REDTEAMTP

Generate, Commit, and Deploy Offensive Infrastructure using GitHub Actions

Alex Martirosyan

TABLE OF CONTENTS

01

Generate

02

Commit

03

Deploy

Ø1

Generate

Creators

Created by Alex Martirosyan and Artur Saradzhyan

https://github.com/CultCornholio/RedTeamTP/

almart (Alex Martirosyan) (Lead Penetration Tester)

sarartur (Art Sarad) (Senior Software Engineer)

Acquire Infrastructure

“ Adversaries may buy, lease, rent, or obtain infrastructure that can be used during targeting. A wide variety of infrastructure exists for hosting and orchestrating adversary operations. ”

|| T1583 - MITRE ATT&CK® ||

Purpose of RedTeamTP

CI/CD automated red team infrastructure deployment using GitHub Actions.

Building upon the foundation laid by Ralph May (WarHorse)

Collaborative environment for deploying *non-static* offensive operations

HashiCorp Terraform

ANSIBLE

Development Process

Industry wide problem-set for offensive tooling/development

Offsec != Software Engineering

Creation of projects that all do similar things

Poor architecture choices, no test cases, lack of scalability

01. GENERATE

Development Goals

RedTeamTP objective was to leverage **existing** projects to scale

RedTeamTP should be able to **scale**

Create unit tests and follow standard frameworks

Environment must be user friendly and collaborative for offensive teams

Extending WarHorse

https://www.antisyphontraining.com/course/hackerops-with-ralph-may/

HackerOps with Ralph May educates offensive practitioners DevOps principles

WarHorse was designed with OPSEC Safe choices (Terraform/Ansible)

Can “generate” compliant playbooks to defeat static deployment issue

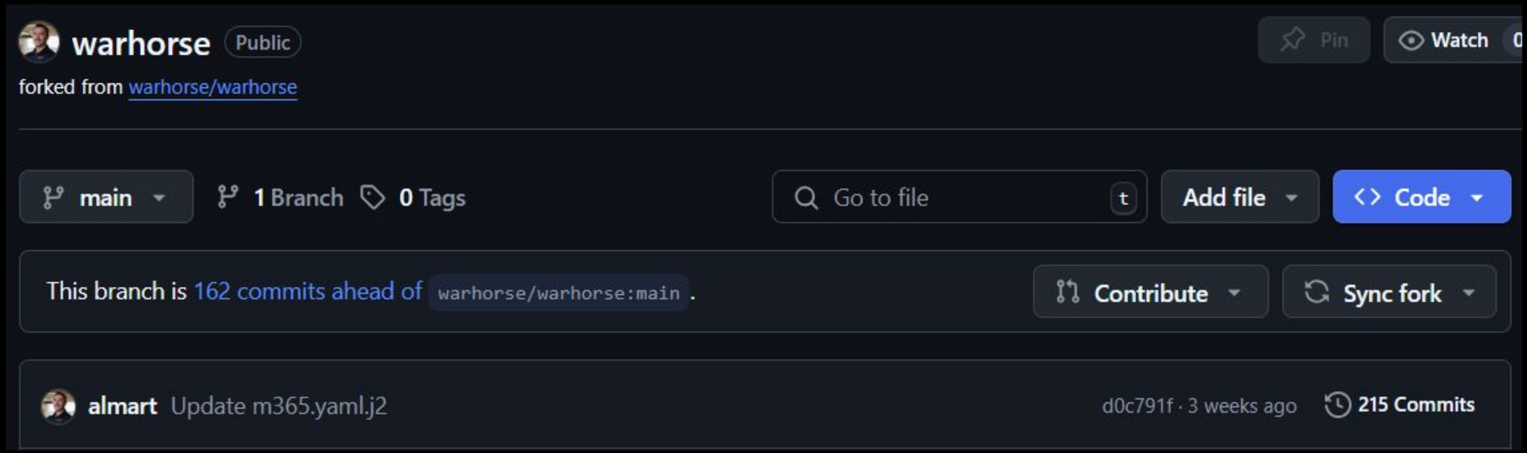
warhorse/warhorse: Infrastructure Automation

Extending WarHorse

Updating WarHorse to support latest version of Terraform/Ansible

Updating relevant roles (CobaltStrike/Evilginx/Mythic)

OPSEC safety updates for several projects



warhorse Public

forked from warhorse/warhorse

This branch is 162 commits ahead of `warhorse/warhorse:main` .

almart Update m365.yaml.j2 d0c791f · 3 weeks ago 215 Commits

RedTeamTP Overview

Leverages up-to-date WarHorse version to make compliant yaml files

Application made within a GitHub repository

PR’s support collaborative environment to learn and validate deployments

Python based app for compliant yaml files and local testing

Actions is used as the “runner” for deployments via WarHorse

02

Commit

RedTeamTP Generation

We use RedTeamTP to generate compliant WarHorse configurations

GitHub Actions uses a container to deploy the playbook

PR’s can be modified on-demand and reviewed by an offensive team

All secrets handled for you via GitHub Secrets

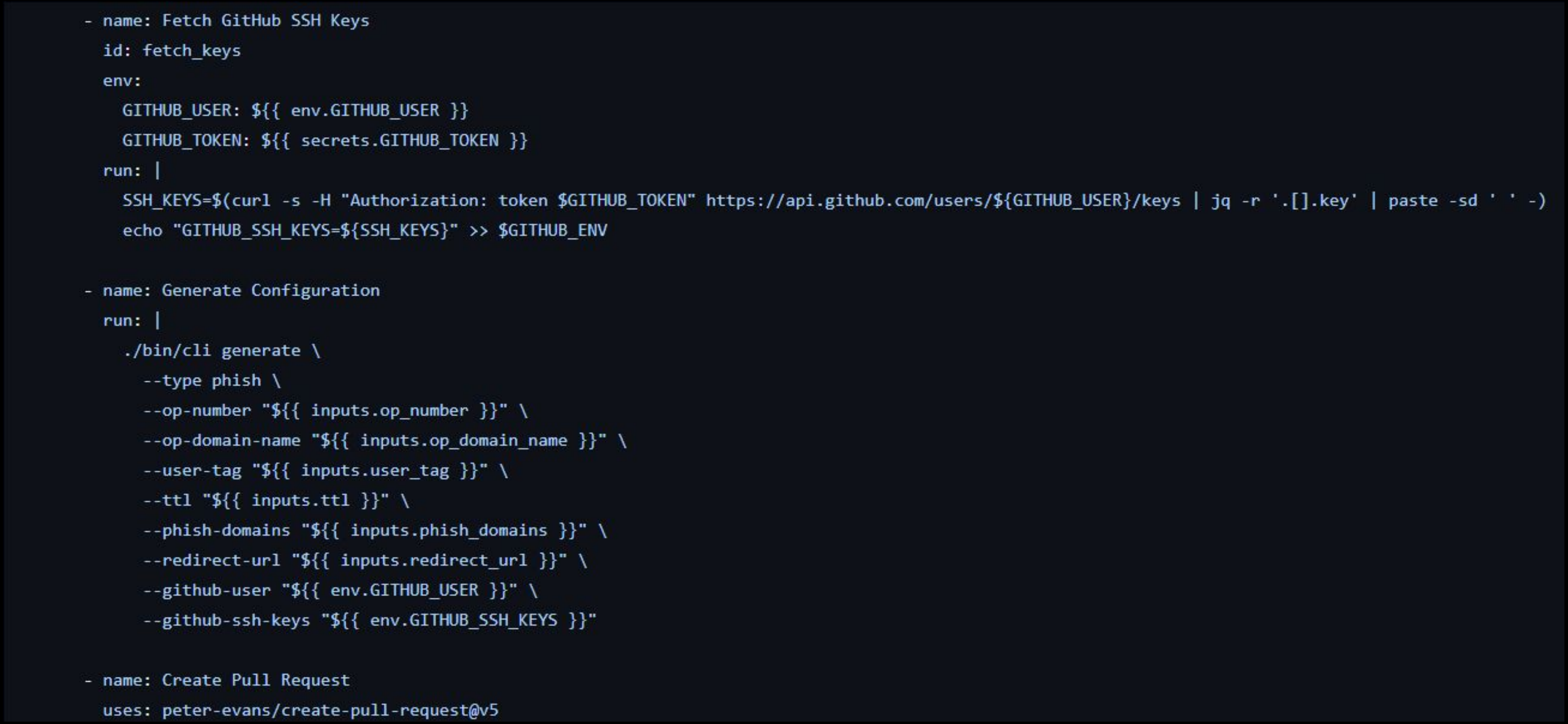
RedTeamTP Generation

Actions leave verbose logging for in-depth reviews

Buy a domain, grab a coffee, and run the workflow...

No more dependency hell or relying on a infrastructure wizard

Workflow Generate



```
- name: Fetch GitHub SSH Keys
  id: fetch_keys
  env:
    GITHUB_USER: ${{ env.GITHUB_USER }}
    GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
  run: |
    SSH_KEYS=$(curl -s -H "Authorization: token $GITHUB_TOKEN" https://api.github.com/users/${GITHUB_USER}/keys | jq -r '.[].key' | paste -sd ' ' -)
    echo "GITHUB_SSH_KEYS=${SSH_KEYS}" >> $GITHUB_ENV

- name: Generate Configuration
  run: |
    ./bin/cli generate \
      --type phish \
      --op-number "${{ inputs.op_number }}" \
      --op-domain-name "${{ inputs.op_domain_name }}" \
      --user-tag "${{ inputs.user_tag }}" \
      --ttl "${{ inputs.ttl }}" \
      --phish-domains "${{ inputs.phish_domains }}" \
      --redirect-url "${{ inputs.redirect_url }}" \
      --github-user "${{ env.GITHUB_USER }}" \
      --github-ssh-keys "${{ env.GITHUB_SSH_KEYS }}"

- name: Create Pull Request
  uses: peter-evans/create-pull-request@v5
```

Workflow Generate

Action runner leverages Python application to prepare the yaml file

Uses the actors public SSH key on-demand for deployments

PR’s can be modified on-demand and reviewed by an offensive team

03

Deploy

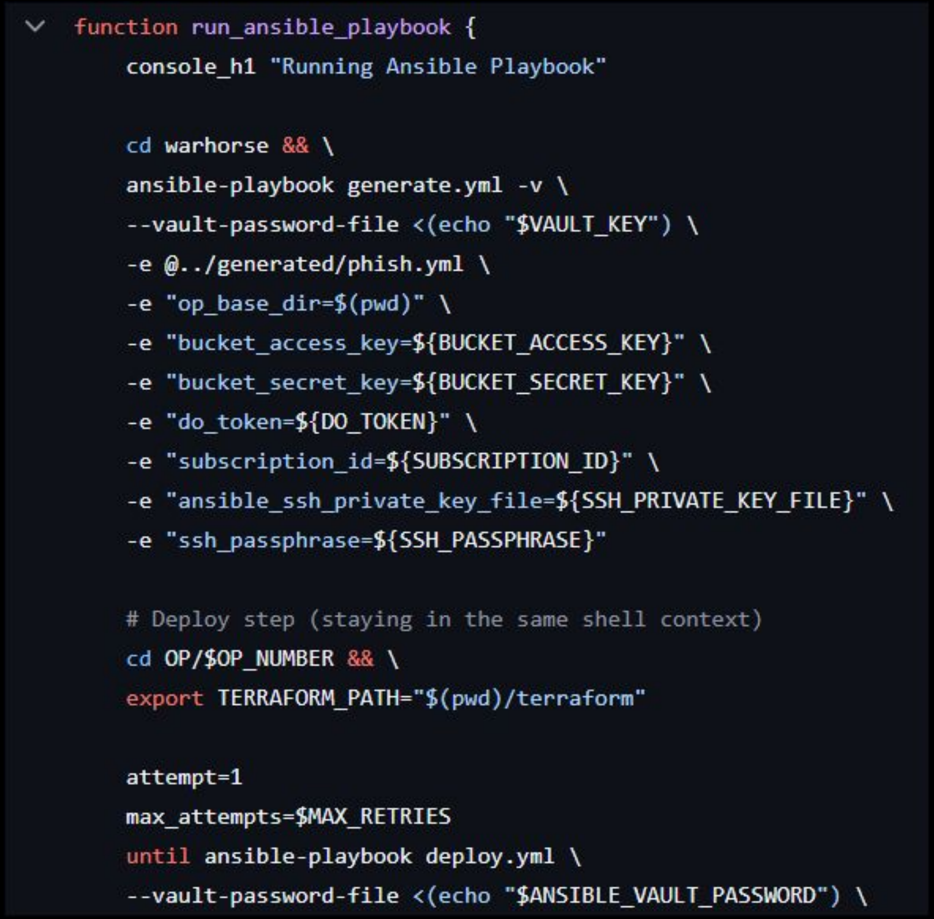
03. DEPLOY

Workflow Deploy

Workflows pre-built to utilize secrets and run WarHorse

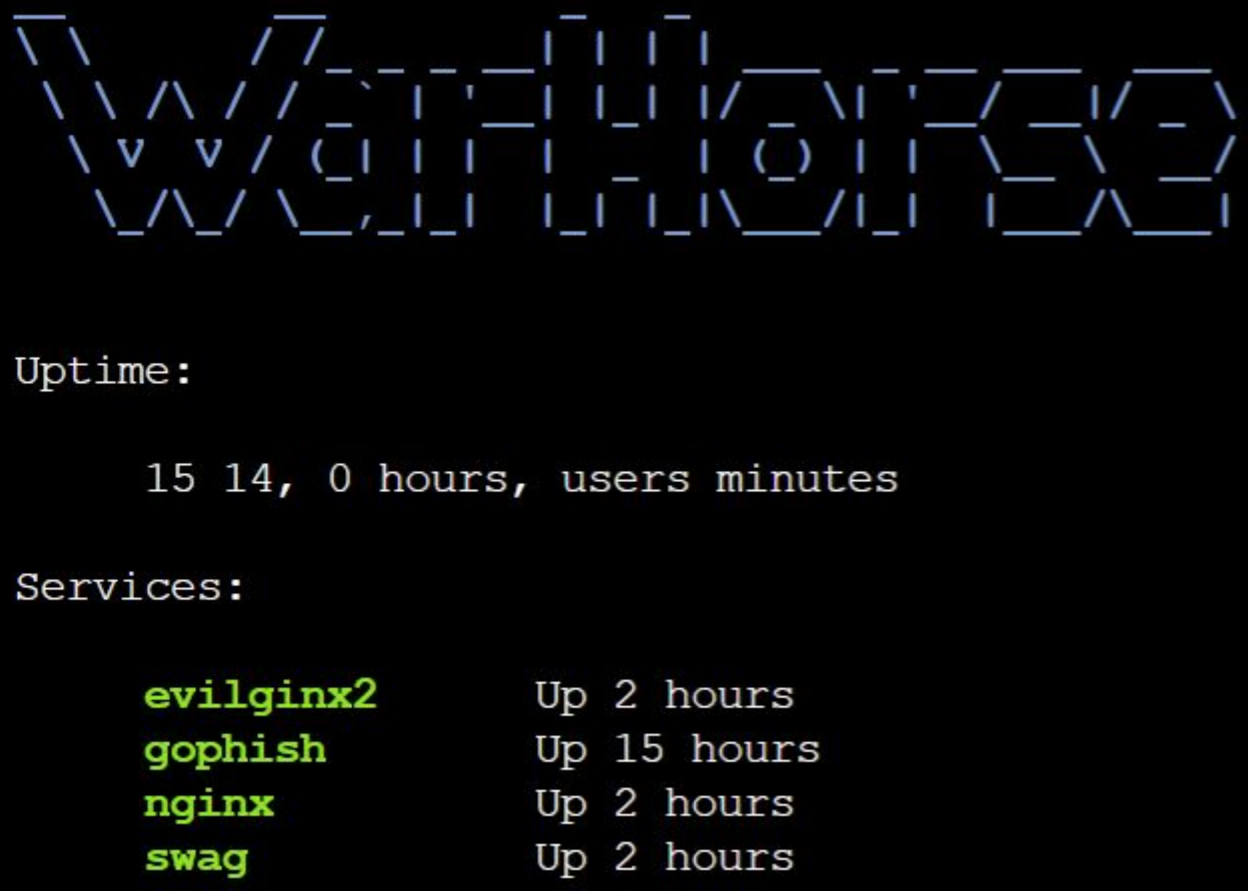
GitHub Runner deploys in a container to leave an artifact

Re-run deployments and develop configuration files for the team



```
function run_ansible_playbook {
    console_h1 "Running Ansible Playbook"

    cd warhorse && \
    ansible-playbook generate.yml -v \
    --vault-password-file <(echo "$VAULT_KEY") \
    -e @../generated/phish.yml \
    -e "op_base_dir=$(pwd)" \
    -e "bucket_access_key=${BUCKET_ACCESS_KEY}" \
    -e "bucket_secret_key=${BUCKET_SECRET_KEY}" \
    -e "do_token=${DO_TOKEN}" \
    -e "subscription_id=${SUBSCRIPTION_ID}" \
    -e "ansible_ssh_private_key_file=${SSH_PRIVATE_KEY_FILE}" \
    -e "ssh_passphrase=${SSH_PASSPHRASE}"

    # Deploy step (staying in the same shell context)
    cd OP/$OP_NUMBER && \
    export TERRAFORM_PATH="$(pwd)/terraform"

    attempt=1
    max_attempts=$MAX_RETRIES
    until ansible-playbook deploy.yml \
    --vault-password-file <(echo "$ANSIBLE_VAULT_PASSWORD") \
```

Workflow Deploy



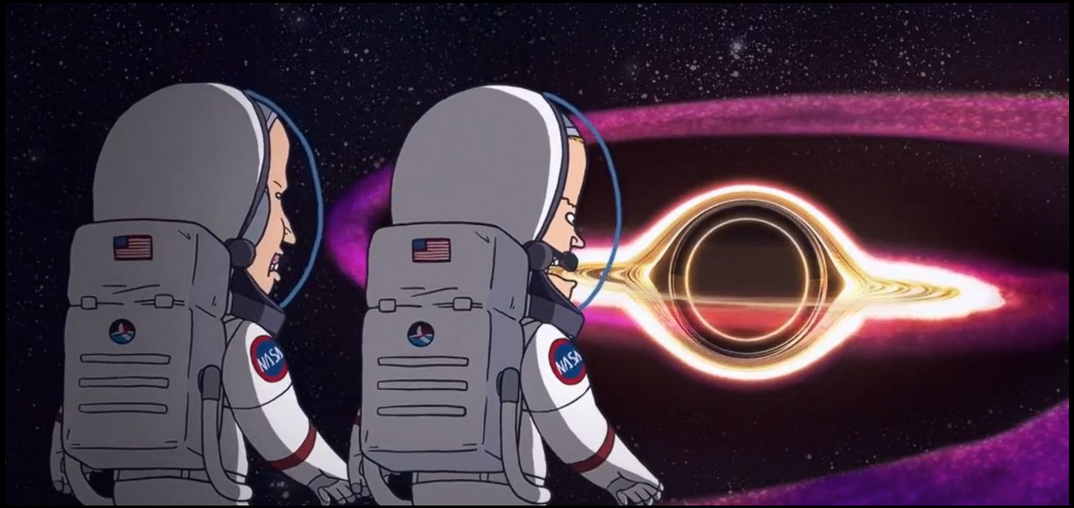
```
__        __          _   _
\ \      / /_ _ _ __| | | | ___  _ __ ___  ___
 \ \ /\ / / _` | '__| |_| |/ _ \| '__/ __|/ _ \
  \ V  V / (_| | |  |  _  | (_) | |  \__ \  __/
   \_/\_/ \__,_|_|  |_| |_|\___/|_|  |___/\___|

Uptime:

     15 14, 0 hours, users minutes

Services:

     evilginx2      Up 2 hours
     gophish        Up 15 hours
     nginx          Up 2 hours
     swag           Up 2 hours
```

Future Work

Easy to update and add new roles to WarHorse

Test and deploy using RedTeamTP

Iterate and scale using the project!



THANK YOU

Alex Martirosyan

https://x.com/almartiros

amartirosyan@wolfandco.com

www.densecure.com