises can update application instances in minutes with high confidence that they will deliver the best experience securely.

Routing traffic to new instances of service or stop traffic to terminated instances:

1. App teams add or delete instances of the application, to update features or scale up/down.
2. Consul discovers these changes because it continuously monitors the application environment.
3. Network and security teams can pre-define routing and security policies once and tailor configuration code provided by Citrix, including Web Application Firewall, Bot management and API security.
4. Consul-Terraform-Sync reuses the code and auto-generates the new ADC configuration as Terraform resource and workflows.

5. Terraform configures production-ready services and service groups in Citrix ADC for the changed instances.
6. The platform team reviews and approves the automated configurations and workflows using the CTS dashboard.

The Citrix-Hashicorp difference is to reduce application / service updates from weeks to minutes without any loss of application availability, performance or security. Conversely, the application experience and security will likely improve and be more consistent because it is using predefined policies and hardened configuration code.

Use Case 2: Reliable, consistent app migration to hybrid and multi-cloud

Migrating a major enterprise application to hybrid cloud or multi-cloud can reduce capex, increase operating efficiency, improve user-experience or provide back-up to improve up-time. But the project is error-prone because of the need to preserve and validate hundreds of security policies or routing rules. Additionally, operating on multi-cloud requires IT teams to maintain consistency across different cloud providers, manually.

Now with Citrix ADC and Hashicorp Consul-Terraform-Sync and Terraform, cloud migration can be error-free and operations can maintain consistency automatically.

Replicating an application in a new cloud:

1. App teams create the application in the new cloud using the cloud provider's provisioning tools
2. Network and security teams identify the routing and security policies only once and tailor the configuration code provided by Citrix for the desired application behavior.
3. Automatically create new load balancing vserver and bind to services to expose new application to end users.
4. Consul-Terraform-Sync automatically apply global routing configuration, security polices every time App team introduces new app
5. Terraform configures services and service groups in ADC for the new application.
6. The platform team reviews and approves the automated configurations and workflows using the CTS dashboard.