Carnegie Mellon University
Software Engineering Institute

Crucible

A Cyber Simulation Framework

Introducing Crucible

Crucible delivers low-cost dynamic virtual environments that maximize interoperability and scalability for cyber simulations. Crucible leverages open standards and modular application programming interfaces to deliver an immersive, browser-based experience. Virtual environments are built using an "infrastructure-as-code" approach to maximize reuse and iteration. Crucible strives to be the Linux of cyber modeling, simulation, and exercises—powerful, open-source, highly extensible, and cost effective.

Key Features

1. Open-source cyber simulation framework
2. Modular design featuring extensive APIs
3. Customizable, immersive, browser-based user interface
4. "Infrastructure as code" approach to topology building
5. Flexible integration of powerful, third-party open-source tools
6. Scenario-based simulation and assessment
7. Efficiency through automation
8. Interoperability through open standards

Addressing Persistent Challenges

Most cyber simulations today are developed manually inside proprietary systems. Manual topology configuration negatively impacts time, cost, and quality due to the inevitable introduction of human error that occurs when re-implementing configurations. Furthermore, manual execution of scenario events introduces human error and inhibits automated execution. Finally, the use of proprietary, closed-source software increases costs and can lead to vendor lock-in. In response to a decade of experiencing these frictions while creating and facilitating Department of Defense team-based cyber exercises, the SEI developed the Crucible cyber simulation framework.

Enabling Cyber Simulations

Crucible simulates environments for individual labs and team-based exercises as well as operational test and rehearsal events. Crucible labs and exercises place cyber operators into realistic virtual environments that feature a modeled topology, simulated user activity, and scenario events that prompt performance of mission-essential tasks and/or individual qualification requirements. Crucible also enables rapid operational test and rehearsal events thanks to version-controlled modeling of high-fidelity topologies, simulation of realistic network activities, and extensive environment instrumentation to enable detection of state changes. These scenario-based simulations can either require a live white-cell to facilitate or be fully automated.

Simulation Workflow

Crucible automates the workflow for creating, deploying, facilitating, and assessing simulations. Any Crucible content developer can create a new simulation template by specifying a topology, scenario, assessments, and user interfaces.

Coding a Topology

Crucible integrates several popular open-source solutions to form a topology-creation application that is both user-friendly and tailored to ease simulation development. Terraform, an "infrastructure-as-code" tool, enables scripted deployment of cyber infrastructure. Further provisioning of services to this infrastructure is achieved using Ansible, a software provisioning, configuration-management, and application-deployment tool. GitLab, a version control system and code-repository, is used to store topology modules and ease code reuse.

Crafting a Scenario

Crucible integrates a number of open-source solutions into a scenario-automation application that enables the effective planning of scenario events and environment sensing. Scenario events are mapped to specific simulation objectives. The MITRE ATT&CK framework is used to map each stage of inject execution. StackStorm, an event-driven automation platform, scripts scenario events and senses the simulation environment. SEI's GHOSTS Non-Player Character (NPC) orchestration framework deploys and shapes the activities of NPCs throughout the simulation.

Creating Assessments

Crucible's assessment applications simplify the validation of mission readiness using knowledge-based and performance-based assessments. Assessment reports map training objectives to scenario events to performance assessments. Moodle/H5P, an interactive learning management system, eases the embedding of interactive quiz content. Assessments and other user-experience data can be captured and reported. The Experience API (xAPI) and a learning record store, such as Learning Locker, can enable analysis across all types of learning experiences.

Designing User Interfaces

Crucible's user-experience application enables simulation developers to customize the user interface for each participant—shaping how scenario information, assessments, and virtual environments are presented through the use of integrated applications. Simulation developers can also define teams and assign permissions to specific cyber terrain and scenario events. Crucible relies upon powerful open-source application frameworks, including Angular and .NET Core, to deliver this responsive browser-based user experience.

Conducting a Simulation

Crucible users can schedule the launch of a scenario simulation at a desired time or can join instances of already running simulations. Resource consumption can be effectively managed with the assignment of resource quotas. Following the simulation, reports provide a summary of knowledge and performance assessments.

Operational Deployment

Crucible applications leverage the SEI's Identity Service or any other single-sign-on solution that supports the OAuth2 or OpenID Connect authentication protocols. Crucible applications are deployed as containers using Docker, a platform that employs operating system level virtualization to isolate containers from one another. Container deployment, scaling, and management services can be obtained using either Docker Swarm or Kubernetes, two popular container-orchestration systems.

Learn More About Crucible

Crucible has enabled numerous large- and small-scale DoD cyber exercises since 2018—successfully driven by its modular design and "infrastructure-as-code" approach. Crucible enables the deep practice needed to increase operator performance. While originally available only to the Department of Defense, Crucible is now available to the public under open-source licensing and will be ready for broad community adoption by the fall of 2020. Watch for updates on the continued evolution of Crucible at: **http://www.sei.cmu.edu/go/crucible**. For more information, email **info@sei.cmu.edu**.

About the SEI

The Software Engineering Institute is a federally funded research and development center (FFRDC) that works with defense and government organizations, industry, and academia to advance the state of the art in software engineering and cybersecurity to benefit the public interest. Part of Carnegie Mellon University, the SEI is a national resource in pioneering emerging technologies, cybersecurity, software acquisition, and software lifecycle assurance.

Contact Us

CARNEGIE MELLON UNIVERSITY
SOFTWARE ENGINEERING INSTITUTE
4500 FIFTH AVENUE; PITTSBURGH, PA 15213-2612

sei.cmu.edu
412.268.5800 | 888.201.4479
info@sei.cmu.edu