Citrix Security Development Processes

Introduction

Citrix has a dedicated Product Security Team (PSEC) responsible for the Secure Development Lifecycle (SDL) of Citrix products. This team works hand-in hand with the Product Development Team during the development lifecycle of the product. Additionally, Citrix has a dedicated Security Response Team that handles, prioritizes and coordinates the remediation of any externally reported security issue.

This document provides an overview of the security work and processes that are performed on the Citrix product line.

Per-release – feature-specific security work

For each upcoming release, PSEC engages with the development teams to evaluate the security implications of any new features associated with the release in order to kick start the SDL process as applicable. This is a derivative of the industry standard Security Development Lifecycle, as defined by Microsoft. As part of the SDL process, the following stages are executed by the product security team:

Threat Modeling

Every new feature, service, or new interactions between existing services go through an extensive Threat Model where the security team and the development architects get together to identify the assets, attack surface, attackers and corresponding threats in the system. Then we proceed with making any changes in the design to mitigate the threats. In cases where the threats can be mitigated by implementation or config changes, those are implemented as well. Threat models also point to security tests which get implemented and added as unit or integration tests.

Code Review

The new feature has to go through and extensive manual code review for any security sensitive changes like multi-tenancy flow, RBAC, cryptographic code, authentication/authorization, etc. While performing manual code reviews, we try to identify issues that would otherwise be missed by SAST tools including but not limited to logic errors leading to broken authentication or memory corruption. Over and above the manual code review of security sensitive section, the code goes through an automated code review tool as well which is integrated and managed by the build team.

Penetration Testing

Pen testing is done in multiple phases for each release:

1. We have an in-house automated scanning framework that aggregates and runs an array of industry standard vulnerability scanners like Qualys QualysGuard, Tenable Nessus, Rapid7 Nexpose, BeyondTrust(eEye) Retina, HP Webinspect and IBM AppScan on our product to identify any defects and mitigate/fix those issues. These scans are run as part of our build cycle.
2. We run an extensive 3-4 weeks long manual pentesting where the security team focuses on identifying any OWASP-Top-10 and CWE-Top-25 style defects. Over and above these, we look for defects in business logic flows which can lead to Elevation of Privilege or Cross Tenant bypass style attacks. During this phase, we may use tools like application proxies and exploitation frameworks to assist in our testing
3. We also perform fuzzing, and integrate fuzzing tools such as Peach and AFL with applicable unit test cases to improve coverage for suitable functions and protocols.

Release Criteria

We do not allow products to ship with outstanding critical or high severity security vulnerabilities. All new Critical and High Severity defects (CVSS 7.5 to 10.0) are deemed as release blockers.

Per-quarter and annually – system wide and component-specific security work

Every quarter, we perform component-specific security reviews to account for changes that have occurred to the component across feature releases. For each component, we update the threat models, recommend test cases to developers, and perform penetration testing using the same pentesting approach we follow for our feature-specific work.

Once a year we do a high-level systemwide threat model and review that is used to inform our approach to the quarterly component-based security review.

Response process

We have a vulnerability response process with a dedicated Security Response Team where any customer or security researcher can report a vulnerability by sending us an email on secure@citrix.com. We use the issues reported on this channel as a feedback to improve on the features and components that tend to attract externally reported issues. Given this insight, we prioritize these components/features for a retrospective Security Development Lifecycle, and use the opportunity to identify and implement class fixes for the security issues.